



UNIVERSIDAD NACIONAL  
AUTÓNOMA DE  
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN  
INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y  
SISTEMAS  
FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN

ÁREA DE CÓMPUTO CIENTÍFICO

SIMULACIÓN DE UN MODELO PARA LA ACCIÓN DE UN NANOBOT SOBRE  
UNA OBSTRUCCIÓN LOCALIZADA EN UN CAPILAR

TESIS

QUE PARA OPTAR POR EL GRADO DE  
DOCTOR EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

PRESENTA:  
ROBERTO FEDERICO ORTIGOZA DOMINGUEZ

DRA. SUEMI RODRÍGUEZ ROMO  
FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN

CIUDAD UNIVERSITARIA, CIUDAD DE MÉXICO, MARZO 2019



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

“Si alguno desea el mucho saber, ella es lo que sabe lo pasado, y forma juicio de lo futuro; conoce los artificios de los discursos, y las soluciones de los argumentos; adivina los prodigios y maravillas antes que sucedan, y los acontecimientos de los tiempos y de los siglos”.

*Sab 8, 8.*

## Agradecimientos.

- Ante todo agradezco a Dios por sin su ayuda nada es posible.
- Agradezco a la Dra. Suemi Rodríguez Romo por su invaluable ayuda, por sus consejos y correcciones siempre que fue necesario. Seguirá siendo una insustituible formadora de excelentes profesionistas en nuestra valiosa institución.
- Agradezco a mi mamá María del Carmen Domínguez Anaya que es mi sostén, mi motivación y mi orgullo. Para ella todo mi respeto y cariño. Todo el empeño que puse en mi trabajo se lo dedico a ella.
- Agradezco a mi hermana Iraís Margarita Ortigoza Domínguez que es para mí motivo de inspiración y admiración.
- Agradezco a la Universidad Nacional Autónoma de México que siempre me proveyó de los recursos necesarios para mi trabajo.
- Agradezco al Consejo Nacional de Ciencia y Tecnología por el apoyo prestado para la realización del presente trabajo.
- Finalmente agradezco a mis profesores, tutores, familiares, compañeros, amigos y personas que me impulsaron y que confiaron en mí. A ellos también les debo que este trabajo haya sido posible.

# Índice.

Título.	1
Motivación.	1
Hipótesis.	1
Objetivos.	1
Introducción.	3
<i>Capítulo 1.</i>	7
<b>1.1 Planteamiento del Problema y Antecedentes.</b>	7
<b>1.2 Nanomedicina.</b>	11
<b>1.3 Paralelización.</b>	13
<b>1.4 La Ecuación de Redes de Boltzmann.</b>	14
<i>Capítulo 2</i>	29
<b>2.1 Modelo Conceptual, Geometría.</b>	29
<b>2.2 Modelo Matemático y Estrategia computacional.</b>	34
<b>2.3 Algoritmo utilizado.</b>	41
<i>Capítulo 3</i>	47
<b>3.1 Arquitectura de los Dispositivos de Cómputo Unificado y Breve Introducción al Estándar de Paralelización MPI.</b>	47
<b>3.2 Estrategia de Paralelización.</b>	53
<b>3.3 Detalle de la Implementación.</b>	61
<b>3.4 Optimización.</b>	66
<i>Capítulo 4.</i>	69
<b>4.1 Resultados y Discusión.</b>	69
<b>4.2 Métricas.</b>	80
<b>4.3 Conclusiones.</b>	84
Apéndice A – Especificaciones del Hardware Utilizado.	87
Apéndice B – Adimensionalización.	90
Apéndice C – Cálculo del Momento de Inercia de la Célula de Sangre Roja.	92

Bibliografía.	96
Páginas Web Consultadas.	104

## Título.

### **Simulación de un Modelo para la Acción de un Nanobot sobre una Obstrucción Localizada en un Capilar.**

## Motivación.

- Desarrollo de una herramienta poderosa de cómputo científico que sirva interdisciplinariamente a investigadores que se esfuerzan en combatir enfermedades mortales, tales como aquellas de tipo circulatorio desde múltiples escalas no tratadas con anterioridad y que hace uso de nuevas e innovadoras tecnologías, para optimizar la eficiencia de su ejecución.

## Hipótesis.

- El cómputo científico es una herramienta poderosa en la investigación de métodos terapéuticos avanzados tales como el uso de “nanobots<sup>1</sup>”, que permiten estudiar y analizar su acción en el flujo sanguíneo, en particular sobre un tumor<sup>2</sup> o lesión localizada en un vaso capilar, además de obtener resultados numéricos de manera razonablemente exacta y eficiente aplicando técnicas computacionales de paralelización heterogénea.

## Objetivos.

1. Se propondrá un modelo para simular la acción de un “nanobot” sobre una obstrucción en un vaso capilar, por ejemplo un tumor.

---

<sup>1</sup> Dispositivos que aprovechan las ventajas físicas, químicas y biológicas de materiales fabricados a escala nanométrica ( $\leq 100 \text{ nm}$ , aunque en realidad se utilizan dispositivos más grandes v. gr. Los dispositivos mencionados en (Fontana, Licciardi, Mansueto, Schillaci & Giammona, 2001),  $220 \text{ nm} - 320 \text{ nm}$ ) y que para nuestro caso consisten de una substancia bioactiva (o agente terapéutico) y un polímero como medio de transporte y liberación de la misma capaz de entregar su carga en sitios y tiempos específicos (Donega, 2014) y (Hossain, Hossainy, Bazilevs, Calo, & Hughes, 2012).

<sup>2</sup> Masa de células transformadas, con crecimiento y multiplicación anormales [W1].

2. Se creará un algoritmo para tal fin y software asociado para simular el modelo.
3. Se realizará la paralelización del software mediante un ambiente heterogéneo de cómputo, que consistirá de una combinación de clústeres de CPUs que tengan al menos una Unidad Gráfica de Proceso (GPU por sus siglas en inglés) asociada por CPU.

## Introducción.

El estudio del flujo sanguíneo es un campo esencial y estratégico de investigación ya que implica un mejor entendimiento de los procesos y mecanismos involucrados en el mismo, lo cual es de suma importancia para el desarrollo de nuevas estrategias terapéuticas y dispositivos médicos que ayuden a combatir enfermedades tales como el cáncer o la arterioesclerosis, siendo esta última la principal causa de muerte en los países occidentales (Melchionna, 2011).

En especial la cama capilar es una estructura particularmente interesante debido entre otras razones a que no colabora significativamente a la resistencia periférica total al flujo<sup>3</sup>. Esto debido a su enorme número (según estimaciones entre 10,000 y 40,000 millones de capilares, lo que se traduce en una superficie total enorme de 1000 m<sup>2</sup> (Lauralee, 2011), (Silbernagl & Despopoulos, 2011)) y a que en conjunto con la mínima velocidad de flujo (0.02 cm/s – 0.1 cm/s) la hace óptima para el intercambio de gases, nutrientes y líquidos con todas las células del cuerpo (Lauralee, 2011).

La simulación computacional se ha convertido en una herramienta valiosa para la investigación médica, ya que en muchos casos los estudios *in vivo*<sup>4</sup> son muy difíciles de realizar con la mayoría de los parámetros reales y bajo ciertas condiciones fisiológicas, principalmente en los lugares de más difícil acceso<sup>5</sup> (como los vasos capilares), además de estar sujetos a consideraciones éticas. A excepción de lo último mencionado algo similar se puede decir de los estudios *in vitro* (Chandran, Udaykumar, & Reinhardt, 2010). En especial la simulación del flujo sanguíneo es desafiante por que ciertamente es uno de los sistemas más complejos que existen dadas todas las variables, componentes y circunstancias fisiológicas involucradas.

Asimismo existe ya una amplia gama de aplicaciones biomédicas sin precedentes mediante la utilización del cómputo científico, que van desde el

---

<sup>3</sup> Es una medida de la resistencia al flujo sanguíneo a semejanza de la ley de Ohm (Silbernagl & Despopoulos, 2011).

<sup>4</sup> *In vivo* significa que la investigación y la experimentación se hace en un organismo vivo, en cambio *in vitro* significa que dicha investigación y experimentación se hace fuera de un organismo vivo, aislando una parte del mismo dentro de un ambiente artificial controlado (v. gr. Un tubo de ensayo) [W2].

<sup>5</sup> Como un ejemplo en (Hossain, Hossainy, Bazilevs, Calo, & Hughes, 2012) se menciona que para el caso del uso de la nanotecnología en el tratamiento de placas ateroscleróticas en las arterias coronarias, sólo las herramientas numéricas pueden proporcionar información crucial no disponible por cualquier otro método.

estudio de terapias novedosas que hacen uso de la nanotecnología (Hossain, Hossainy, Bazilevs, Calo, & Hughes, 2012) hasta la planeación de pre-tratamientos para intervenciones y procedimientos quirúrgicos vasculares según se menciona en (Tirado–Ramos, Groen, & Sloot, 2005).

Recientemente se ha incrementado el espectro de aplicaciones de los “nanobots” (o nanopartículas) por ejemplo: en productos de cuidado personal, tecnologías de inyección de tinta, farmacéuticos, etc. (Ververg, Alexeev, & Balaazs, 2006). Esta última aplicación es en la que se enfoca nuestro trabajo con el se pretende generar una herramienta de cómputo que sirva para estudiar los mecanismos involucrados en terapias tales como la destrucción de tumores cancerosos localizados en los vasos sanguíneos y que impiden el libre flujo de la sangre a través de ellos.

Es debido a lo mencionado en los dos últimos párrafos que surge nuestra motivación para continuar las investigaciones en este campo, lo que llevaremos a cabo desarrollando una simulación computacional en tercera dimensión del flujo sanguíneo en un vaso capilar. Esto se hace bajo condiciones fisiológicas normales y en la cual se incluye la liberación de un fármaco adecuado para este tipo de terapias, el cual se encarga de destruir una obstrucción que obstaculiza el libre flujo en el vaso sanguíneo. La liberación se realiza teniendo como origen una nanopartícula de tamaño comercial también adecuado para tratamientos similares, siendo esto la primera vez que se realiza pues normalmente se simulan “microcarriers<sup>6</sup>”.

Ahora bien, desde hace algunos años se han venido haciendo progresos cada vez más interesantes en la utilización de métodos numéricos basados en un modelo continuo para la simulación del flujo sanguíneo de utilidad en investigaciones biomédicas importantes (ver por ejemplo (Soudah, et al., 2013) (basada en el método del elemento finito) y (Gohil, et al., 2011) (análisis comparativo entre implementaciones basadas en los métodos del elemento finito y volumen finito)), pero lo cierto es que en cuanto a las simulaciones de flujo de fluidos en general los *modelos de las redes de Boltzmann*<sup>7</sup> han mostrado ser más eficientes que los anteriores (Clausen, Reasor, & Aidun, 2010), al menos en lo que a paralelización se refiere<sup>8</sup>.

---

<sup>6</sup> O también llamados “nanocarriers”, que son microcápsulas que consisten de una capa polimérica porosa y un fluido interior capaces de entregar su carga en lugares y tiempo específicos (Verberg, Alexeev & Balazs, 2006).

<sup>7</sup> Se podrían englobar dichos modelos en un método numérico general (en la sección 1.4 “La Ecuación de Redes de Boltzmann” se explicará con más detalle esto) al cual llamaremos **método de las redes de Boltzmann** (MRB). Estos métodos están basado en la ecuación del mismo nombre obtenida en 1872 por el insigne

Una de las razones por las cuales es importante emplear una técnica de paralelización en éste tipo de simulaciones es que, aparte de aprovechar las oportunidades que brinda la tecnología actual, la simulación de la dinámica de fluidos (en particular de la sangre) es en si misma computacionalmente demandante (Xiong, et al., 2012). Además de que el tipo de fenómeno es complejo dada la multiescala (desde nanómetros a micras) a la que se quiere modelar pues involucra cuerpos suspendidos con geometrias complejas en movimiento; por lo que es necesaria alguna técnica de paralelización a lo cual, dada la intrínseca localidad del paso de colisión y la dependencia de sólo los nodos vecinos para el paso de propagación, hace que el MRB se adapte de manera casi natural<sup>9</sup> (Crowl & Fogelson, 2010), (Krüger, Varnik, & Raabe, 2011), (Pan, Prins, & Miller, 2004) a las arquitecturas de los sistemas modernos de cómputo de alto rendimiento (Jonathan B., 2014).

Aunque recientemente se han realizado varias publicaciones en donde se simula computacionalmente el flujo sanguíneo bajo varios MRBs donde se han estudiado y simulado aplicaciones de los “nanobots” que transportan fármacos (ver el estado del arte mostrado en la siguiente sección), hasta donde sabemos este es el primer trabajo que realiza la simulación de un proceso de interacción “nanobot” – liberación de fármaco – célula de sangre roja – destrucción de tumor; esto a la escala de un capilar y no considerando las nanopartículas como puntos en el sistema (es por esto que es multiescala o “cross scale”), lo que constituye nuestra aportación al área de simulación de procesos biofísicos por computadora y que se espera contribuya de forma significativa al campo de aplicación de la ciencia computacional a la hemodinámica.

---

científico Ludwig Boltzmann (1844 - 1906) y que se basa en considerar los fluidos desde el punto de vista de la mecánica estadística, en donde se obtendrían las propiedades macroscópicas del fluido a partir de ***promediar estadísticamente*** las interacciones microscópicas del mismo. (Succi, 2001) indica que como idea intuitiva esto último se refiere a tomar en un sistema termodinámico una muestra del mismo lo suficientemente pequeña respecto al mismo pero estadísticamente representativa de todo el sistema.

<sup>8</sup> Adicionalmente diversos estudios han demostrado que el MRB es comparable en cuanto a exactitud y eficiencia con los métodos numéricos convencionales más comunes, según (Mazzeo & Coveney, 2008).

<sup>9</sup> Aunque esto solamente es estrictamente aplicable a los modelos de Boltzmann para fluidos de una sola fase debido a que puede haber aplicaciones (v. gr. Fluidos multifase en medios porosos) donde las interacciones en cada nodo dependen de los nodos vecinos, es posible mejorar a pesar de ello la eficiencia de ejecución para aplicaciones de fluidos multifase combinando una adecuada descomposición de dominio con una implementación mejorada del MRB como se hizo en (Pan, Prins, & Miller, 2004). Esto sin contar con las muchas investigaciones que ya se han hecho al respecto para optimizar la paralelización mediante dicho MRB, ver por ejemplo (Axner, et al., 2008) y (Bernaschi, et al., 2009).

Para finalizar la introducción es necesario mencionar que se han hecho también avances significativos en la obtención de datos relevantes del campo de velocidades mediante técnicas de medición no invasivas v. gr. El ultrasonido Doppler (consulte referencia (Gohil, et al., 2011)), sin embargo el mismo autor reconoce que aún falta mucho por hacer para que dichas técnicas entreguen datos secundarios de interés adicionales de manera confiable, como ya lo hacen los métodos numéricos tal como el MRB ya mencionado.

# Capítulo 1.

## 1.1 Planteamiento del Problema y Antecedentes.

Los vasos sanguíneos dentro del cuerpo humano se dividen en cuatro tipos: arterias tales como la aorta o las arterias coronarias que salen del corazón hacia los órganos y que sirven como reserva de presión, las arteriolas que irrigan cada órgano en particular, las venas que son la vía de regreso del los órganos al corazón y que sirven como reserva de sangre, y finalmente los capilares ( $5 \mu\text{m} - 10 \mu\text{m}$ ) que son los que permiten la absorción directa del oxígeno por las células de los tejidos que componen los órganos propiamente dichos y que forman parte de la microcirculación (Lauralee, 2011),

En la Introducción se mencionan algunas características de los capilares que la distinguen de otras redes de flujo sanguíneo, pero también tiene propiedades distintivas muy especiales como por ejemplo que el fluido sanguíneo dentro de cada capilar se comporta como una suspensión multifase de partículas deformables (de manera extrema en el caso de las células de sangre roja sanas) (Doddi & Bagchi, 2009). De hecho esta estructura es uno de los principales objetivos de todo el sistema circulatorio; todas las otras actividades o sistemas se enfocan en asegurar la distribución adecuada de sangre a los capilares (Lauralee, 2011).

El modelo hidrodinámico del flujo sanguíneo a la escala que se va a considerar no puede aproximarse como un fluido Newtoniano, debido principalmente a fenómenos tales como el efecto Fahraeus–Lindqvist<sup>10</sup> el cual

---

<sup>10</sup> (Doddi & Bagchi, 2009) menciona que este efecto se refiere a una reducción de la viscosidad aparente (que es la viscosidad dinámica y que se define como la relación entre el esfuerzo cortante y la tasa de corte, consultar (Aldi et al., 2017)) de la sangre mientras se reduce a su vez el diámetro del vaso sanguíneo. Según el mismo esto puede deberse a la migración lateral de las células de sangre roja (RBCs por sus siglas en inglés)

es causado a su vez por la ya mencionada alta deformabilidad de las células de sangre roja, a la naturaleza multifase de la sangre, entre otras (Doddi & Bagchi, 2009), (Lázaro, Hernández–Machado, & Pagonabarraga. II, 2014) y (Tan, Thomas, & Liu, 2011).

La estructura formada por los vasos sanguíneos capilares será el objeto de nuestro estudio en el cual se analiza con detalle los diversos y complejos mecanismos reológicos, mecánicos y químicos involucrados así como las consecuencias médicas que involucran ya que se espera apoyar los esfuerzos que se hacen por investigar terapias que combatan enfermedades circulatorias, utilizando para ello las tecnologías más avanzadas en este campo tales como el uso de “nanobots” con liberación controlada de alguna droga terapéutica (en nuestro caso paclitaxel) que destruya un tumor que obstruye el libre flujo de las RBCs (o eritrocitos) en algún vaso sanguíneo de este tipo.

Recientemente se han hecho aportes para ampliar el conocimiento de la microcirculación como por ejemplo (Doddi & Bagchi, 2009), que entre otros estudios investigan el ancho de la capa libre de células y la variación en la viscosidad aparente de la sangre y el hematocrito como una función de la sección transversal del vaso sanguíneo; esto para afinar posteriormente un modelo de flujo sanguíneo en microvasos y obtener resultados de perfiles de velocidad congruentes con simulaciones ya comprobadas, o las de (Lázaro, Hernández–Machado, & Pagonabarraga. II, 2014) que investigan que parámetros, por ejemplo el incremento en la velocidad del flujo, dictan la distribución de células a lo largo de microcanales ( $10\ \mu\text{m} - 30\ \mu\text{m}$  de diámetro).

También las de (Shi, Zhang, & Wang, 2014), quienes hacen estudios varios acerca de las interacciones hemodinámicas entre eritrocitos en un capilar (por ejemplo la modificación del flujo microcirculatorio debido a dichas células y viceversa), entre otros estudios de diversos tipos (consultar referencias (Shi, Wang, & Zhang, 2013), (Shen, et al., 2016), (Yin, Thomas, & Zhang, 2013), en otras).

Algo en común que tienen los anteriores aportes es el modelo matemático subyacente en el que se basan: la *ecuación de redes de*

---

hacia el centro del vaso sanguíneo lo que ocasiona una región libre de partículas cerca de las paredes de dicho vaso.

*Boltzmann*<sup>11</sup> (ERB), la cual se ha constituido como un interesante esquema numérico en lo que a paralelización de sistemas complejos se refiere debido a que se trata de un derivado de los autómatas celulares de gases en retículo<sup>12</sup> (LGCAs por sus siglas en inglés) (Wolf–Gladrow, 2000). En este tipo de máquinas abstractas los cálculos numéricos se realizan en unidades discretas de trabajo (celdas o nodos) cuya actualización sólo depende de los nodos locales vecinos, lo que beneficia su implementación en diversas arquitecturas paralelas.

Aunque debido a su simplicidad de programación, su elegante eficiencia computacional, entre otros atributos, la ERB fue seleccionada como el esquema numérico subyacente para la simulación computacional de un subsistema microcirculatorio cuya finalidad terapéutica se detallará más adelante. Es necesario mencionar que también se han realizado aportes utilizando otras técnicas de simulación alternativas a la ERB, como por ejemplo (Fedosov, Dao, Em Karniadakis, & Suresh, 2014) quienes, basados en el método de dinámica de partículas disipativas<sup>13</sup> estudian el comportamiento general de la sangre tanto en estado de buena salud como en estado patológico (que incluyen varios tipos de cáncer, entre otros desórdenes). El mismo método puede capturar detalles realistas de la reología de la sangre así como de sus componentes individuales en el flujo sanguíneo.

Otro método similar es el utilizado por (McWhirter, Noguchi, & Gompper, 2009), el cual combina el método multipartícula a escala mesoscópica dinámica de colisión multipartícula para la simulación del comportamiento hidrodinámico del plasma sanguíneo y un modelo de red triangulada que consta de resortes armónicos para la membrana elástica celular lo que permite modelar células deformables, donde esta última

---

<sup>11</sup> Esquema de cómputo numérico basado en autómatas celulares pero sin las desventajas de éstos y que en cambio cuenta con varias bondades, entre ellas: fácil incorporación de geometrías complejas, simulación de partículas suspendidas, paralelización eficiente, etc. Adicionalmente se menciona que el método se basa en un enfoque numérico a escala mesoscópica, es decir en un nivel intermedio entre el quasicuántico nivel microscópico (átomos, moléculas) y el nivel macroscópico (fenómeno observable o medible) que trata de simplificar la extensa complejidad del primero pero sin perder la física del segundo. Debido a que para obtener la ERB es necesario considerarla previamente dentro del contexto de un MRB determinado utilizaremos ambos términos de forma indistinta en el presente trabajo.

<sup>12</sup> Autómata celular que divide la regla de actualización en colisión y propagación con el fin de mantener simple la misma y así poder modelar más fácilmente procesos físicos tales como por ejemplo flujo de fluidos (Wolf-Gladrow, 2000), en especial para el caso de fenómenos físicos, (O'Connor, 2016) menciona que la ERB es especialmente útil en la simulación de sistemas microcirculatorios complejos como la aquí presentada.

<sup>13</sup> El cual es un derivado de los métodos de dinámica molecular, que aplica la mecánica clásica al movimiento de partículas microscópicas.

característica, según mencionan, juega un papel importante en la liberación del oxígeno transportado por los eritrocitos. El mismo autor menciona que la deformabilidad de estas células se ve reducida por enfermedades tales como la diabetes mellitus y la anemia falciforme.

También se han propuesto modelos acoplando métodos continuos con algún otro método de dinámica molecular, por ejemplo el realizado por (Moreno, Vignal, Li, & Calo, 2013), en el cual se utiliza el método del elemento finito para simular las escalas macroscópicas, y un método basado en el correspondiente de dinámica de partículas disipativas para simular las características de escala más fina permitiendo mediante esto último simular de manera consistente la conocida fluctuación térmica en el nivel mesoscópico (que en el caso de los eritrocitos es un factor que afecta el comportamiento de la membrana), lo que a su vez permite comprender mejor el funcionamiento fisiológico de dicha célula.

Ahora bien, como ya se mencionó en la introducción, los métodos basados en la ERB tienen la ventaja de la eficiencia de ejecución sobre los modelos continuos sobretodo en lo que a la paralelización se refiere. Adicionalmente, según (Fedosov, Noguchi & Gompper, 2014) la ERB tiene la ventaja sobre los modelos continuos de incorporar geometrías complejas y características de escala mesoscópica (por ejemplo la fluctuación térmica) sin mayor dificultad, en cambio en los modelos continuos puede ser nada trivial la inclusión de dichas características.

Otras razones por las que se seleccionó la ERB como esquema numérico para el desarrollo de nuestro software para simulación es que debido a la forma en que está constituido dicho esquema numérico se asegura la conservación de masa y momento lo que facilita en gran medida la modelación de geometrías complejas. También, el campo de presión y el esfuerzo cortante está disponible en cualquier punto de la red discretizada del modelo en cualquier instante de tiempo (Bernaschi, Fatica, Melchionna, Succi, & Kaxiras, 2010). Y finalmente, debido al hecho de que la ERB es resuelta en un estencil bien establecido y definido previamente lleva a una correcta evaluación de la hidrodinámica del fluido del sistema (Fedosov, Noguchi & Gompper, 2014).

Sin embargo también es necesario mencionar que el último autor menciona que si el número de solutos (para nuestro caso esto se refiere a partículas suspendidas) se incrementa, el performance puede verse afectado.

Sin embargo se han creado a su vez técnicas de paralelización que logran hacer, por ejemplo que en una simulación donde múltiples partículas se encuentran suspendidas en un fluido, que las interacciones fluido – partícula sean locales (Bernaschi et al., 2009). Logrando con ello mejorar substancialmente el performance global de ejecución. Otro logro en este sentido puede consultarse en (Clausen, Reasor Jr. & Aidun, 2010)<sup>14</sup>.

## 1.2 Nanomedicina.

La nanotecnología ha irrumpido en el escenario de los avances tecnológicos como una opción prometedora para un sinnúmero de nuevas aplicaciones en los más diversos campos, desde la industria de la energía solar hasta la optoelectrónica pasando por un verdadero auge para las aplicaciones médicas<sup>15</sup> (Sahoo, Parveen, & Panda, 2007), tales como la separación y ordenación de células para diagnóstico clínico así como hipertermia de fluido magnético para tratamiento de cáncer.

La nanotecnología aplicada a la medicina (nanomedicina) ayudará a combatir algunas de las enfermedades de más alta mortalidad en la actualidad. Por ejemplo la aterosclerosis que es el primer lugar en decesos en los EEUU con casi 550,000 muertes cada año por alguna afección cardiaca, lo que representa 1 de cada 5 muertes en ese país, según cifras de la Asociación Americana del Corazón (AHA por sus siglas en inglés). También, según información proporcionada por el mismo organismo, el impacto social seguirá en aumento (Hossain, Hossainy, Bazilevs, Calo, & Hughes, 2012).

Adicionalmente es necesario mencionar que la utilización de nanopartículas para la liberación de drogas terapéuticas tiene varias ventajas, en especial para las partículas poliméricas<sup>16</sup>. Entre ellas se encuentran: mejor solubilidad del fármaco, localización precisa, liberación controlada del

---

<sup>14</sup> Aunque en este ejemplo la simulación es de RBCs deformables fluyendo en una vena sanguínea, ilustra perfectamente como el MRB se puede adaptar con otros métodos para así incrementar el performance sin sacrificar en lo más mínimo los detalles de la simulación ni la física del sistema, para un compendio de varios de estos métodos puede consultarse en (Freund, 2014).

<sup>15</sup> Probablemente la siguiente revolución industrial según (Cinelli, Coles, Sadik, Karn, & Kirwan, 2016).

<sup>16</sup> Con un amplio campo de aplicación (Prasad Rao & Geckeler, 2011), (Banik, Fattahi, & Brown, 2016) menciona que por sí solas representan un nuevo paradigma en la nanomedicina, en especial para combatir el cáncer, desórdenes neurodegenerativos y enfermedades cardiovasculares.

fármaco, protección de la carga, etc. (Felicetti, Femminella, Reali, & Liò, 2016), (Sahoo, Parveen, & Panda, 2007), (Narayanan, et al., 2010).

Existe además una amplia literatura en el tema que demuestra que es una interesante área de investigación interdisciplinaria que está atrayendo la atención de gobiernos e investigadores (Jo, Park, Eunkyung Kim, & Lee, 2016) (se recomienda consultar la investigación de la estructura conceptual actualizada en este campo, la cual se muestra en el artículo citado, así como el rediseño del inventario de productos ofrecido a los consumidores en todo el mundo realizado en (Vance, et al., 2015)).

Ya se han realizado algunos estudios utilizando simulación computacional que hacen uso de la ERB en el flujo sanguíneo con la intención de mejorar la eficiencia terapéutica de nanopartículas suspendidas en el mismo. Por ejemplo (Buxton, 2008) estudió como la deformación de una nanopartícula que transporta un agente terapéutico en el tratamiento del cáncer sujeta a esfuerzo cortante causado por el flujo sanguíneo afecta la capacidad de la misma para liberar el fármaco transportado. Más recientemente (Gekle, 2016) muestra cómo se puede acelerar la migración de partículas deformables hacia la capa libre de células cuando éstas van contracorriente en un vaso sanguíneo del tamaño aproximado de una arteriola mediante la aplicación de una fuerza externa.

Existen importantes diferencias en las características, propiedades y estudios que se pueden llevar a cabo a la escala de los vasos sanguíneos capilares que abordamos en nuestro trabajo respecto a lo anterior. Una propiedad es que las células (al menos las RBCs) tienden a migrar siempre hacia el centro del vaso capilar efectuándose, como ya se mencionó, el efecto de Fahraeus–Lindqvist que produce un aumento en la viscosidad aparente de la sangre. Una característica es la gran deformación que sufren las RBCs debido al tamaño del capilar que hace que sólo pasen en fila y que para cierta distancia entre las células se afecten unas a otras, v. gr. En la velocidad de translación. Un estudio que se puede llevar a cabo es el movimiento aislado de una sola RBC cuando esta a una distancia suficiente de cualquier otra célula en la fila de células de forma que su deformación no se vea afectada por otras células en dicha fila, entre otras propiedades, características y estudios a llevar a cabo (Shi, X., Zhang, S., & Wang, S.–l., 2014), (Shi, X., Wang, S., & Zhang, S., 2013), (Posrikidis, 2005).

Están también los estudios que se han hecho mediante otros modelos. Por ejemplo (Djohari & Dormidontova, 2009) hace uso del método de partículas disipativas para estudiar el cambio en la cinética de nanopartículas dirigidas hacia una superficie celular que contiene una alta concentración de receptores del fármaco a aplicar.

Hasta el momento el campo de investigación sigue abierto, al menos en lo que a simulación numérica de transporte y liberación de drogas terapéuticas se refiere<sup>17</sup> y es sin duda uno de las más prometedoras áreas de aplicaciones a explorar y explotar en la ciencia de la nanomedicina (Sahoo, Parveen, & Panda, 2007).

### **1.3 Paralelización.**

Aunque nuestro trabajo no se enfoca en la mejora de las técnicas ya existentes (ver por ejemplo (Ahmadian, Firoozbakhsh, & Hasanian, 2012)) para la simulación de flujo sanguíneo en capilares ciertamente si trata de eficientar la ejecución de un sistema computacional complejo mediante técnicas de paralelización masiva. Primeramente porque en si la simulación del flujo de fluidos es computacionalmente demandante (Xiong, et al., 2012), y enseguida por que el MRB ya se ha probado con éxito en sistemas de cómputo de alto rendimiento (HPC) utilizados para investigación científica (ver por ejemplo (Clausen, Reasor, & Aidun, 2010), (Janoschek, Toschi, & Harting, 2010) y otros).

(Clausen, Reasor, & Aidun, 2010) menciona al menos dos estudios para incrementar la eficiencia de las implementaciones mediante el MRB para diversas aplicaciones (Bernaschi, et al., 2009), (Zeiser, Hager, & Wellein, 2009). Y que el método es altamente escalable aún para simulaciones de tipo turbulento (Vahala, et al., 2008).

Además, el MRB se adapta de manera casi natural a la mayoría de las arquitecturas de cómputo de alto rendimiento ya que el cómputo puede hacerse de manera local. Esto al costo de tener en la memoria del servidor una copia de la matriz entera de las distribuciones correspondientes a los nodos del fluido (Röhm, 2011).

---

<sup>17</sup> (Koumoutsakos, Pivkin, & Milde, 2013) menciona que después de 15 años de investigación sólo se han realizado un par de formulaciones terapéuticas para la liberación de nanopartículas a tumores cancerosos.

Las Unidades de Procesamiento Gráfico (GPUs por sus siglas en inglés) se han mostrado como interesantes alternativas para procesar la ingente cantidad de procesamiento numérico de muchas aplicaciones en la actualidad tales como el procesamiento de señales, simulación de fenómenos físicos, finanzas computacionales, biología computacional, renderización en tercera dimensión, etc., debido a la gran cantidad de operaciones de punto flotante que pueden realizar respecto a un CPU multinúcleo<sup>18</sup>, así como también al ancho de banda al que se puede comunicar dicho GPU con su memoria principal respecto al ancho de banda de la comunicación similar del CPU.

Es en parte debido a lo anterior y a que ya se han reportado resultados satisfactorios en arquitecturas de paralelización combinadas, por ejemplo con el ya conocido estándar Interfaz de Paso de Mensajes (MPI por sus siglas en inglés), utilizando estos dispositivos para simulaciones de sistemas biofluídicos (por ejemplo consultar (Bernaschi, Fatica, Melchionna, Succi, & Kaxiras, 2010)) que se decidió explorar el paradigma de paralelización GPU–MPI en nuestra simulación computacional particular.

Con base en los trabajos anteriormente mencionados los cuales nos dan una idea del avance que se ha logrado en lo que a simulación computacional del flujo sanguíneo en general se refiere<sup>19</sup> así como de sus aplicaciones en el campo de la nanomedicina, es que nuestra intención es precisamente seguir cooperando en el desarrollo de técnicas terapéuticas innovadoras en enfermedades graves proporcionando una herramienta de investigación para el estudio de las interacciones microhidrodinámicas que ocurren a la escala de una simple nanopartícula, lo cual hasta donde sabemos no existen estudios similares realizados a la fecha.

#### **1.4 La Ecuación de Redes de Boltzmann.**

Inicialmente considerada como una herramienta para calcular la viscosidad de los LGCAs (Wolfram, 1986) y (Frisch, She, & Sulem, 1987), la

---

<sup>18</sup> Unidad Central de Proceso (CPU por sus siglas en inglés) con dos o más Subunidades Centrales de Proceso independientes cuya finalidad es optimizar la ejecución de un programa secuencial haciendo uso del paralelismo intrínseco en dicho CPU (Behrooz, 1999).

<sup>19</sup> Para aportaciones en la simulación de flujo sanguíneo no capilar véase (Clausen, Reasor, & Aidun, 2010), (Janoschek, Toschi, & Harting, 2010), (MacMECCAN, Clausen, Neitzel, & Aidun, 2009), entre varios otros autores.

ERB fue propuesta por primera vez como esquema numérico por (McNamara & Zanetti, 1988) mostrando ya desde entonces sus bondades en lo que a eficiencia computacional se refiere, además de eliminar el ruido estadístico presente en simulaciones similares utilizando LGCA's (Wolf-Gladrow, 2000).

La ERB se discretiza a partir de la ecuación de Boltzmann la cual describe el movimiento de las partículas presentes en un fluido, el cual puede componerse de uno o más fluidos y/o de diversas fases de los mismos. En realidad resulta relativamente fácil obtener la ecuación de Boltzmann si consideramos los supuestos que asumió el célebre científico Austriaco en 1872 para desarrollar una ecuación que logró unir dos mundos hasta entonces desconectados: la mecánica estadística y la termodinámica (Succi, 2001).

Según (Succi, 2001) se puede considerar un modelo físico como un sistema formado por  $N$  partículas (o moléculas) puntuales todas de masa  $m$ , con una trayectoria libre media (longitud media entre cada colisión) lo suficientemente grande como para considerar que dichas colisiones son raras y que cuando lo hacen es sólo entre dos partículas a lo mucho y que no interactúan entre sí de otra manera que no sea mediante fuerzas intermoleculares de corto alcance (la fuerza intermolecular  $\ll$  trayectoria libre media); por lo que se podrían considerar como colisiones elásticas entre esferas duras de masa  $m$ . Esta aproximación es compatible con la idea original de Boltzmann de desarrollar su ecuación para un gas muy diluido pero que a partir de ésta es posible aplicarla para fluidos mucho más densos e inclusive con comportamientos de tipo turbulento.

(Succi, 2001) menciona que para realizar la descripción exacta del movimiento de cada partícula del sistema anterior esto puede hacerse mediante las ecuaciones que describen su movimiento,

$$\begin{aligned} \frac{d\mathbf{x}_i}{dt} &= \mathbf{v}, \\ \frac{d\mathbf{v}_i}{dt} &= \mathbf{a} \end{aligned} \quad \text{Para } i = 1, 2, 3, \dots, N \quad (1.1a)$$

o bien mediante las ecuaciones basadas en el Hamiltoniano del sistema,

$$\frac{\partial H(\mathbf{x}_i, \mathbf{p}_i)}{\partial \mathbf{p}_i} = \frac{d\mathbf{x}_i}{dt} = \mathbf{v},$$

$$\frac{\partial H(\mathbf{x}_i, \mathbf{p}_i)}{\partial \mathbf{x}_i} = \frac{d\mathbf{p}_i}{dt}$$
(1.1b)

pero que aún cuando se pudiera crear un sistema computacional en donde se tuviese un conocimiento exacto de todas las posiciones y velocidades de cada una de las  $N$  partículas en el sistema (eso sin contar con las simplificaciones que habría que tomar en cuenta, como por ejemplo anular el efecto de todas las demás partículas en el universo entero (Cercignani, 1988), y también que una incertidumbre mínima en las condiciones iniciales del sistema crece exponencialmente en el tiempo) resolviendo para ello el sistema de ecuaciones (1.1) esto sería de hecho inútil, pues la información que realmente interesa son las variables termodinámicas (presión, temperatura, etc.).

Esta última información puede obtenerse por medio de promedios estadísticos. La idea intuitiva de éste concepto está a su vez relacionada con una de las ideas centrales de la mecánica estadística que es la del ensamble de elementos que componen un sistema muy grande que por ejemplo, para un gas con las características antes mencionadas, alcanza hasta las  $6.02 \times 10^{23}$  moléculas/mol (número de Avogadro).

Si se supone que el sistema mencionado se encuentra bajo ciertas condiciones determinadas como son: el número de elementos,  $N$ , los cuales están contenidos en un volumen dado,  $V$ , y sujetos a una determinada energía,  $E$ , (macroestado del sistema) definimos el ensamble como un conjunto de copias del sistema cada uno siguiendo su propio comportamiento microscópico independiente pero con todos los elementos del ensamble bajo el mismo macroestado.

Para la determinación de los promedios estadísticos del sistema primeramente se asumen las siguientes condiciones:

1. Para un determinado macroestado del sistema todos los elementos del ensamble pueden pasar por todos los posibles estados del mismo (llamémosles microestados, que para nuestro caso serían las posibles posiciones y momentos de cada una de las  $N$  partículas del sistema).

2. Asumir que el comportamiento promedio de cada uno de los elementos del ensamble es el mismo para un determinado instante de tiempo que el comportamiento promedio del sistema para un cierto lapso de tiempo<sup>20</sup>.

Entonces, si se desea estudiar la evolución en el tiempo de un elemento del ensamble para un macroestado dado a partir de una condición inicial, digamos  $P_0$  (condiciones iniciales de posición y momento de todas las partículas del sistema,  $\mathbf{x}_{10}, \mathbf{x}_{20}, \dots, \mathbf{x}_{N0}, \mathbf{p}_{10}, \mathbf{p}_{20}, \dots, \mathbf{p}_{N0}$ , para partícula 1, 2, ...,  $N$ ), observaríamos que sigue una determinada trayectoria de acuerdo a las ecs. (1.1) en un sistema  $N$ -coordinado compuesto por las componentes cartesianas de las posiciones y los momentos de todas las partículas del sistema lo que llamaremos espacio fase.

Si se considera ahora un determinado intervalo de puntos iniciales (uno para cada elemento del ensamble) a partir de los cuales parte la evolución de dicho macroestado general cada uno siguiendo su propia trayectoria dentro del espacio fase de acuerdo con las ecs. (1.1) pero limitada por el volumen  $V$  en el caso de la posición y por la energía  $E$  en el caso de la posición y el momento por medio del Hamiltoniano del sistema,  $H(\mathbf{x}_i, \mathbf{p}_i)$ , se formaría una superficie dentro del espacio fase como sigue,

$$H(\mathbf{x}_i, \mathbf{p}_i) = E \tag{1.2}$$

con  $i =$  partícula 1, 2, ...,  $N$  (Pathria & Beale, 2011). Se observaría entonces que todos los puntos o elementos del ensamble se tenderían a agrupar dentro de la región del espacio fase limitada por la ec. (1.2) formando una especie de “enjambre” de dichos puntos más o menos densa dependiendo del comportamiento promedio del conjunto formado por el sistema original y los elementos del ensamble<sup>21</sup>.

---

<sup>20</sup> (Pathria & Beale, 2011) Menciona que esto es razonable ya que si se considera que durante un cierto lapso de tiempo el sistema original puede pasar por cualquiera de todo el vasto conjunto de microestados del sistema, entonces lo que se observaría durante dicho lapso de tiempo es el comportamiento promedio del sistema, que se esperaría sería prácticamente el mismo para cualquier otro elemento del ensamble o copia de dicho sistema.

<sup>21</sup> Es necesario mencionar que lo anterior sólo se cumple para el ensamble microcanónico en el cual los parámetros del macroestado del sistema son las propiedades replicadas en cada elemento del mismo, sin embargo la forma final de la ecuación de Boltzmann parte de la asunción de un ensamble canónico en el cual el número total de partículas, el volumen y la temperatura son los parámetros termodinámicos independientes (Laurendeau, 2005). El texto de la presente sección sólo pretende dar nociones que ayuden al lector a situarse en el contexto de la aplicación de la ecuación de Boltzmann a la simulación del flujo de fluidos, las particularidades del desarrollo físico y matemático subyacente esta fuera del alcance de este trabajo.

De acuerdo con lo anterior surge entonces la idea de escribir una ecuación que describa la evolución del comportamiento del sistema en términos de una función de distribución probabilística de  $N$  cuerpos<sup>22</sup>,  $P(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N, \mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N, t)$ , la cual contiene la información estadística de las  $N$  partículas en cada instante de tiempo. Nótese entonces que la cantidad  $P(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N, \mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N, t) \Delta^N x \Delta^N p$  (donde  $\Delta^N x = \Delta x_1 \Delta x_2 \dots \Delta x_N$ ,  $\Delta^N p = \Delta p_1 \Delta p_2 \dots \Delta p_N$ ) dará una idea de como esta distribuida la probabilidad de los puntos del ensamble (limitados por la región de la ec. (1.2)) en el espacio fase<sup>23</sup>, en otras palabras cual es el conjunto de microestados más probables al interior de dicha región.

Ya con esto último según (Cercignani, 1988) se podría calcular, al menos en teoría, el promedio de una determinada cantidad física, digamos  $T$ , de la siguiente forma:

$$T_{prom} = \int_R P(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N, \mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N) T d^N x d^N p \quad (1.3)$$

Si se define ahora una función de distribución probabilística,  $f_1(\mathbf{x}, \mathbf{p}, t)$  de un cuerpo, que indica la probabilidad de que una molécula dada se encuentre alrededor de la posición  $\mathbf{x}$  y el momento  $\mathbf{p}$  en el tiempo  $t$ , entonces, según (Succi, 2001) se puede escribir,

$$f_1(\mathbf{x}, \mathbf{p}, t) = \frac{n}{\Delta^N x \Delta^N p} \quad (1.4)$$

donde  $n(\mathbf{x}, \mathbf{p}, t)$  es el número probable de moléculas cuya posición  $\mathbf{x}$  y momento  $\mathbf{p}$  se encuentran dentro de los intervalos  $\mathbf{x} + \Delta \mathbf{x}$  y  $\mathbf{p} + \Delta \mathbf{p}$  en el tiempo  $t$ . Nótese que la función  $f_1$  también es la función de distribución de la

<sup>22</sup> Es decir la probabilidad de que la posición y momento de las partículas 1, 2, ...,  $N$  se encuentren alrededor de  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$  y  $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N$ , respectivamente en un tiempo  $t$  dado con los subíndices 1, 2, ...,  $N$  correspondiendo a las partículas de igual numeral.

<sup>23</sup> Aunque estrictamente hablando dicho espacio vectorial está formado por las componentes cartesianas de la posición y el momento de las  $N$  partículas de nuestro sistema,  $(x_{1x}, x_{2x}, \dots, x_{Nx}, x_{1y}, x_{2y}, \dots, x_{Ny}, x_{1z}, x_{2z}, \dots, x_{Nz}, q_{1x}, q_{2x}, \dots, q_{Nx}, q_{1y}, q_{2y}, \dots, q_{Ny}, q_{1z}, q_{2z}, \dots, q_{Nz})$ , la forma final a obtener de la ecuación de Boltzmann sólo está dada en términos de la función de distribución de un solo cuerpo, por lo que el espacio fase sólo estará formado por una dupla de las componentes cartesianas de la velocidad y el momento. Esto es posible aún considerando que en dicha ecuación se incluye un término de colisión que involucra interacciones entre dos partículas ya que en base a las simplificaciones asumidas previamente para las interacciones interpartícula y las simplificaciones varias que se han propuesto para emplear de manera práctica la ecuación de Boltzmann (por ejemplo la simplificación BGK, la cual se describe más adelante en la presente sección), se puede describir la evolución del sistema mediante la ecuación de Boltzmann en términos de la función de distribución de un cuerpo.

densidad de partículas que se encuentran dentro del volumen  $\Delta^N x \Delta^N p$  en el tiempo  $t$ .

Si se multiplica  $m$  por la ecuación anterior y definimos  $f = mf_1$  se tiene,

$$f(\mathbf{x}, \mathbf{p}, t) = \frac{mn}{\Delta^N x \Delta^N p} \quad (1.5)$$

donde la función  $f$  es la distribución de probabilidad de densidad (de partículas) por unidad de volumen de espacio fase para el tiempo  $t$ .

Ahora bien, para el caso 3D y suponiendo que no existen interacciones entre las partículas para la función anterior en un tiempo  $t + \Delta t$ , entonces el número probable de partículas que se encuentran en el elemento de volumen  $\Delta^3 x \Delta^3 p$  será,

$$\frac{1}{m} f(\mathbf{x} + \Delta \mathbf{x}, \mathbf{p} + \Delta \mathbf{p}, t + \Delta t) \Delta^3 x \Delta^3 p = \frac{1}{m} f(\mathbf{x}, \mathbf{p}, t) \Delta^3 x \Delta^3 p \quad (1.6)$$

lo anterior es posible afirmarlo dado que, aun cuando se aplique una determinada fuerza externa  $\mathbf{F}$  al conjunto de partículas, el área de las proyecciones de las superficies que forman el nuevo volumen ocupado por dichas partículas no excede las correspondientes  $\Delta x_x \Delta p_x$ ,  $\Delta x_y \Delta p_y$  y  $\Delta x_z \Delta p_z$  (Román Román, 2007), es decir las partículas seguirán ocupando el mismo espacio de volumen  $\Delta^3 x \Delta^3 p$  en ambos tiempos. De acuerdo con esto si eliminamos de ambos lados de la ecuación dicho término tenemos entonces,

$$\frac{1}{m} f(\mathbf{x} + \Delta \mathbf{x}, \mathbf{p} + \Delta \mathbf{p}, t + \Delta t) = \frac{1}{m} f(\mathbf{x}, \mathbf{p}, t) \quad (1.7)$$

Si se considera ahora que las partículas pueden interactuar entre sí mediante colisiones binarias elásticas entre ellas, se debe adicionar al lado derecho de la ec. (1.7) la diferencia entre la cantidad de las partículas debido a colisiones que entran al elemento de volumen  $\Delta^3 x \Delta^3 p$  menos la cantidad de partículas debido a colisiones que salen del mismo. Si por conveniencia se selecciona la tasa temporal de dicha diferencia  $C_1$  se tiene,

$$\frac{1}{m} f(\mathbf{x} + \Delta \mathbf{x}, \mathbf{p} + \Delta \mathbf{p}, t + \Delta t) = \frac{1}{m} f(\mathbf{x}, \mathbf{p}, t) + C_1 \Delta t \quad (1.8)$$

Asúmase ahora que nuestra función  $f$  es continua en todo el espacio fase, si eliminamos elementos comunes en la ec. (1.8), dividimos por  $\Delta t$ , reacomodamos y tomamos el límite cuando  $\Delta t \rightarrow 0$  tenemos,

$$\lim_{\Delta t \rightarrow 0} \frac{f(\mathbf{x} + \Delta \mathbf{x}, \mathbf{p} + \Delta \mathbf{p}, t + \Delta t) - f(\mathbf{x}, \mathbf{p}, t)}{\Delta t} = \frac{df(\mathbf{x}, \mathbf{p}, t)}{dt} = m \lim_{\Delta t \rightarrow 0} C_1 \quad (1.9)$$

$$= mC_1 = C$$

Con lo que se obtiene,

$$\left( \frac{d\mathbf{x}}{dt} \cdot \frac{\partial}{\partial \mathbf{x}} + \frac{d\mathbf{p}}{dt} \cdot \frac{\partial}{\partial \mathbf{p}} + \frac{\partial}{\partial t} \right) f(\mathbf{x}, \mathbf{p}, t) = C \quad (1.10)$$

Identificando  $d\mathbf{x}/dt = \mathbf{v}$  y  $d\mathbf{p}/dt = m\mathbf{a} = \mathbf{F}$  y sustituyendo en la ecuación anterior, se tiene:

$$\left( \mathbf{p} \cdot \frac{\partial}{\partial \mathbf{x}} + \mathbf{F} \cdot \frac{\partial}{\partial \mathbf{p}} + \frac{\partial}{\partial t} \right) f(\mathbf{x}, \mathbf{p}, t) = C \quad (1.11)$$

con el término de colisión aún por determinar. Esta sería la *ecuación de Boltzmann de un cuerpo*. Nótese que el término de colisión involucra interacciones entre dos partículas, lo que implica que se deban utilizar funciones de distribución de dos cuerpos. Sin embargo, como ya se mencionó en una nota anterior, la ec. (1.11) es posible expresarla en términos de una función de distribución de un cuerpo asumiendo las simplificaciones para las interacciones interpartícula mencionadas al principio de la presente sección y las varias simplificaciones que se han propuesto para tratar el complicado término integro-diferencial de colisión, el cual se describe a continuación.

Nótese que el lado izquierdo de la ec. (1.11) describe la trayectoria de las partículas descritas como una función de distribución probabilística a lo largo de las líneas de flujo del fluido en tanto que el lado derecho representa las interacciones interpartícula que desvían a las partículas de las líneas de flujo (Succi, 2001).

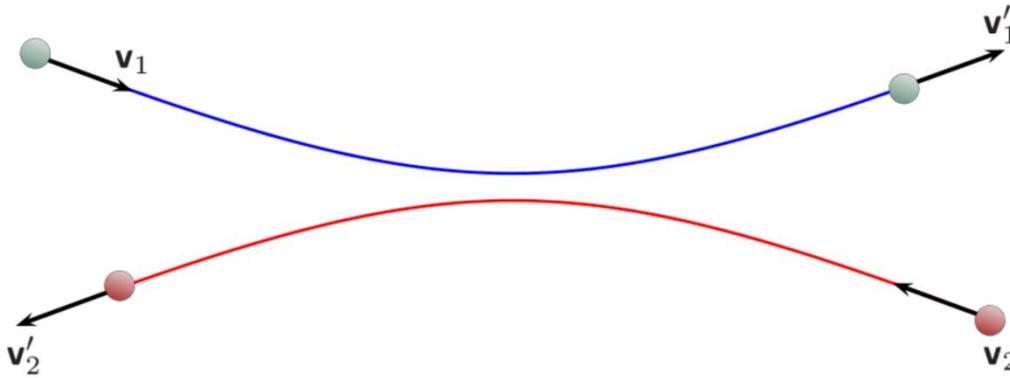


Figura 1. Colisión directa de dos partículas con velocidades iniciales  $v_1$  y  $v_2$ . Impreso con permiso de la Universidad Nacional Autónoma de México.

El término de colisión  $C$  expresa el hecho de que las interacciones intermoleculares contribuyen a la evolución del sistema tanto como el movimiento mecánico del mismo. Para el caso de sistemas en equilibrio termodinámico (Succi, 2001) hace notar que debido a que el término de colisión se puede considerar como independiente del tiempo y del espacio es posible que exista un cierto equilibrio termodinámico local mientras temporal y espacialmente haya variaciones de las variables hidrodinámicas velocidad del flujo y temperatura del sistema.

Para ver mejor esto último considérese la colisión entre dos partículas 1 y 2 dirigiéndose una hacia la otra con momentos  $\mathbf{p}_1$  y  $\mathbf{p}_2$  respectivamente, dicha colisión sucede en un determinado elemento de volumen de espacio fase  $\Delta^3x\Delta^3p$  y hace que ambas partículas abandonen dicho elemento de volumen (colisión directa).  $\mathbf{p}'_1$  y  $\mathbf{p}'_2$  son los correspondientes momentos de ambas partículas después de la colisión, lo anterior se ilustra en la Figura 1. (Román Román, 2007) indica que en el caso de las partículas que entran al elemento de volumen  $\Delta^3x\Delta^3p$  debido a una colisión restitutiva (es decir en este caso las partículas tienen como momentos iniciales  $\mathbf{p}'_1$  y  $\mathbf{p}'_2$ , ver Figura 2) los momentos de las partículas 1 y 2 después de la colisión son equivalentes a los momentos iniciales de las partículas para el caso de la colisión directa, es decir los momentos iniciales serían  $\mathbf{p}'_1$  y  $\mathbf{p}'_2$ , en tanto que los momentos finales serían  $\mathbf{p}_1$  y  $\mathbf{p}_2$ .

Este resultado es importante porque una manera de calcular el término de colisión es considerándose como un haz de partículas que incide en un blanco fijo genera un ángulo sólido  $\Omega$  formado por el ángulo de dispersión  $\theta$  y

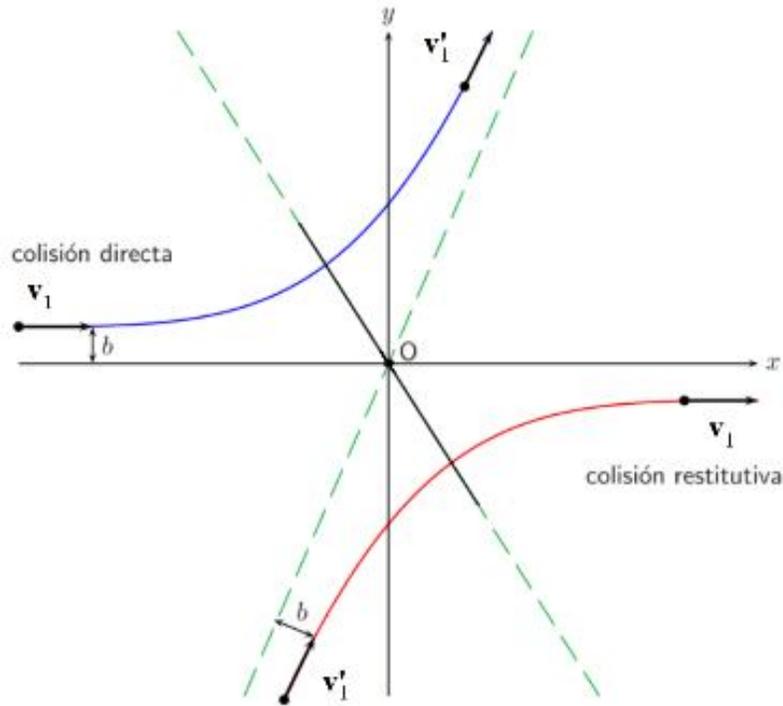


Figura 2. Relación entre la colisión directa y la colisión restitutiva. Impreso con permiso de la Universidad Nacional Autónoma de México.

el ángulo azimutal  $\phi$  de la manera como se muestra en la Figura 3,  $b$  es un parámetro del plano donde ocurre la colisión. Dicho ángulo sólido depende de la sección eficaz<sup>24</sup> (y viceversa) el cual es perpendicular al haz incidente como se muestra en la misma Figura.

Entonces, la tasa temporal de partículas que salen del elemento de volumen  $\Delta^3x\Delta^3p$  durante del intervalo de tiempo  $dt$  debido a las colisiones directas sería, según el mismo autor:

$$Pérdida\ de\ partículas/u.\ de\ tiempo = \int d^3 p_2 \int f_{12}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{p}_1, \mathbf{p}_2, t) |\mathbf{p}_2 - \mathbf{p}_1| \sigma(\Omega) d\Omega$$

para la anterior fórmula se consideran las posibles colisiones del haz de partículas de momento  $\mathbf{p}_2$  contra la partícula 1 en un intervalo de tiempo  $dt$  que inciden a través de la sección eficaz  $\sigma(\Omega)$  y son dispersadas a través de la sección de ángulo sólido  $d\Omega$ . Nótese como se emplea la función de

<sup>24</sup> Es un área imaginaria colocada transversalmente al haz de partículas incidentes y de forma tal que cuando es atravesada por dicho haz la cantidad de partículas que son dispersadas en  $\Omega$  por el blanco es la misma que la cantidad de las que atravesaron el área [W3].

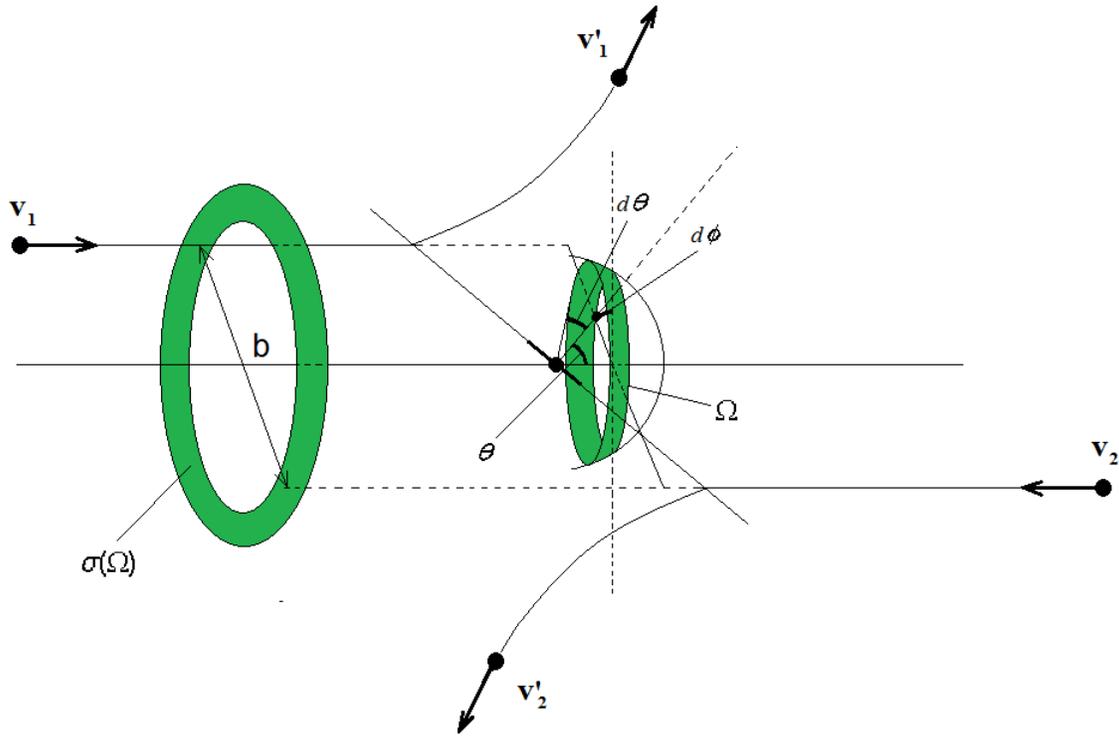


Figura 3. Formación de ángulo sólido  $\Omega$  y sección eficaz  $\sigma(\Omega)$  correspondiente debido a colisión entre dos cuerpos.

distribución de dos cuerpos  $f_{12}$  para las colisiones binarias que ocurren en el elemento de volumen  $\Delta^3 x \Delta^3 p$ .

Asimismo para obtener el término que considera las partículas que entran al elemento de volumen  $\Delta^3 x \Delta^3 p$  con momentos iniciales  $\mathbf{p}'_1$  y  $\mathbf{p}'_2$ , esto lo podemos evaluar considerando que las partículas que son dispersadas mediante colisiones restitutivas lo hacen a través de la misma sección de ángulo sólido del caso anterior, la sección eficaz de dispersión de las colisiones restitutivas  $\sigma_{rest}(\Omega_{rest}) = \sigma_{rest}(\Omega) = \sigma(\Omega)$  es igual a la sección eficaz de las colisiones directas, asimismo es posible demostrar que  $|\mathbf{p}'_2 - \mathbf{p}'_1| = |\mathbf{p}_2 - \mathbf{p}_1|$  y dado que  $d^3 p_2 = d^3 p'_2$  podemos escribir el término de la tasa temporal de la ganancia de las partículas que entran al elemento de volumen  $\Delta^3 x \Delta^3 p$  durante el intervalo de tiempo  $dt$  de manera similar a como escribimos el término de pérdida como:

$$\text{Ganancia de partículas/u. de tpo.} = \int d^3 p_2 \int f_{1'2'}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{p}'_1, \mathbf{p}'_2, t) |\mathbf{p}_2 - \mathbf{p}_1| \sigma(\Omega) d\Omega$$

Boltzmann propuso que  $f_{12} = f_1 f_2$ , a la cual se le conoce como hipótesis de caos molecular o stosszahlansatz, la cual no se puede deducir de las leyes de la Mecánica Clásica y establece que no hay interacción entre las partículas antes de la colisión<sup>25</sup>. Luego entonces sustituyendo el término de colisión  $C = (\text{Ganancia} - \text{Pérdida de partículas}) / (\Delta t = dt)$  en la ec. (1.11) tenemos,

$$\left(\mathbf{p} \cdot \frac{\partial}{\partial \mathbf{x}} + \mathbf{F} \cdot \frac{\partial}{\partial \mathbf{p}} + \frac{\partial}{\partial t}\right) f(\mathbf{x}, \mathbf{p}, t) = \int d^3 p_2 \int (f_1 f_2' - f_1' f_2) |\mathbf{p}_2 - \mathbf{p}_1| \sigma(\Omega) d\Omega \quad (1.12)$$

La ecuación anterior es el punto de partida para los diferentes modelos de Boltzmann<sup>26</sup> modernos que existen actualmente. (Wolf–Gladrow, 2000) menciona que los antecesores de estos últimos estaban plagados de varios inconvenientes entre ellos, un término de advección no isotrópico, violación de la invariancia Galileana, que la presión depende explícitamente de la velocidad, etc. Uno de los factores que contribuyen a lograr la estabilidad de los modelos de Boltzmann es quizá la simplificación que se hace del complicado operador de colisión. De entre otras simplificaciones sin duda la más conocida es la propuesta por (Bhatnagar, Gross, & Krook, 1954), que ha resultado ser muy útil y eficiente para aplicar la ERB en la resolución de una amplia gama de problemas de transporte en flujo de fluidos, incluso para hallar en algunos casos soluciones analíticas a los mismos.

Para entender la simplificación BGK anterior es necesario comprender qué es la función de equilibrio, para lo cual se comienza definiendo el equilibrio termodinámico local de nuestro sistema como el estado alrededor de un punto del espacio fase en el cual las ganancias y pérdidas de partículas están perfectamente balanceadas de modo que no hay un flujo neto de masa alrededor de dicho punto (nótese que esto es a nivel microscópico, en tanto que a nivel macroscópico, es decir medible, podría no haber dicho equilibrio, adicionalmente debe existir también un equilibrio de transferencia de calor, momento, etc., es decir el equilibrio debe ser total), ahora bien si el término de colisión es nulo, de la ec. (1.12) se sigue que  $f_1 f_2' - f_1' f_2 = 0$ . Se puede demostrar a partir de esto último que el logaritmo natural de la función de distribución es una invariante de colisión aditiva y por lo tanto debe ser

<sup>25</sup> O que la probabilidad de que la partícula 1 se encuentre alrededor de la posición  $\mathbf{x}_1$  y el momento  $\mathbf{p}_1$  es independiente de que la partícula 2 se encuentre alrededor de la posición  $\mathbf{x}_2$  y momento  $\mathbf{p}_2$ .

<sup>26</sup> Un modelo de Boltzmann incluye: un tipo de lattice (cuadrada, hexagonal, etc.), una determinada función de equilibrio (que depende del número de términos seleccionados en la expansión de la serie de Taylor, se verá un poco más adelante) y una **ecuación de Boltzmann discreta** (la cual se obtiene a partir de la forma final de la ec. (1.12) y cuyo nombre depende del tipo de simplificación que se escoja para el término de colisión, esto se muestra a continuación) (Wolf–Gladrow, 2000).

función de las invariantes de colisión dinámicas (masa, momento y energía cinética:  $m, m\mathbf{v}, mv^2$ ).

(Wolf–Gladrow, 2000) muestra que las invariantes de colisión generales pueden escribirse como una combinación lineal de las anteriores cantidades,

$$\phi(\mathbf{v}) = a_1 + a_2 \mathbf{v} + a_3 v^2 \quad (1.13)$$

y que las funciones de distribución que hacen nulo el operador de colisión son todas de la forma  $f(\mathbf{v}) = e^{a_1 + a_2 \mathbf{v} + a_3 v^2}$ , siendo la función de distribución de Maxwell es un caso especial entre dichas soluciones,

$$f^{eq} = \frac{\rho}{m(2\pi RT)^{D/2}} e^{-\frac{(\mathbf{v}-\mathbf{u})^2}{2RT}} \quad (1.14)$$

para  $D$  dimensiones espaciales<sup>27</sup>, donde  $\rho = \rho(\mathbf{x}, t)$  es la densidad del fluido,

$$\rho = m \int f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}, \quad (1.15)$$

$R$  es la constante universal de los gases,  $T$  es la temperatura del fluido, y  $\mathbf{u}$  es la velocidad macroscópica del fluido,

$$\mathbf{u} = \frac{m}{\rho} \int \mathbf{v} f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v} \quad (1.16)$$

La simplificación BGK propone que para el caso que estamos considerando (un gas ideal sujeto a temperatura y volumen constante (macroestado)) la función de distribución alrededor de un determinado punto del sistema en equilibrio termodinámico local es muy cercana a la función de distribución de equilibrio,  $f^{eq}$ . Es decir, en cualquier tiempo  $t$  dado la función de distribución del fluido,  $f$ , varía de  $f^{eq}$  una cierta cantidad  $\varepsilon$  de manera tal que  $f - f^{eq} \ll f^{eq}$ , por lo que dicha función de distribución  $f$  tenderá o se relajará hacia la función de distribución de equilibrio (Stensholt, 2009). De esta manera la función de distribución en el siguiente paso de tiempo en, por

---

<sup>27</sup> Sólo para estar en consonancia con la literatura escribiremos a partir de ahora la función de distribución  $f$  en función de la velocidad en lugar del momento.

ejemplo el punto  $(\mathbf{x}, \mathbf{p})$ , será la misma función de distribución más una pequeña cantidad proporcional a la diferencia entre ambas funciones de distribución<sup>28</sup>, digamos  $-\frac{f(\mathbf{x}, \mathbf{v}, t) - f^{eq}(\mathbf{x}, \mathbf{v}, t)}{\tau}$ .

Si se sustituye ésta última cantidad en la ec. (1.11) se tiene,

$$\left(\mathbf{v} \cdot \frac{\partial}{\partial \mathbf{x}} + \frac{\mathbf{F}}{m} \cdot \frac{\partial}{\partial \mathbf{v}} + \frac{\partial}{\partial t}\right) f(\mathbf{x}, \mathbf{v}, t) = -\frac{f(\mathbf{x}, \mathbf{v}, t) - f^{eq}(\mathbf{x}, \mathbf{v}, t)}{\tau} \quad (1.17)$$

donde  $\tau$  es dicha constante de proporcionalidad a la que se le llama el tiempo de relajación, lo anterior de (Stensholt, 2009) y (Succi, 2001). El autor indica además que en realidad el fenómeno de la relajación de un estado de no equilibrio a un estado de equilibrio en el flujo de un fluido ocurre a varias escalas de tiempo y que la variable temporal representativa de dicha relajación,  $\tau$ , varía de acuerdo a estas escalas, por lo que escoger  $\tau$  como una constante equivale a englobar todas estas escalas de tiempo en un solo valor. Existen mejores propuestas para la simplificación de la ecuación de Boltzmann que toman en cuenta más escalas de relajación y que se conocen como métodos de multirelajación. Algunos ejemplos de aplicación de éstos métodos se pueden consultar en (Premnath & Abraham, 2007), (Hong, Huang, Lin & Lin, 2015).

Obsérvese que el lado izquierdo de la ec. (1.17) indica la trayectoria que siguen las partículas representadas por las funciones de distribución a lo largo de las líneas de flujo, en tanto que el lado derecho muestra que esto depende de las interacciones de las dichas partículas mediante colisiones locales y que estas colisiones suponen una relajación continua hacia la distribución de equilibrio donde dicha relajación ocurriría a una sola escala de tiempo representada por el tiempo constante de relajación (o tiempo entre colisiones<sup>29</sup>),  $\tau$ . Esto último sólo para la simplificación BGK.

---

<sup>28</sup> (Succi, 2001) menciona que aunque el teorema en que está basada esta suposición de la simplificación BGK (derivada del cumplimiento del teorema H enunciado por el mismo Boltzmann) ha sido sujeto de polémicas ello no demerita el logro del gran científico al conectar por primera vez la física del complejo mundo microscópico y el mundo macroscópico ya que por vez primera explica de manera convincente por qué los procesos irreversibles en el universo siguen siempre una misma dirección y pasan continuamente a un estado de mayor desorden.

<sup>29</sup> Esto es así debido a que se asume el tiempo de duración de las colisiones como instantáneo.

Para obtener la ecuación de redes de Boltzmann considérese que la misma sólo existe para un conjunto de  $i$  velocidades por lo que, sin considerar por simplicidad el término de la fuerza externa se puede escribir la ec. (1.17) como,

$$\left(\mathbf{v}_i \cdot \frac{\partial}{\partial \mathbf{x}} + \frac{\partial}{\partial t}\right) f_i(\mathbf{x}, t) = -\frac{f_i(\mathbf{x}, t) - f_i^{eq}(\mathbf{x}, t)}{\tau} \quad (1.18)$$

donde la función de distribución  $f_i^{eq}(\mathbf{x}, t)$  también está definida sólo para el conjunto de velocidades  $\mathbf{v}_i$ . A la ec. (1.18) se le conoce como la *ecuación de Boltzmann discreta*.

Considérese ahora que el sistema, que es el espacio físico donde se encuentra el fluido y los demás componentes del mismo, es discretizado espacialmente como un retículo o red compuesto como una serie de puntos o nodos de dicha red en donde, a semejanza de los LGCA, sólo se permite que para los nodos que pertenezcan al fluido las distribuciones  $f_i$  se propaguen únicamente a través de las direcciones permitidas para las velocidades  $i$  hacia los nodos vecinos y las colisiones que suceden solamente en los nodos son las correspondientes reglas de actualización (lado derecho de la ec.(1.18)), las cuales ocurren con las distribuciones entrantes a los mismos. En adelante se considerará el método de las redes de Boltzmann también como un autómata que controla la simulación del flujo de un fluido.

En base a lo anterior (Wolf–Gladrow, 2000) demuestra que es posible discretizar espacial y temporalmente el lado izquierdo de la ec. (1.18) mediante diferencias finitas tomando como referencia el punto  $(t + \Delta t)$ , quedando la ecuación discretizada de la siguiente manera,

$$f_i(\mathbf{x} + \Delta \mathbf{x}, t + \Delta t) = f_i(\mathbf{x}, t) - \frac{\Delta t}{\tau} (f_i(\mathbf{x}, t) - f_i^{eq}(\mathbf{x}, t)) \quad (1.19)$$

Donde la densidad en cada nodo de la red se calcula:

$$\rho(\mathbf{x}, t) = m \sum_i f_i(\mathbf{x}, t) \quad (1.20)$$

La ecuación (1.19) es la llamada *ecuación de redes de Boltzmann BGK*. Es importante notar como la ecuación (1.19) indica de que manera es la

evolución de la función de distribución  $f_i$ , en la cual se identifican dos sucesos que ocurren en dos tiempos diferentes: en el tiempo  $t$ , en el nodo  $\mathbf{x}$ , ocurre el así llamado paso de colisión, en cual se suma  $f_i(\mathbf{x}, t)$  con el término de colisión,  $\frac{\Delta t}{\tau}(f_i^{eq}(\mathbf{x}, t) - f_i(\mathbf{x}, t))$  (nótese como se evalúa lo anterior sólo de manera local), y posteriormente, en el tiempo  $t + \Delta t$ , ocurre el paso de propagación, en el cuál se propaga o transfiere el resultado del paso de colisión al nodo adyacente  $\mathbf{x} + \Delta\mathbf{x}$ , en el sentido de la velocidad  $\mathbf{v}_i$ .

Finalmente se menciona que es posible recuperar en el límite macroscópico las ecuaciones de Navier–Stokes que son las que rigen el comportamiento de cualquier fluido a partir de la ec. (1.19) por medio de la expansión de Chapman–Enskog, el cual es un procedimiento matemático que parte de la escala mesoscópica y mediante métodos perturbativos obtiene las ecuaciones de Navier–Stokes en el límite macroscópico, razón por la cual se le llama también método multiescala. Para una explicación de la obtención de las ecuaciones de Navier–Stokes mediante la expansión de Chapman–Enskog puede consultar (Wolf–Gladrow, 2000) y (Velázquez Ortega, 2009).

# Capítulo 2

## 2.1 Modelo Conceptual, Geometría.

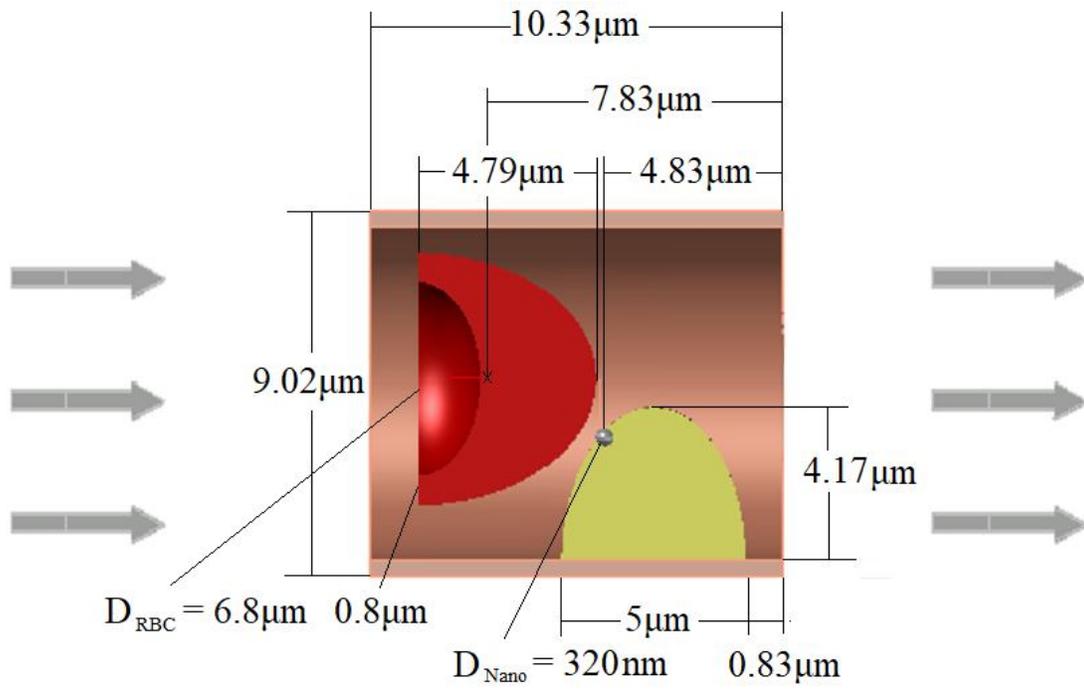
El punto de partida de nuestro modelo es el mencionado en (Janoschek, Toschi, & Harting, 2010) el cual realiza la simulación computacional del flujo sanguíneo en una bifurcación capilar de  $9.4 \mu\text{m}$  de diámetro y un vaso cilíndrico de  $63 \mu\text{m}$  de diámetro con estenosis en el primer caso.

Adicionalmente este artículo resultó útil para determinar un balance adecuado entre una resolución útil del sistema y un tiempo razonable de ejecución de la simulación para el hardware que se utilizó en este trabajo. Las especificaciones técnicas de dicho hardware se muestran en el Apéndice A.

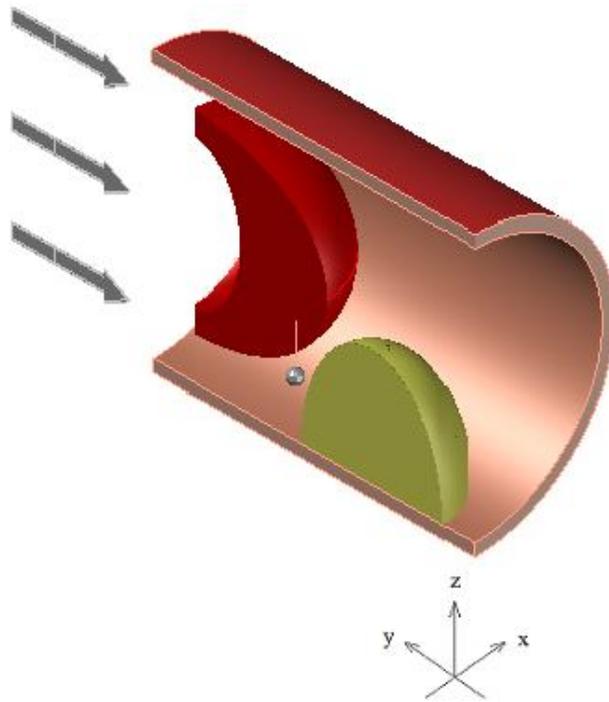
Es importante mencionar que todas las magnitudes físicas se deben transformar a magnitudes adimensionales antes de utilizarse en la implementación numérica del método de las redes de Boltzmann. Los cálculos realizados para la adimensionalización de las cantidades físicas más importantes se muestran en el Apéndice B.

Según se ilustra en la Figura 4, se propone como modelo conceptual el siguiente sistema en tres dimensiones:

1. Nuestro sistema se compone de un segmento de vaso capilar humano, la presencia de una RBC suspendida en el plasma sanguíneo circundante, un tumor canceroso situado en un extremo del vaso, una



(a)



(b)

Figura 4. Modelo geométrico.

concentración inicial de un determinado agente terapéutico para destruir el tumor y finalmente una nanopartícula que está liberando cierta dosis de dicho agente. Se considera que la misma cantidad de plasma que sale del vaso sanguíneo es la que entra a él fluyendo de la manera que se muestra en la misma Figura 4a. El objetivo principal de nuestro estudio es la relación de interacción entre el agente terapéutico y el tumor para la destrucción del mismo y como esta relación es afectada por los restantes componentes del sistema, el “nanobot” y célula de sangre roja. Esto para dos casos: cuando el agente terapéutico es liberado desde el “nanobot” y cuando el sistema es inicializado con una cantidad arbitraria de dicho agente terapéutico.

2. Se considera un vaso capilar promedio de dimensiones y condiciones fisiológicas correspondientes a un adulto sano normal, el cual tiene forma de un segmento cilíndrico de  $9.02 \mu\text{m}$  de diámetro (Shi, Zhang, & Wang, 2014) y  $10.33 \mu\text{m}$  de largo. El mínimo largo del capilar utilizado se seleccionó de manera que la RBC sea una célula puntero de una posible fila de células de sangre roja y de manera que la deformación de la misma no se vea afectada por otra célula inmediatamente atrás de ella (Pozrikidis, 2005). También que la misma viaje la distancia justo antes de que aparezca alguna otra célula e inmediatamente antes de colisionar con el tumor. Se considera que la sección transversal del cilindro está situado sobre el plano que forman los ejes X y Z del plano Cartesiano en 3D, el eje Y es paralelo al eje central de dicho cilindro, y el origen está localizado en la esquina inferior izquierda del cuadrado circunscrito de la sección transversal del extremo derecho del vaso según se ilustra en la Figura 4b.

3. Una RBC cuyo contorno exterior tiene forma de segmento de paraboloides de revolución, que en base a los parámetros proporcionados por (Shi, Zhang, & Wang, 2014) y (Shi, Wang, & Zhang, 2013) se obtiene para dicha célula una altura y diámetro de  $4.8 \mu\text{m}$  y  $6.8 \mu\text{m}$  respectivamente e interiormente tiene una cavidad de la misma forma de la primera situada en el centro de la base de la misma con dimensiones de  $1.7 \mu\text{m}$  de altura y  $5.2 \mu\text{m}$  de diámetro. Estas dimensiones dan a la RBC la forma aproximada de un paracaídas. Las dimensiones de la

célula se tomaron para un parámetro de deformación de la misma<sup>30</sup> de 0.1 (adimensional) según (Pozrikidis, 2005), y en base al cual también se determinó una velocidad promedio en el centro del capilar de

$$U_{AVG} = \left( \frac{(0.1) \left( 4.2 \times 10^{-6} \frac{\text{Kgm}}{\text{ms}^2} \right)}{1.2 \times 10^{-3} \frac{\text{Kg}}{\text{ms}}} \right) = 3.5 \times 10^{-4} \frac{\text{m}}{\text{s}}.$$

Los valores para estos cálculos fueron tomados también de (Pozrikidis, 2005).

4. Asimismo, de acuerdo con (Lázaro, Hernández-Machado, & Pagonabarraga II., 2014) el número de capilar para la velocidad promedio determinada en el punto previo es  $Ca = \frac{\mu_s U_{AVG} a^3}{\kappa b} = 92.78$  (note que  $Ca > 90$ , lo cual es importante para deducir el régimen morfológico de la célula de sangre roja<sup>31</sup>), donde  $a = 7.82 \times 10^{-6}$  m es la longitud característica de la célula de sangre roja,  $\kappa = 4.8 \times 10^{-19} \frac{\text{kgm}^2}{\text{s}^2}$  es la rigidez de flexión de la membrana y  $b$  es el diámetro del capilar.  $\kappa$  describe el efecto relativo de la viscosidad del fluido y las propiedades elásticas de la membrana de la RBC.

5. Un “nanobot” de tipo polimérico y de forma esférica hidrofóbica<sup>32</sup> de 320 nm de diámetro que libera Paclitaxel, una droga hidrofóbica utilizada comercialmente para este tipo de terapias y que ha sido ampliamente estudiada por Edelman y colaboradores (Creel, Lovich, & Edelman, 2000), (Hwang & Edelman, 2002), etc., con un coeficiente de

<sup>30</sup> (Shi, Zhang, & Wang, 2014) Menciona que de acuerdo a ciertos estudios realizados éste es el parámetro de control de deformación de la célula y está definido como  $\mu_f U_{AVG} / \mu_s$  donde  $\mu_f$  es la viscosidad del plasma sanguíneo,  $U_{AVG}$  es la velocidad promedio en el centro del capilar y  $\mu_s$  es la viscosidad del módulo cortante de la membrana.

<sup>31</sup> Aunque en (Lázaro, Hernández-Machado, & Pagonabarraga I., 2014) se menciona que la forma de paracaídas de una RBC se encuentra típicamente para  $Ca = 111$  en realidad, desde  $Ca > 90$ , la morfología de la RBC cambia abruptamente de una forma de zapatilla a una forma de paracaídas según se menciona en el mismo artículo. En nuestro modelo nos enfocamos en la alta energía de la forma morfológica de la RBC de modo que nuestros cálculos para los parámetros dados se encuentran dentro del régimen mostrado en (Pozrikidis, 2005).

<sup>32</sup> En este contexto, que repelen el agua y no son fácilmente solubles en ella.

difusión libre de  $4.2667 \times 10^{-6} \text{ cm}^2/\text{s}$  (según se estima en (Cremasco & Wang, 2012)) y un peso molecular de  $0.854 \text{ kDa}$ <sup>33</sup>. El agente terapéutico presente en el sistema sale del mismo una vez que alcanza los extremos del vaso sanguíneo.

6. Un tumor con forma de semielipsoide de semiejes de base de  $1.667 \mu\text{m}$  y  $2.5 \mu\text{m}$  (que forman un plano paralelo al formado por los ejes X–Y) y semieje ortogonal de  $4.167 \mu\text{m}$ , situado el centro a  $4.17 \mu\text{m}$  del extremo derecho del vaso capilar, esto para tratar de disminuir al máximo la posible influencia en los resultados del perfil de velocidades (y en el fenómeno de difusión-advección del agente terapéutico) debido a la deformación del flujo del plasma al pasar por el tumor situado en el otro extremo del vaso sanguíneo<sup>34</sup>.

7. El plasma sanguíneo se modela como un fluido Newtoniano sujeto a un gradiente de presión constante, que corresponde a un campo de fuerza de aproximadamente  $0.000768 \text{ pN}$  en la dirección del flujo entrante al vaso capilar.

8. Con base a la curva de tasa de liberación del agente terapéutico mencionada en (Hossain, Hossainy, Bazilevs, Calo, & Hughes, 2012) y dado el intervalo de tiempo real con el que se trabaja en la simulación (un máximo de  $0.0039 \text{ s} \approx 250,500$  iteraciones con una masa del “nanobot” modificada<sup>35</sup>) se toma como aproximación liberación continua de la droga las pruebas ejecutadas con la misma<sup>36</sup>.

9. Se considera que ocurren, para el caso del agente terapéutico, solamente los fenómenos de difusión y advección en el flujo del plasma sanguíneo.

---

<sup>33</sup> Se refiere a la unidad de masa atómica unificada conocida también como Dalton, es una unidad de masa que se utiliza para expresar la masa de átomos y moléculas, equivale a  $1.66 \times 10^{-27} \text{ kg}$ .

<sup>34</sup> Ya que se utilizan condiciones de frontera periódicas para el flujo de plasma entrante y saliente del segmento de capilar.

<sup>35</sup> Para la masa calculada del “nanobot” (ver final del apéndice B) se añadió una cantidad de masa suficiente para extender en el tiempo la estabilidad del sistema.

<sup>36</sup> Para efectos de observar al menos de manera cualitativa la destrucción del tumor se inicializa el espacio plasmático con una concentración adimensional del agente terapéutico después de 2 días de liberación del fármaco, que es cuando se alcanza la intensidad más alta de acción terapéutica según se menciona también en (Hossain, Hossainy, Bazilevs, Calo, & Hughes, 2012). Es debido a esto último que dicha concentración inicial se utiliza además como un umbral para iniciar el proceso de destrucción del tumor.

## 2.2 Modelo Matemático y Estrategia computacional.

Como un breve preámbulo a lo que sigue, es necesario indicar que existen varios modelos de las redes de Boltzmann que, como ya se mencionó en la introducción, son englobados bajo el mismo nombre de método de las redes de Boltzmann y que constan de tres partes principales según (Wolf–Gladrow, 2000):

1. Una red o malla que, como se menciona más adelante, sirve para discretizar el dominio o espacio fase (compuesto de una o más variables físicas de interés) del sistema termodinámico a estudiar.
2. La función de distribución de equilibrio, que para nuestro caso es el número de términos con que se desea trabajar la expansión de la serie de Taylor de la función de distribución de Maxwell. Para el caso del plasma sanguíneo la ecuación sería,

$$f^{eq} = \frac{\rho}{m(2\pi RT)^{D/2}} e^{-\frac{(\mathbf{v}-\mathbf{u})^2}{2RT}} \quad (2.1)$$

donde  $D$  es el número de dimensiones espaciales a considerar,  $\rho$  es la densidad del fluido,  $R$  es la constante universal de los gases ideales,  $T$  es la temperatura,  $\mathbf{v}$  es la velocidad individual de las partículas y  $\mathbf{u}$  es la velocidad macroscópica del fluido. Dicha distribución representa el perfecto equilibrio termodinámico del sistema a nivel local, es decir dentro de una determinada región espacial y temporal<sup>37</sup> de forma que si consideramos que las interacciones interpartícula son solamente colisiones binarias elásticas entre partículas rígidas puntuales (aproximación a partir de la cual parte Boltzmann para obtener su ecuación), entonces para cada microvolumen dentro de dicha región que contenga una determinada cantidad de colisiones, las partículas que entran a dicho volumen son exactamente iguales a las que salen del

---

<sup>37</sup> Nótese que las variables hidrodinámicas que intervienen en la ec. (2.1) dependen de la posición y el tiempo.

volumen de forma que la ganancia (o pérdida) neta de partículas en cada microvolumen debido a las colisiones es cero (Succi, 2001).

3. Finalmente una ecuación simplificada que describa el comportamiento del sistema. Esto debido a que es difícil trabajar con el complicado término íntegro-diferencial de colisión, por lo que varios autores han propuesto aproximaciones que faciliten el tratamiento numérico. En nuestro caso se utilizará la simplificación más común, que es la propuesta por (Bhatnagar, Gross, & Krook, 1954), la cual parte del teorema H propuesto en 1872 por Boltzmann, que como hace notar (Succi, 2001) se relaciona por primera vez la mecánica (interacciones moleculares) y la termodinámica (variables físicas de interés). Como ya se mencionó antes esta simplificación engloba todas las escalas de relajación que existen entre el régimen colisional y el equilibrio termodinámico local en una sola constante de proporcionalidad  $\tau$  llamada tiempo de relajación al equilibrio; con lo que hacemos que el término de colisión sólo sea función de las variables hidrodinámicas invariantes. La ecuación de la función de distribución de equilibrio para nuestro modelo se presenta un poco más adelante.

Para el caso del flujo del plasma sanguíneo, éste se modela según la ecuación de redes de Boltzmann<sup>38</sup> con aproximación BGK como sigue:

$$f_i(\mathbf{x} + \Delta\mathbf{x}, t + \Delta t) = f_i(\mathbf{x}, t) + \frac{\Delta t}{\tau} (f_i^{(eq)}(\mathbf{x}, t) - f_i(\mathbf{x}, t)) + F_i \quad (2.2)$$

donde  $f_i(\mathbf{x}, t)$  es una función de distribución de probabilidad de un cuerpo que indica la probabilidad de que una partícula de fluido virtual dada se encuentre en una posición entre  $\mathbf{x}$  y  $\mathbf{x} + \Delta\mathbf{x}$  para una velocidad permitida  $i$  en un tiempo  $t$ ,  $\tau$  es el tiempo de relajación al equilibrio,  $f_i^{(eq)}$  es la función de distribución de Maxwell para la velocidad  $i$  (mostrada al final del párrafo) y  $F_i$  es un término

---

<sup>38</sup> La cual, como ya se mencionó en el Capítulo 1, es una discretización espacial, temporal y del espacio de velocidades de la ecuación de Boltzmann con el término de colisión simplificado BGK (primer término del

lado derecho),  $\frac{\partial f}{\partial t} + \mathbf{v}\nabla f = -\frac{1}{\tau}(f - f^{(eq)}) - \mathbf{F} \cdot \frac{\partial f}{\partial \mathbf{p}}$ .

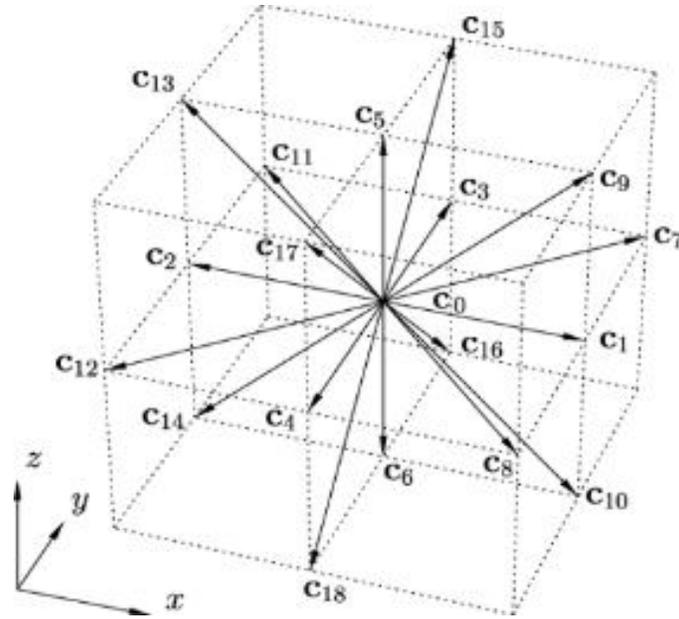


Figura 5. Modelo D3Q19. Este estencil se elige debido principalmente a la eficiencia que ya ha demostrado para simulaciones que involucran flujo sanguíneo (Janoschek, Toschi, & Harting, 2010).

relacionado con un campo de fuerza externa aplicada al sistema<sup>39</sup>. Se considera que el sistema se discretiza como un conjunto de puntos o nodos que están conectados entre sí formando un estencil<sup>40</sup> que muestra también las trayectorias (y las posibles velocidades) a través de las cuales se permite a dichas partículas del fluido moverse. Por el momento se etiqueta cada nodo del estencil simplemente como fluido para los nodos de fluido del plasma sanguíneo, pared para los nodos de las paredes del vaso capilar y partícula para los nodos ocupados por las partículas suspendidas.

<sup>39</sup> Cuando se trabaja numéricamente con la ecuación de redes de Boltzmann esto se hace de forma adimensional,  $\Delta t$  y  $\Delta x$  corresponden a las unidades básicas del espacio discretizado de tiempo y de longitud respectivamente, llamándosele a dichas unidades adimensionales paso de tiempo (pt) para el primer caso y unidades de lattice (ul) para el segundo, ver ref. (Sukop & Thorne, 2010).

<sup>40</sup> No todos los tipos de estenciles son factibles de utilizarse para simular sistemas físicos, sólo aquellos que conserven la isotropía, la invariancia Galileana, etc., ver (Succi, 2001). El primer término se refiere a que los componentes de los tensores del estencil ( $\sum_i c_{iC1} c_{iC2} \dots c_{iCn}$ , donde  $c_{iC1}$ ,  $c_{iC2}$ , ... son las componentes

cartesianas de la velocidad  $c_i$ ) de segundo y cuarto rango deben ser las mismas en cualquier sistema de coordenadas para poder recuperar las ecuaciones de Navier-Stokes en el límite macroscópico (Wolf-Gladrow, 2000). Algunos ejemplos son (según la nomenclatura introducida por (Qian, D'Humières, & Lallemand, 1992): D (dimensión), Q (velocidades permitidas)): D2Q7 (red triangular), D2Q9 (red cuadrada incluyendo velocidad cero), D2Q13, D2Q21, D3Q15 y D3Q19.

Para el modelo de lattice de Boltzmann seleccionado cuyo estencil se muestra en la Figura 5 la función de distribución al equilibrio puede obtenerse a partir de la ec. (2.1) mediante una expansión en términos de  $\mathbf{v}$  como sigue,

$$f_i^{eq} = \rho w_i (A_i + \frac{1}{c_s^2} \mathbf{v} \cdot \mathbf{c}_i + \frac{1}{2c_s^4} \overline{\mathbf{v}\mathbf{v}} : \mathbf{c}_i \mathbf{c}_i + \overline{\mathbf{G}} : \mathbf{c}_i \mathbf{c}_i) \quad (2.3)$$

donde se ha introducido el tensor  $\overline{\mathbf{v}\mathbf{v}} = \mathbf{v}\mathbf{v} - \frac{v^2}{3} \mathbf{1}$ , y  $c_s$  es la velocidad adimensional del sonido en la lattice. Las constantes  $w_i$  y  $A_i$  dependen del modelo específico de la lattice, que para nuestro caso (modelo de Boltzmann<sup>41</sup> D3Q19) son:  $w_i = 4/9$  para  $i = 0$ ,  $w_i = 1/9$  para  $i = 1, 2, 3, 4$ ,  $w_i = 1/36$ ,  $i = 5, 6, 7, \dots$ , y  $A_i = 1$ .

Debido a las velocidades del fluido y los valores de las variables propuestas para nuestro modelo que resultan en el número de capilar determinado anteriormente en el modelo conceptual, obsérvese que las fuerzas dinámicas ejercidas por el fluido son dominantes sobre las fuerzas elásticas ejercidas por la membrana (cuyo efecto en nuestro modelo se detalla un poco más adelante), por lo que su contribución es pequeña (Lázaro, Hernández-Machado, & Pagonabarraga II., 2014), es por esta razón que se despreció el tensor  $\overline{\mathbf{G}}$  en el cómputo de la función de distribución al equilibrio.

Las propiedades elásticas de la membrana RBC son descritas por la energía de flexión de Helfrich. Recientemente, los avances en la síntesis de partículas han facilitado las investigaciones en el papel que la elasticidad de las partículas juega en el proceso de liberación de sustancias desde varios medios, incluyendo nanodispositivos. En general, la literatura sugiere una relación compleja entre la elasticidad de las partículas y el proceso de liberación de drogas. Pero aún sin tomar en cuenta lo anterior, está claro por numerosos estudios in vitro e in vivo que la elasticidad de la partícula es un parámetro importante que puede ser aprovechado para mejorar la circulación sanguínea, localización para la aplicación de la sustancia e interacciones específicas con las células (Krüger, Varnik, & Raabe, 2011).

---

<sup>41</sup> El cual está ubicado en 3 dimensiones Euclidianas y puede transferir información (velocidad, concentración u otro) en 19 direcciones permitidas.

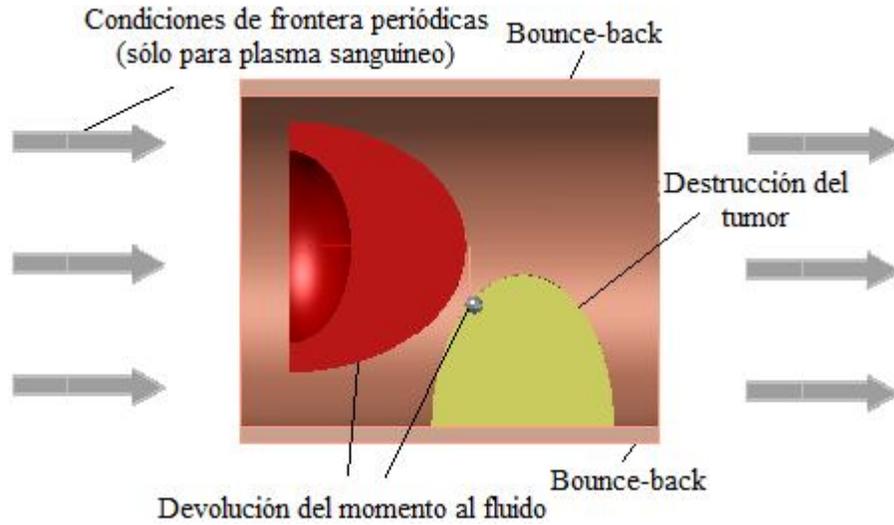


Figura 6. Condiciones de frontera del modelo.

Para nuestro modelo se utiliza el término de fuerza externa propuesto por (Lázaro, Hernández-Machado, & Pagonabarraga (II.), 2014) para microcanales altamente confinados (como es nuestro caso) y que toma en cuenta la interacción de la membrana de la célula de sangre roja con el fluido circundante,

$$F_i = 4w_i \left( 1 - \frac{1}{2\tau} \right) [\mathbf{f}_{ext} \cdot \mathbf{c}_i (1 + \mathbf{v} \cdot \mathbf{c}_i) - \mathbf{v} \cdot \mathbf{f}_{ext}] \quad (2.4)$$

La ecuación (2.4) toma en cuenta simultáneamente la fuerza externa ejercida sobre la membrana de la RBC por el fluido del plasma y la ecuación de Navier-Stokes que describe las interacciones hidrodinámicas entre el fluido interior y exterior de la membrana de la RBC, ver referencias (Lázaro, Hernández-Machado, & Pagonabarraga (II.), 2014) y (Lázaro, Hernández-Machado, & Pagonabarraga (I.), 2014).

Ahora bien, según se ilustra en la Figura 6, se consideran como condiciones de frontera las siguientes para el fluido plasmático y el agente terapéutico:

- Condiciones de frontera periódicas para el fluido de plasma que entra por un extremo del vaso capilar y sale por el otro extremo.

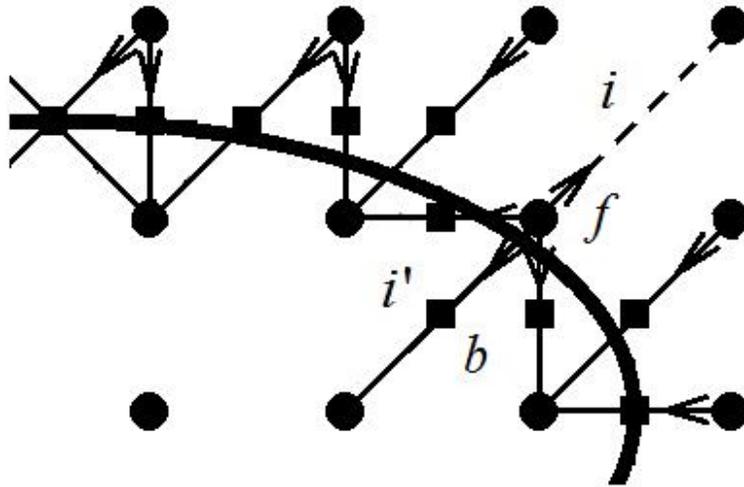


Figura 7. Sección de la partícula en la red de Boltzmann en donde se muestra la interacción de las distribuciones de los nodos de fluido que se encuentran afuera de la partícula dirigiéndose hacia los nodos internos de la misma así como la frontera que se encuentra en la trayectoria (a medio camino o enlace) entre unos y otros nodos (cuadrados).

- Para el caso de la droga presente en el plasma, simplemente se difunde o sale hacia afuera la misma por los extremos del vaso capilar.
- Condiciones de frontera “bounce-back” “no-slip” (misma velocidad que la frontera sólida) de medio enlace (la frontera real entre el fluido y el sólido está a la mitad de la distancia entre los nodos de fluido y los nodos de partícula) tanto para el plasma sanguíneo como para la droga.
- Para el caso de las paredes del capilar lo anterior significa simplemente que las distribuciones del fluido son proyectadas tal cual de regreso al fluido, en tanto que para las paredes de las partículas suspendidas se considera la devolución del momento de la partícula al mismo fluido debido a su movimiento, según lo propuesto por (Ladd & Verberg, 2001) pero aplicando al algoritmo según (Aidun, Lu, & Ding, 1998) y, al igual que el punto anterior se aplican éstas condiciones de frontera tanto para el plasma sanguíneo como para la droga terapéutica, la siguiente ecuación ilustra el último caso,

$$f_i(\mathbf{x}, t + \Delta t) = f_i(\mathbf{x}, t_+) + \frac{2\rho}{m} W_i \mathbf{c}_b \cdot \mathbf{c}_i \quad (2.5)$$

- donde  $i'$  es la velocidad opuesta a la velocidad  $i$  (o enlace sobre el cuál esta la dirección de dicha velocidad, véase Figura 7),  $t_+$  es el tiempo inmediatamente posterior al paso de colisión ( $t_+ - t \ll 1$ ),  $\rho$  es la densidad del fluido en el tiempo  $t$ ,  $c_i = |\mathbf{c}_i|$  es la velocidad adimensional para el enlace  $i$  ( $= 0$  para  $i = 0$ ,  $= 1$  para  $i = 1, \dots, 6$ ,  $= \sqrt{2}$  para  $i = 7, \dots, 18$ ),  $m$  es la masa adimensional de la partícula (normalmente = 1 unidad de masa (um)),  $W_i = w_i/(RT)$  y  $\mathbf{c}_b$  es la velocidad de la frontera (situada a la mitad de la distancia entre los nodos de partícula y los nodos de fluido adyacentes<sup>42</sup>, ver Figura 7), que se puede calcular en base a la posición del centro de masa de la partícula y su velocidad lineal y angular, el valor de  $f_i(\mathbf{x}, t_+)$  es simplemente el valor de  $f_i(\mathbf{x}, t)$  reflejado sobre el enlace  $i'$ .

Cuando, debido al movimiento de la partícula, un nodo de fluido se convierte en un nodo de partícula, la distribución para el nuevo nodo de fluido es, de acuerdo con la función de distribución del sistema en equilibrio para el modelo D3Q19, como sigue,

$$f_i^{eq}(\mathbf{x}, t) = f_i^{eq}(\rho(\mathbf{x}, t), \mathbf{u}(\mathbf{x}, t)) = \frac{\rho}{m} w_i \times \left[ 1 + \frac{3(\mathbf{c}_i \cdot \mathbf{u})}{c^2} + \frac{9(\mathbf{c}_i \cdot \mathbf{u})^2}{2c^4} - \frac{3\mathbf{u}^2}{2c^2} + \frac{9(\mathbf{c}_i \cdot \mathbf{u})^3}{2c^6} - \frac{9\mathbf{u}^2(\mathbf{c}_i \cdot \mathbf{u})}{2c^4} \right] \quad (2.6)$$

donde, si se sabe que  $c_s = c/\sqrt{3}$ , la ecuación anterior es simplemente una forma

más desarrollada de la ecuación (2.3),  $\rho(\mathbf{x}, t) = \sum f_i(\mathbf{x}, t)$ ,  $\mathbf{u}(\mathbf{x}, t) = \frac{1}{\rho} \sum f_i(\mathbf{x}, t) \mathbf{c}_i$

y  $c$  es la así llamada velocidad básica de la lattice (normalmente se elige = 1 ul / pt por simplicidad).

Para la destrucción del tumor por parte del agente terapéutico, se realiza un acoplamiento entre el autómata que controla la simulación de difusión del fármaco y el que simula el flujo del plasma sanguíneo de manera que la

---

<sup>42</sup> De acuerdo con lo mencionado en el Capítulo 1 y que se muestra más explícitamente en la Figura 7 nótese que la red o lattice en que se ha discretizado espacialmente el sistema se puede ver como un conjunto de puntos o nodos igualmente espaciados, que para efectos de la simulación computacional descrita en la siguiente sección se etiquetarán como nodos que corresponden al fluido, pared del vaso capilar, de partícula, etc.

advección del fármaco sea simulada considerando que no tiene por si misma velocidad de transporte sino que ésta se toma directamente de la velocidad macroscópica del flujo del plasma y considerando además que no existen interacciones entre ambos componentes del fluido según lo comentado para la simulación de transporte de solutos en (Sukop & Thorne, 2010).

Como ya se mencionó previamente en la introducción, se promedia la concentración del agente terapéutico sobre la superficie del tumor y si dicha concentración es mayor o igual a la concentración con que se inicializa el sistema según se menciona en el modelo físico se remueve uno o una cantidad de nodos del tumor proporcional a la cantidad de droga terapéutica presente en el flujo sanguíneo. Se sustituyen los nodos removidos por nodos de fluido y se inicializan éstos con la función de distribución de equilibrio (ec. (2.6)) para una densidad inicial  $\rho_{ini}$  y sustituyendo  $\mathbf{u} = 0$  en la misma ecuación.

Finalmente, para el cómputo de los movimientos de las partículas suspendidas, esto se hace aplicando las leyes de la mecánica clásica,

$$\begin{aligned} \frac{d\mathbf{X}(t)}{dt} &= \mathbf{C}(t) \\ M \frac{d\mathbf{C}(t)}{dt} &= \mathbf{F}(t) \\ \mathbf{I} \cdot \frac{d\boldsymbol{\Omega}(t)}{dt} + \boldsymbol{\Omega}(t) \times (\mathbf{I} \cdot \boldsymbol{\Omega}(t)) &= \mathbf{T}(t) \end{aligned} \tag{2.7}$$

donde  $\mathbf{C}(t)$  es la velocidad instantánea de la partícula,  $M$  es la masa de la misma,  $\mathbf{F}(t)$  es la resultante de las fuerzas ejercida sobre el centro de masa de la partícula,  $\mathbf{I}$  es el tensor del momento de inercia de la partícula,  $\mathbf{T}(t)$  es la resultante de la torca ejercida respecto al centro de masa de la partícula y  $\boldsymbol{\Omega}(t)$  es la velocidad angular de la partícula correspondiente.

### 2.3 Algoritmo utilizado.

La nanomedicina es uno de los campos donde más avances computacionales han tenido lugar. Como ya se mencionó anteriormente en el presente trabajo se intenta contribuir en el desarrollo de terapias innovadoras en aquellas enfermedades que cambian la morfología de interior del vaso capilar proveyendo una valiosa herramienta de investigación para el estudio de las interacciones microhidrodinámicas que ocurren a la escala de una simple nanopartícula.

En su generalidad el algoritmo utilizado para la implementación del método de las redes de Boltzmann consta de tres partes,

1. Se inicializa la parte del dominio compuesta por nodos de fluido con una densidad inicial, tal que  $\rho = \rho_{ini} = \sum f_i(\mathbf{x}, t_0)$  para cada posición  $\mathbf{x}$  de dichos nodos.
2. Se realiza el paso de colisión, que aunque en el presente trabajo lo realizaremos según se muestra a continuación, se desea utilizar una propuesta más realista para el término de colisión, por ejemplo el método de multirelajación (para algunos ejemplos de aplicación puede consultar (Premnath & Abraham, 2007) y (Hong, Huang, Lin, & Lin, 2015)),

$$f_i(\mathbf{x}, t_+) = -\frac{1}{\tau}(f_i(\mathbf{x}, t) - f_i^{(eq)}) + F_i \quad (2.8)$$

con  $f_i^{(eq)}$  según se indica en la ec. (2.6).

3. A continuación se realiza el paso de propagación que se podría resumir como sigue,

$$f_i(\mathbf{x}, t + \Delta t) = \begin{cases} f_i(\mathbf{x}, t_+) + \frac{2\rho}{m} W_i \mathbf{c}_b \cdot \mathbf{c}_i & \forall i' \in \text{fluido-partícula} \\ f_i(\mathbf{x}, t_+) & \forall i' \in \text{fluido-pared} \\ f_i(\mathbf{x} + \mathbf{c}_i, \Delta t, t_+) & \text{cualquier otro caso} \end{cases}$$

donde en los primeros dos casos se sustituye la propagación normal por las condiciones de frontera para la partícula suspendida y la pared del vaso capilar mencionadas en la sección anterior. Nótese que para la propagación normal, ésta se da sobre los enlaces que unen un nodo con sus vecinos y en la misma dirección de la velocidad permitida (discretizada) de la distribución a propagar, en cambio para los enlaces fluido-partícula y fluido-pared del capilar se reflejan las distribuciones post-colisión del mismo nodo que se tenían en el anterior paso de tiempo, más el momento que el mismo fluido está transfiriendo a la partícula para el caso del enlace fluido-partícula.

4. Se regresa al paso 2.

A continuación se explicará más a detalle la manera en que implementan las condiciones de frontera así como la forma de calcular la posición del centro de masa de las partículas suspendidas.

Iniciando nuestro cálculo en el tiempo  $t_0$ , en el tiempo  $t_0 + 1/2$  ocurre un intercambio de momento del fluido con la frontera efectiva de la partícula como se muestra en la Figura 7 (cuadrados); considerando que la fuerza asociada a dicho intercambio de momento es uniforme desde el tiempo  $t_0$  a  $t_0 + 1$  y en donde, al igual que en el caso de la interacción del fluido con las paredes endotélicas, se aplica una condición de frontera “bounce-back” de medio enlace, pero modificada según se muestra a continuación. Cabe hacer notar que para la aplicación de lo anterior se consideran las partículas como un cuerpo sólido rígido.

La distribución y el momento transmitido a un nodo de fluido  $\mathbf{x}$  adyacente desde un nodo de la partícula sobre el enlace  $i$  está dada según (Ladd & Verberg, 2001) y (Ding & Aidun, 2003) por,

$$f_i(\mathbf{x}, t + \Delta t) = f_i(\mathbf{x}, t_+) + \frac{2\rho}{m} W_i \mathbf{c}_b \cdot \mathbf{c}_i \quad (2.9)$$

$$\Delta \mathbf{p}_i = 2m\Delta x^3 (f_i(\mathbf{x}, t + \Delta t) - \frac{\rho}{m} W_i \mathbf{c}_b \cdot \mathbf{c}_i) \mathbf{c}_i \quad (2.10)$$

donde  $i'$  es el enlace exactamente opuesto a  $i$ ,  $\mathbf{c}_b$  es la velocidad del nodo frontera situado entre  $\mathbf{x}$  y  $\mathbf{x} + \mathbf{c}_i \Delta t$ , y  $f_{i'}(\mathbf{x}, t_+)$  es la distribución en el nodo  $\mathbf{x}$  sobre el enlace o en la dirección  $i'$  inmediatamente después de que ha ocurrido el paso de colisión en dicho nodo.

Si se sustituye (2.9) en (2.10) se tiene,

$$\Delta \mathbf{p}_i = 2m\Delta x^3 (f_{i'}(\mathbf{x}, t_+) + \frac{\rho}{m} W_i \mathbf{c}_b \cdot \mathbf{c}_i) \mathbf{c}_i \quad (2.11)$$

Dado que se considera que la colisión del fluido con la partícula es elástica, entonces la fuerza sobre la partícula en la dirección  $i'$  en el tiempo  $t_0 + 1/2$  y la posición  $\mathbf{x} + 1/2 \mathbf{c}_i \Delta t$  es como sigue,

$$\mathbf{F}_{i'}(\mathbf{x} + 1/2 \mathbf{c}_i \Delta t, t_0 + 1/2) = -\frac{2m\Delta x^3}{\Delta t} (f_{i'}(\mathbf{x}, t_+) + \frac{\rho}{m} W_i \mathbf{c}_b \cdot \mathbf{c}_i) \mathbf{c}_i \quad (2.12)$$

y la torca con respecto al centro de masa de la partícula, de posición  $\mathbf{X}$ , es,

$$\mathbf{T}_{i'}(\mathbf{x} + 1/2 \mathbf{c}_i \Delta t, t_0 + 1/2) = (\mathbf{x} + 1/2 \mathbf{c}_i \Delta t - \mathbf{X}(t_0)) \times \mathbf{F}_{i'}(\mathbf{x} + 1/2 \mathbf{c}_i \Delta t, t_0 + 1/2) \quad (2.13)$$

Asimismo la velocidad del nodo frontera correspondiente está dado por,

$$\mathbf{c}_b(t) = \mathbf{C}(t) + \mathbf{\Omega}(t) \times (\mathbf{x} + 1/2 \mathbf{c}_i \Delta t - \mathbf{X}(t)) \quad (2.14)$$

según la dinámica rotacional.

Para el caso en que haya un contacto entre la partícula y alguna de las paredes endotéticas, la fuerza ejercida sobre la partícula es como sigue,

$$\mathbf{F}_i(\mathbf{x} + 1/2 \mathbf{c}_i \Delta t, t_0 + 1/2) = 2f_i^{eq} (\rho = \rho_{ini}, \mathbf{u} = 0) \mathbf{c}_i \quad (2.15)$$

dirigida hacia la partícula. La torca correspondiente respecto al centro de masa de la partícula se calcula de la siguiente forma,

$$\mathbf{T}_i(\mathbf{x} + 1/2 \mathbf{c}_i \Delta t, t_0 + 1/2) = (\mathbf{x} + 1/2 \mathbf{c}_i \Delta t - \mathbf{X}(t_0)) \times \mathbf{F}_i(\mathbf{x} + 1/2 \mathbf{c}_i \Delta t, t_0 + 1/2) \quad (2.16)$$

La fuerza y torca resultantes en el tiempo  $t_0 + 1/2$  es la suma de todas las correspondientes fuerzas y torcas  $i$  actuantes sobre y respecto al centro de masa de la partícula.

Ahora bien, la fuerza en el tiempo  $t_0$  es el promedio de los valores obtenidos en el intervalo  $[t_0 - 1/2, t_0 + 1/2]$  como sigue (Aidun, Lu, & Ding, 1998),

$$\mathbf{F}(t_0) = \frac{1}{2}(\mathbf{F}(t_0 + 1/2) + \mathbf{F}(t_0 - 1/2)) \quad (2.17)$$

$$\mathbf{T}(t_0) = \frac{1}{2}(\mathbf{T}(t_0 + 1/2) + \mathbf{T}(t_0 - 1/2)) \quad (2.18)$$

Por simplicidad se utiliza el método de Euler<sup>43</sup> para el cálculo de  $\mathbf{C}(t)$  y  $\mathbf{\Omega}(t)$  en el tiempo  $t_0 + \Delta t$  ( $\Delta t = 1$ ),

$$\mathbf{C}(t_0 + 1) = \mathbf{C}(t_0) + \mathbf{F}(t_0) / M \quad (2.19)$$

$$\mathbf{\Omega}(t_0 + 1) = \mathbf{\Omega}(t_0) + \mathbf{T}(t_0) / I \quad (2.20)$$

donde  $I$  es el momento de inercia de la partícula en el plano Cartesiano correspondiente.

Para el caso de la RBC sólo se realizó el cálculo del momento de inercia en el plano X–Z debido a que se observa que las resultantes de la torca ejercida sobre la partícula en dicho plano así como la resultante de la velocidad angular correspondiente son mucho mayores respecto a las que se ejercen sobre los demás planos del tensor de inercia  $\mathbf{I}$ , por lo que las otras componentes se consideran despreciables. Esto está de acuerdo con la observación que hace (Posrikidis, 2005) el cuál menciona que para el régimen de flujo en que se encuentra suspendida dicha célula, ésta se mantiene en una posición estable y alineada con el eje axial del vaso capilar (para nuestro caso dicho eje es paralelo al eje Y).

---

<sup>43</sup> Aunque finalmente si se consiguió una implementación estable del software que utiliza el método de Heun sin iteración para realizar los cálculos de las velocidades lineal y angular del centro de masa de las partículas suspendidas, las pruebas finales se ejecutaron mediante el método de Euler.

En el Apéndice C se muestra el cálculo del momento de inercia en el plano X-Z para la RBC.

Igualmente para el cálculo de  $\mathbf{X}(t_0 + 1)$  se tiene,

$$\mathbf{X}(t_0 + 1) = \mathbf{X}(t_0) + \mathbf{C}(t_0) \quad (2.21)$$

Finalmente se hace notar que si se compara la ec. (2.15) con la ec. (2.11) y dependiendo de la elección de  $\Delta t$ ,  $\Delta x$  y  $m$ , la ecuación (2.15) es exactamente el momento transferido entre la partícula y el fluido circundante en cada paso de tiempo si se considera a la partícula como un sólido rígido en reposo y el fluido adyacente en reposo también y en estado de equilibrio. Para el modelo que se toma como base (Janoschek, Toschi, & Harting, 2010) menciona que debido a lo anterior y a que el fluido es incompresible y las velocidades son pequeñas, las fuerzas de lubricación que pueden resultar del contacto entre dos partículas son totalmente equilibradas y por lo tanto no causan atracción artificial entre las mismas, por lo que no es necesario hacer una corrección debido a estas fuerzas de lubricación como se hace normalmente en los modelos que consideran a las partículas como cuerpos sólidos rígidos.

Lo anterior es importante si se considera una expansión futura del presente trabajo al estudio de la interacción liberación de droga-RBC-destrucción del tumor para una concentración realista de nanopartículas en el flujo sanguíneo ya que, por ejemplo, simplificaría la complejidad del cómputo, aumentaría la eficiencia de ejecución, etc.

# Capítulo 3

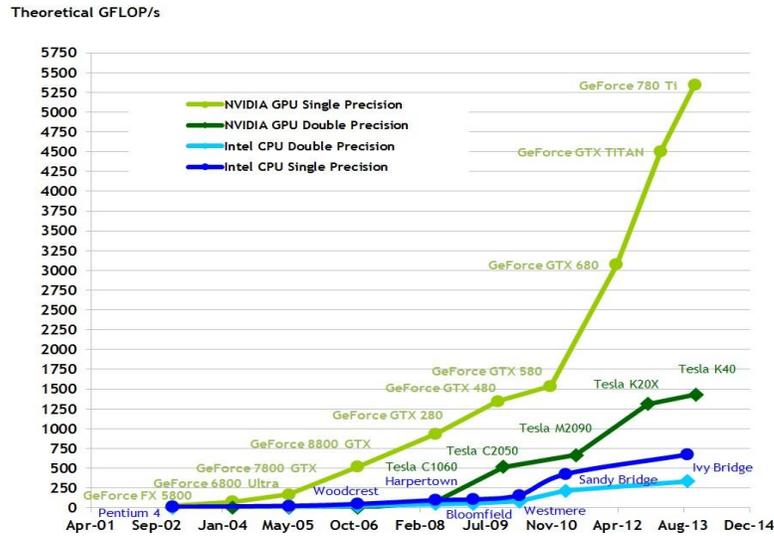
## **3.1 Arquitectura de los Dispositivos de Cómputo Unificado y Breve Introducción al Estándar de Paralelización MPI.**

En el presente trabajo proponemos un cómputo mediante la optimización eficiente de la comunicación y la construcción de la escalabilidad con un excelente potencial para una expansión futura utilizando el estado del arte de la tecnología actual y bajo un presupuesto razonable.

Sin embargo, éste es también un trabajo de exploración en el cual investigamos diferentes formas de optimización para cada componente paralelizable del sistema computacional dependiendo a veces de los recursos disponibles en cada experimento particular aprovechando las principales ventajas de cada tecnología seleccionada, pero partiendo de un esquema general que como ya se mencionó en el Capítulo 1 será el paradigma GPU-MPI.

Asimismo, en algunas partes de la arquitectura de paralelización, se eligen diferentes caminos de optimización dependiendo de diversos factores: partes demandantes de la funcionalidad, ancho de banda para transferencia de estructuras de datos varias entre el GPU y el servidor “host”, etc.

En la Figura 8 se puede apreciar una comparación entre el GPU de nVIDIA<sup>®</sup> y el CPU multinúcleo de Intel<sup>®</sup> para los años 2002 a 2013, donde se aprecia que la tendencia es que siga siendo cada vez más grande la relación de cantidad de operaciones de punto flotante por segundo entre ambas tecnologías.



(a)



(b)

Figura 8. Evolución de la cantidad de operaciones de punto flotante por segundo (a) y del ancho de banda de comunicación de su memoria principal (b) de el GPU de nVIDIA<sup>®</sup> y el CPU multinúcleo de Intel<sup>®</sup>. Imagen cortesía de nVIDIA<sup>®</sup> (NVIDIA Corporation, 2015).

La razón de la notable diferencia entre ambas tecnologías radica en que los CPUs son diseñados para ejecutar código secuencial altamente complejo en la forma más eficiente posible, lo que hace que se incorporen al chip más circuitería de lógica de control para lograr, por ejemplo, la predicción de ramificaciones (Parhami, 1999). Asimismo se incluyen también grandes

memorias de caché de instrucciones para reducir la latencia de acceso a instrucciones y datos. En cambio los GPUs, motivados por el creciente mercado de videojuegos se diseñan pensando en resolver paralelamente miles de cálculos aritméticos sencillos. Es por esto que en un multiprocesador de un GPU se optimiza el espacio en el chip minimizando la memoria compartida, la lógica de control y maximizando la cantidad de unidades de ejecución de modo que el multiprocesador ejecute miles de hilos en forma concurrente y pueda basar su estrategia de optimización en ocultar las latencias de instrucciones de más tiempo de ejecución, aritmética de punto flotante con pipeline<sup>44</sup> e instrucciones de bifurcación (Kirk & Hwu, 2010).

Una de las razones por las cuales los GPUs se han vuelto populares es que gracias a su bajo costo, ahora se puede tener todo el poder de la programación en paralelo en una laptop.

La Arquitectura de Dispositivos de Cómputo Unificado (CUDA por sus siglas en inglés) de nVIDIA<sup>®</sup> es una plataforma incipiente de cómputo que permite acceder a la capacidad computacional de los GPUs de la misma compañía mediante una interfaz de programación, la cual es una extensión del lenguaje C++ que elimina la necesidad de utilizar interfaces de programación de aplicaciones gráficas para la construcción de las aplicaciones.

Como ya se mencionó en la introducción, el algoritmo para el método de redes de Boltzmann es idóneo para paralelizarse y por lo tanto aprovechar el gran poder de cómputo ofrecido por los GPUs, que en nuestro caso es la parte de código paralelizable donde se concentra la mayor parte tiempo de cómputo efectivo dentro del ciclo principal de cálculo.

La arquitectura de un GPU CUDA consta de un arreglo de multiprocesadores multihilos, donde a su vez cada multiprocesador consta de un conjunto de procesadores que comparten lógica de control y caché de instrucciones. El multiprocesador organiza el conjunto de hilos de procesamiento en forma de bloques, los cuales comparten ciertos recursos para trabajar de manera cooperativa, además de que es en base a esta

---

<sup>44</sup> Estrategia para aumentar la cantidad de instrucciones que puede ejecutar un microprocesador en un tiempo determinado, la cual se puede ver como una línea de ensamblaje en la cual un producto (instrucción 1), se le realiza el pequeño proceso 1 y cuando pasa al pequeño proceso 2 otro producto (instrucción 2), se le comienza a realizar el proceso 1, y así sucesivamente para las demás instrucciones a realizárseles a los  $n$  procesos en dicho procesador.

agrupación que el planificador del GPU asigna los recursos de procesamiento a los hilos del procesador.

Este mismo multiprocesador administra los hilos de un bloque en grupos de 32 hilos llamados “warps”<sup>45</sup> los cuales además de ejecutarse concurrentemente ejecutan una y la misma instrucción a la vez, y aunque cada hilo del “warp” inicia su ejecución en la misma dirección del programa cada uno lo hace bajo su mismo contador de programa y estado de registros. Por lo que si hay una bifurcación, ésta se ejecutará en forma serial para cada opción de la misma ejecutándose primero los hilos de la primera opción de la bifurcación y deshabilitando los demás hilos, continuando con los hilos de la siguiente opción de la bifurcación y deshabilitando los primeros, y así consecutivamente con los demás hilos hasta que todos los hilos del “warp” vuelven a coincidir al final de dicha bifurcación<sup>46</sup>. De aquí se ve claro que se podrá tener una mejor eficiencia si se cuida que no se tengan demasiadas bifurcaciones en el código que se va a ejecutar en el GPU.

Los “kernels” de CUDA, es decir las extensiones al lenguaje C donde se define el código a ejecutar en el GPU pueden acceder a los diferentes tipos de memoria del mismo, entre las que se incluyen: los registros locales a cada hilo, la memoria compartida por todo el bloque de hilos, la memoria global de tipo DRAM, la memoria constante, la memoria de textura, etc. Los dos primeros tipos de memoria tienen muy baja latencia respecto al tercer tipo de memoria, es decir son muy rápidas de acceder por los núcleos de CUDA dado que se encuentran en el mismo chip de los multiprocesadores, siendo el primer tipo de memoria donde se almacenan las variables locales de cada hilo definidas en el “kernel”.

El segundo tipo de memoria es comúnmente utilizada para que los hilos puedan trabajar de manera cooperativa, es de acceso muy rápido. El tercer tipo de memoria es donde normalmente se almacenan los datos generales con los cuales va a trabajar el conjunto de hilos de toda la aplicación, siendo ésta común a todos los hilos y por lo mismo no es de un acceso tan eficiente como el de los anteriores tipos de memoria, por lo que se recomienda optimizar la

---

<sup>45</sup> El número de hilos por “warp” puede variar con la arquitectura del GPU (NVIDIA Corporation, 2015).

<sup>46</sup> A esta forma que tiene el multiprocesador de agrupar los hilos por medio de “warps” y ejecutar el código instrucción por instrucción por medio de dichos “warps” se le conoce como arquitectura de instrucción individual múltiples hilos o SIMT por sus siglas en inglés.

utilización de su ancho de banda de comunicación con el servidor “host” mediante la coalescencia de memoria<sup>47</sup>.

En el Apéndice A se muestran las especificaciones técnicas de los GPUs utilizados en el presente proyecto.

Para cerrar los comentarios acerca de la tecnología CUDA cabe mencionar que aunque el modelo de programación relacionado con la arquitectura SIMT tiene similitud con el modelo SPMD<sup>48</sup> en que cada hilo en un “warp” es capaz de ejecutarse independientemente de los otros hilos dentro del mismo “warp”. Sin embargo la arquitectura SIMT difiere de la arquitectura SPMD en que para la primera el planificador del “warp” es el que administra la ejecución o no ejecución de los hilos para diferentes rutas de bifurcación, en tanto en el modelo SPMD cada unidad de procesamiento es en sí una máquina de Von Newman ejecutando diferentes secciones del total de datos a procesar, pero todas las máquinas ejecutando el mismo programa (Baron & Higbie, 1992).

Basado en el modelo de programación paralela de paso de mensajes, en el cual los procesos a los cuales están asignadas las distintas tareas en que se divide la aplicación trabajan como entidades independientes (con su propio espacio de direcciones, variables privadas y pila) unos de otros colaborando y compartiendo información entre sí mediante mensajes<sup>49</sup> que pueden llevar datos o ser utilizados para sincronización, el estándar de programación MPI surge después de la colaboración de investigadores, asociaciones industriales, desarrolladores, etc., que en conjunto se le conoce como Forum y que agruparon en una especificación el conjunto de librerías y programas dispersos que para dar soporte al campo industrial de las arquitecturas distribuidas y cuyo fruto fueron primero el estándar MPI-1 liberado en 1994, posteriormente

---

<sup>47</sup> Técnica de optimización proporcionada por el fabricante del GPU mediante la cual se reduce el número de transacciones de lectura/escritura a la memoria global mediante la lectura de segmentos de localidades contiguas de memoria cuyas direcciones iniciales sean múltiplos de 32 bits, 64 bits o 128 bits (que de hecho también son los tamaños de segmentos de memoria accedidos por las transacciones) por parte de los hilos del mismo “warp” para así evitar que el planificador del GPU tenga que crear una transacción por cada localidad de memoria discontinua accedada por cada hilo o conjunto de hilos que así lo requiera (NVIDIA Corporation, 2015).

<sup>48</sup> “Single program, multiple data” – un programa, múltiples datos, considerada como una extensión a la taxonomía de Flynn, consultar (Darema, 2001).

<sup>49</sup> Los cuales no están limitados solamente a comunicaciones intermemoria sino que utilizan también el sistema de entrada /salida (por lo que pueden ser distribuidos).

el MPI-2 fue finalizado en 1996 y finalmente la versión actual, MPI-3, liberada en 2012.

Aunque el modelo asociado al interfaz es MMID (“Multiple Instruction streams, Multiple Data streams”, en esta arquitectura existe más de un procesador independiente que ejecuta su propio programa con sus propios datos) existen facilidades para ejecutar las aplicaciones como un SPMD (todos los procesos ejecutan el mismo programa pero bajo su propia planificación de ejecución).

Bajo el estándar MPI el sistema operativo ejecuta el código de la aplicación  $N$  veces mediante  $N$  procesos ordenados cuyo entero asociado llamado rango se puede consultar mediante las llamadas MPI correspondientes, asimismo se crea un comunicador por defecto, el cual es un objeto que indica el contexto de comunicación para cada proceso siendo posible definir más comunicadores incluyendo en él los procesos (o nodos) que se deseen.

La comunicación entre dos procesos (o comunicación punto a punto) del mismo o diferente comunicador se realiza en forma síncrona o asíncrona utilizando las correspondientes llamadas de envío y recepción de mensajes.

Para una introducción y/o profundización del entorno de programación MPI se puede consultar (Pacheco, 1996) y (Quinn, 2003). La guía del usuario, guía de instalación, tutoriales, etc., de una de las principales implementaciones de MPI, MPICH, del grupo del mismo nombre, puede consultarla en [W4], otras ligas de interés son: [W5], [W6], [W7] y [W8].

Debido al tipo de simulación que se realiza (un fluido con partículas suspendidas en él) y que aún se espera completar mediante la inclusión de otras características tales como paredes del vaso sanguíneo movibles, partículas deformables, etc., es que se hace imperativo utilizar técnicas de paralelización masiva. En este sentido el método de las redes de Boltzmann ha mostrado excelentes resultados cuando de paralelización se trata (ver por ejemplo (Dupin, Halliday, Care, Alboul, & Munn, 2007) y (Clausen, Reasor, & Aidun, 2010), y también para paralelización mediante GPUs, ver (Xiong, et al., 2012)).

Es en base a lo anterior que decidimos explorar dos de las técnicas de paralelización más actualizadas para hacer rentable el tiempo de simulación.

Una de dichas técnicas consistirá en el empleo de clústeres de computadoras y la otra utilizando clústeres de GPUs, donde se verificará la eficiencia obtenida en algunos proyectos previos tales como (Artoli, Hoekstra, & Sloot, 2006) para el caso de los clústeres de computadoras y (Biferale, et al., 2012) para los correspondientes de GPUs.

Cabe mencionar que como una extensión futura para el desarrollo del presente proyecto la transferencia de información entre ambos sistemas de cómputo se podría realizar utilizando una técnica de comunicación que permite una transferencia masiva de datos a través de Internet, para esto se utilizará el software de libre uso ProxyWS (Koulouzis, Cushing, Karasavvas, Belloum, & Bubak, 2012), el cual optimiza la comunicación a través de dicha red mediante flujos de datos, que a diferencia de las tradicionales implementaciones para servicios web basadas en SOAP<sup>50</sup> (“Simple Object Access Protocol”), permite que haya una conexión directa entre los servicios web, lo que a su vez lo hace idóneo para sistemas orientados al cómputo masivo.

### **3.2 Estrategia de Paralelización.**

La manera en que se aplicará la paralelización al modelo propuesto en el presente trabajo será aprovechando las ventajas que por su mismo esquema numérico ofrece el método de las Redes de Boltzmann. En primer lugar ya se ha mencionado con anterioridad que la ecuación (1.19) representa dos sucesos que deben considerarse en el cálculo: el paso de propagación y el paso de colisión. En las primeras versiones secuenciales de nuestro algoritmo se integró el paso de propagación y la aplicación de las condiciones de frontera tanto de las interacciones fluido–pared del vaso capilar como las de fluido–pared de las partículas suspendidas en un solo ciclo anidado dejando que el cálculo del paso de colisión fuera realizado por otro ciclo anidado separadamente.

---

<sup>50</sup> Especificación para el intercambio de información estructurada en implementaciones de servicios web en redes de computadoras [W9] y [W10] (basados a su vez en las conocidas arquitecturas SOA (Service-Oriented Architecture, la cual es una forma de construir sistemas distribuidos en donde la funcionalidad ofrecida es un servicio y las transferencias de información se realizan mediante mensajes (Koulouzis, Cushing, Karasavvas, Belloum, & Bubak, 2012)).

En una depuración posterior el cómputo del paso de propagación quedó en un ciclo anidado separado de la aplicación de las condiciones de frontera. Esto para tratar de simplificar la complejidad del ciclo anidado original correspondiente.

En una primera versión paralelizada se comenzó por utilizar el estándar MPI para distribuir la carga de trabajo de los anteriores ciclos entre cada uno de los nodos MPI levantados. Esto mediante la implementación MPICH2 para el sistema operativo Windows<sup>®</sup> del grupo MPICH, y verificando su correcto funcionamiento mediante la implementación HPC Server 2008 proporcionada por Microsoft<sup>®</sup> [W7], que es donde finalmente se corrieron las pruebas definitivas, notándose que no hubo diferencia aparente en los resultados obtenidos.

En la siguiente y última gran etapa de paralelización, debido a la intrínseca localidad del cálculo del paso de colisión, se decidió transferir ésta tarea de manera directa al GPU en tanto que en el caso del paso de propagación y la aplicación de las condiciones de frontera se decidió realizarlas mediante un estándar de paralelización que aprovechara las bondades proporcionadas por las tecnologías de optimización de código que ofrece un CPU. Esto último se consideró los más adecuado dada la complejidad del flujo del programa para ambos ciclos anidados (para el cálculo del paso de propagación y la aplicación de fronteras de las partículas suspendidas).

La intrínseca localidad del cálculo del paso de colisión significa que su paralelización es total es decir todos los nodos de la red<sup>51</sup> (sus cálculos asociados) que participan en el cómputo global pueden ser enviados directamente para su ejecución al GPU delegando un hilo del mismo a cada nodo de la red para tal efecto por lo que, si no se abusa de los recursos limitados que existen por bloque de hilos que proporciona cada multiprocesador multihilos del GPU, entre otras consideraciones de optimización, se puede aprovechar prácticamente al máximo la potencia de ejecución de dicho GPU.

La versión final en que se combinaron las tecnologías de paralelización GPU y MPI y que de hecho es lo que se denominará aquí como la arquitectura combinada quedó finalmente de la siguiente manera: primero se discretiza el

---

<sup>51</sup> Se refiere a los nodos del estencil del modelo de Boltzmann seleccionado. Diferénciese de los nodos o procesos MPI levantados en un servidor host cuando se haga referencia a ambos más adelante en el texto.

sistema (pared del vaso, fluido y partículas suspendidas) como un conjunto de nodos en una red que abarca dicho sistema de manera que las funciones de distribución sólo pueden “moverse” en los enlaces que unen uno con otro todos los nodos del estencil de la forma como se muestra en la Figura 5.

Enseguida se divide dicho sistema como un conjunto de bloques de datos (es decir de los datos asociados a los mismos como son las funciones de distribución para las distintas velocidades, la densidad y la velocidad macroscópica) en forma de prisma recto de base rectangular a lo largo del vaso y del mismo tamaño de la manera que se muestra en la Figura 9. Nótese que cada bloque tiene además dos capas adicionales de datos que sirven para la comunicación entre los demás bloques.

Cada bloque de datos se asocia a un nodo de MPI, es decir los cálculos del método de redes de Boltzmann correspondientes a dicho bloque son transferidos para su ejecución en el GPU por un proceso que corre en un núcleo de CPU específico levantado por el sistema operativo bajo la administración de la implementación MPI utilizada.

Ahora bien el cálculo de las componentes Cartesianas de la fuerza y la torca ejercida por el fluido sobre las partículas suspendidas y la lógica de control de la simulación, que debido a la carga minúscula de trabajo para el cálculo de las componentes y a la notable complejidad de dicha lógica de control, se “adaptan” de manera natural para su cómputo mediante un núcleo de CPU y por lo tanto serán realizadas de esta manera según nuestro esquema de diseño.

Para el cálculo del paso de colisión cada nodo de MPI será el encargado de hacer la llamada al “kernel” del GPU para realizar el cómputo de dicho paso para los nodos del bloque asociado.

El paso de propagación y el ciclo anidado, donde se aplican las condiciones de frontera “bounce-back” de medio enlace para las interacciones entre el fluido y la pared de la vena así como las correspondientes entre el fluido y la pared de las partículas suspendidas para el plasma sanguíneo se ejecutan directamente en el GPU utilizando para ello una copia de los arreglos involucrados en el cálculo y solamente intercambiando los punteros de éstos

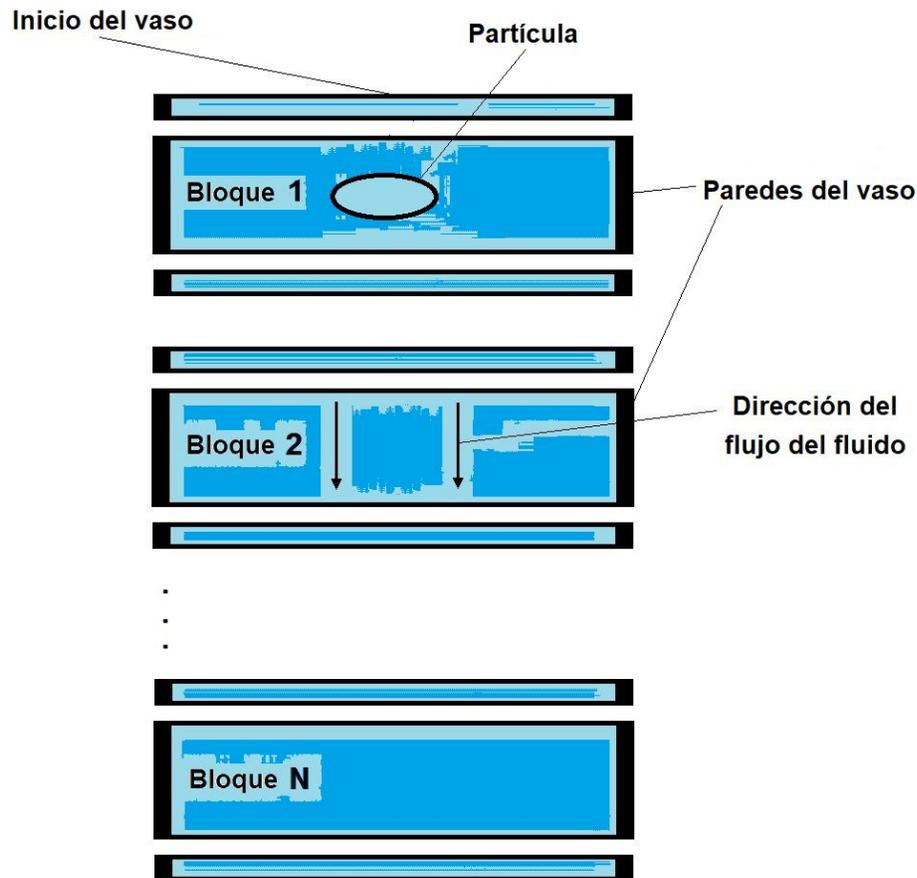


Figura 9. División del sistema vaso + partícula + fluido en bloques de nodos rectangulares.

para así evitar su continua copia entre el “host” y el GPU según se muestra en el esquema de paralelización<sup>52</sup> mencionado en (Rhöm, 2011) buscando con ello la optimización del uso del ancho de banda de acceso a la memoria del GPU. Cabe mencionar que lo anterior no se realizó para el autómata que controla la difusión del fármaco ejecutándose el cómputo del paso de propagación y la aplicación de las condiciones de frontera mediante los procesos MPI levantados en los CPUs multinúcleos del “host”.

La propagación de las distribuciones desde los nodos de fluido<sup>53</sup> de un nodo MPI dado a los nodos de fluido de los nodos MPI vecinos se realizará

<sup>52</sup> Que sería el esquema “push” y que sólo se refiere a ejecutar el paso de colisión antes del paso de propagación (Rhöm, 2011).

<sup>53</sup> Al igual que el modelo propuesto en (Melchionna, 2011), se considera que no existe fluido al interior de las partículas por lo que no habrá propagación desde los nodos correspondientes.

mediante las capas adicionales de nodos para comunicación de cada bloque propagando primero las distribuciones de los nodos frontera de cada bloque a dichas capas y después copiando, mediante las llamadas de comunicación de MPI respectivas, estas propagaciones a los nodos de frontera del nodo MPI vecino hacia donde se está realizando la propagación.

Como ya se mencionó en el Capítulo 1 la simulación computacional tanto del flujo del plasma sanguíneo como de la difusión-advección del agente terapéutico se realizaron mediante autómatas que actualizan las funciones de distribución correspondientes en base al esquema del algoritmo general utilizado para el método de las redes de Boltzmann. Adicionalmente se construyó un tercer autómata independiente de los anteriores para controlar la destrucción paulatina del tumor cuya funcionalidad básica ya se describió en el Capítulo 2.

A continuación se describirá como se realiza el cómputo general basado en el esquema numérico (método de las redes de Boltzmann) únicamente para el autómata que controla simulación del flujo del plasma sanguíneo siendo prácticamente la misma forma básica de computación para el autómata que controla el flujo del agente terapéutico solamente con algunas diferencias.

El cómputo se realizó utilizando dos arreglos para almacenar las funciones de distribución: el arreglo auxiliar,  $f_{temp}$ , y el principal,  $f$ . Como un primer paso se realiza el cómputo del paso de colisión a partir de los datos iniciales del arreglo principal  $f$  y actualizando solamente el mismo arreglo. Posteriormente se ejecuta el paso de propagación a partir de los datos actualizados del arreglo principal  $f$  y actualizando totalmente los elementos del arreglo auxiliar  $f_{temp}$ . Enseguida se transfieren los nodos de comunicación entre los bloques de datos vecinos actualizando su propio arreglo auxiliar local para completar el paso de propagación en todo el sistema. A continuación se realiza la aplicación de las condiciones de frontera tanto para las partículas suspendidas como para las fronteras del vaso capilar y el tumor a partir de los datos actualizados en el arreglo auxiliar del paso anterior y transfiriendo los resultados de regreso al arreglo principal para volverse a utilizar en la siguiente iteración. Finalmente se ejecuta la lógica de control de la simulación actualizando el arreglo principal si es necesario.

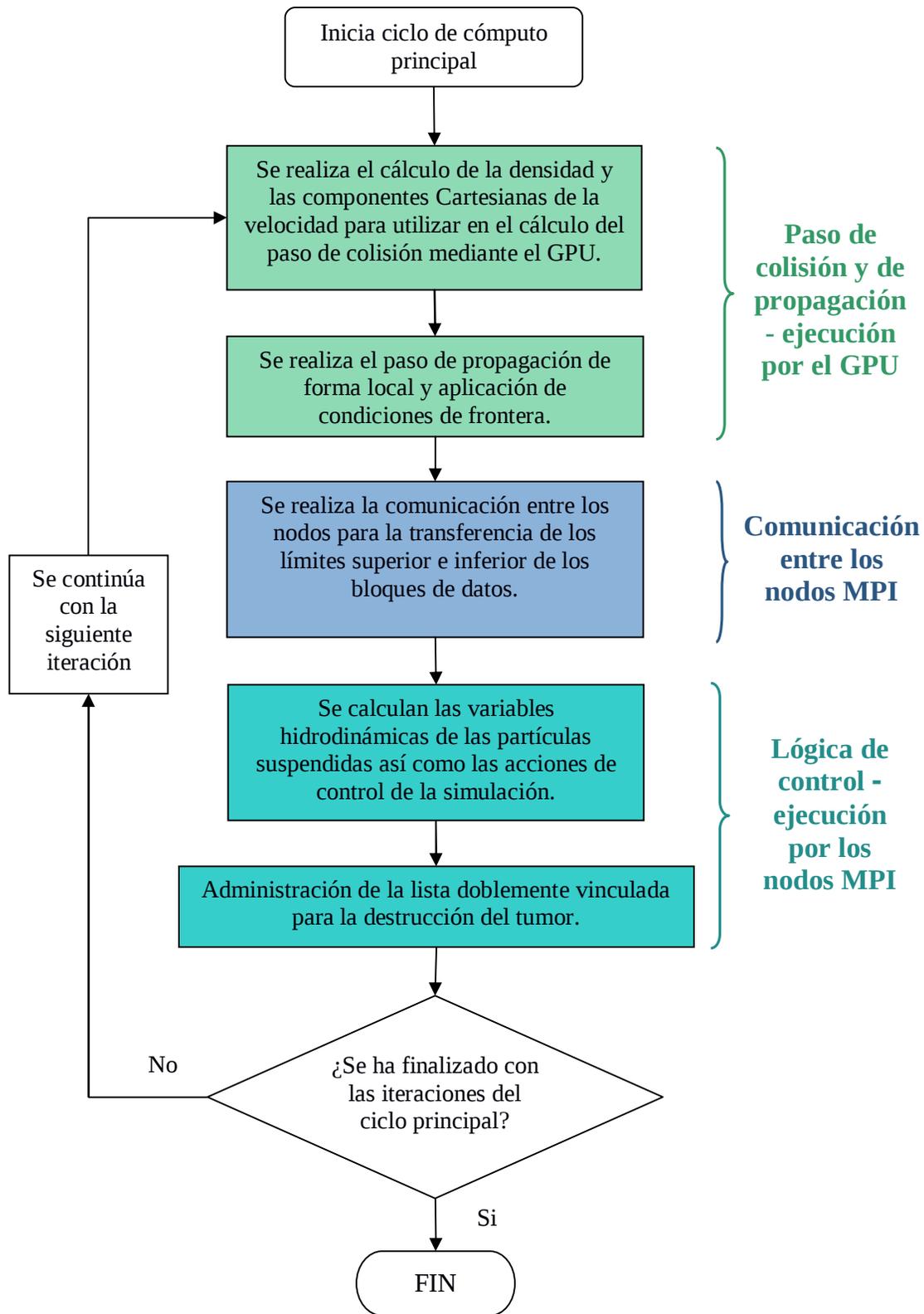


Figura 10. Diagrama general de flujo del ciclo principal de cálculo del programa basado en la arquitectura combinada GPU-MPI.

En la Figura 10 se muestra un primer esquema resumido del flujo del ciclo principal de cálculo en donde se utilizó la arquitectura combinada GPU-MPI.

La aplicación de las diferentes condiciones de frontera se realiza mediante el arreglo de la matriz de obstáculos, el cual es un arreglo de tipo entero de las mismas dimensiones del estencil del modelo de Boltzmann del sistema que sólo almacena en sus elementos un número entero positivo de un dígito para identificar (etiquetar) si el nodo del estencil asociado a dicho elemento pertenece a un nodo de fluido, de la pared del capilar o de partícula suspendida.

Anteriormente la actualización de la matriz de obstáculos se replicaba en cada nodo MPI para asegurar la coherencia de la misma ya que los resultados necesarios para calcular la posición de la partícula se comunicaban a todos los nodos permitiendo así que cada nodo realizara la actualización de dicha matriz de obstáculos calculando la posición de cada nodo de la partícula en el sistema. Sin embargo se detectó que esta tarea generaba un gran cuello de botella en el tiempo máquina para la generación de los nodos de partícula para la RBC. Esto debido a la gran cantidad de nodos a etiquetar en la matriz de obstáculos para la nueva posición de la partícula en cada iteración y que todos los nodos MPI levantados replicaban.

De lo anterior se observó que se podía mejorar substancialmente la escalabilidad del programa paralelizado dado que los nodos de fluido cubiertos o descubiertos por nuevos nodos de partícula es la única información necesaria para actualizar la matriz de obstáculos correspondiente en los nodos MPI vecinos del nodo MPI en cuya partición o bloque de datos asociados se encuentra el centro de masa de la partícula.

De esta manera la lógica de control para la actualización de la matriz de obstáculos sólo es realizada por el nodo MPI en el cual se encuentra o “pertenece” el centro de masa de la partícula (o partículas) suspendida y únicamente se comunican, en caso de ser necesario, la información requerida para la actualización de los nodos MPI vecinos de dicho nodo MPI de “pertenencia”.

Nótese que lo anterior no aplica para el cómputo de las distribuciones de fluido pertenecientes a cada bloque de nodos pues cada proceso de MPI

levantado en el “host” sólo administra las llamadas al GPU encargado de procesar aquellos nodos de fluido que se encuentren dentro de su partición de paralelización correspondiente.

Tal y como se anticipó con anterioridad el cálculo del paso de colisión y de propagación así como la aplicación de las condiciones de frontera son ejecutados mediante llamadas a los “kernels” de CUDA correspondientes, mientras que el resto del cálculo es ejecutado por cada uno de los nodos de MPI que se hayan levantado, asociando un nodo (proceso) de MPI a cada núcleo y a los cuales se les destina la parte de las tareas restantes según se muestra en la misma Figura 10 y que incluye la lógica de control de la simulación.

Es de hacer notar que las pruebas se ejecutaron de forma estable a pesar de que los “kernels” asignados a un mismo GPU se ejecutan de forma secuencial debido a que CUDA no puede ejecutar concurrentemente “kernels” con distinto contexto de ejecución (siendo éste nuestro caso ya que cada nodo MPI es un proceso distinto con su propio entorno o contexto de ejecución). De hecho lo ideal es que cada nodo MPI llame de manera exclusiva a un GPU para ejecutar trabajo en él.

Cabe mencionar que nuestra arquitectura es genérica, en el sentido de que, aunque el programa se corrió bajo una arquitectura convencional, es posible correrla también por ejemplo en un clúster de computadoras o en cualquier otra arquitectura que admita el estándar MPI.

La arquitectura también puede crecer es decir, aunque las pruebas se corrieron asignando a cada nodo MPI un solo GPU se le podrían asignar a cada nodo MPI más de un GPU y aún permitir que un nodo de MPI pueda acceder al GPU asignado a otro nodo, aunque esto último podría complicar considerablemente la lógica de control, además de que probablemente sería inútil pues podría aumentar también la inestabilidad (o al menos la reducción del performance) del programa debido a la gran cantidad de cambios de contexto que haría CUDA al ejecutar “kernels” llamados por nodos MPI distintos debido a que, como ya se mencionó, CUDA no puede ejecutar concurrentemente “kernels” con distinto contexto de ejecución.

A continuación se describirá en forma detallada la manera en que se particiona el dominio de nodos del sistema para la paralelización del programa computacional.

### 3.3 Detalle de la Implementación.

Para realizar el cómputo éste se dividió como se muestra en la Figura 11, en la cual se ilustra de manera gráfica la forma en que se particiona el dominio de nodos del sistema para la paralelización del mismo así como la topología de comunicación entre los procesos a los que se les asigna cada partición. La partición del dominio del sistema es como sigue: dado que básicamente el sistema operativo corre  $N$  procesos distintos e independientes en  $N$  núcleos de procesador, la red de  $L_X \times L_Y \times L_Z$  nodos<sup>54</sup> (correspondiente a las longitudes en cada uno de los ejes coordenados) se divide en bloques de datos de tamaño  $L_X \times \left(\left\lfloor \frac{L_Y}{N} \right\rfloor + 2\right) \times L_Z \times (19 \text{ velocidades según el estencil unitario D3Q19})$ , para los primeros  $N-1$  bloques y sólo sumando  $\text{mod}\left(\frac{L_Y}{N}\right)$  al factor  $\left\lfloor \frac{L_Y}{N} \right\rfloor + 2$  para el  $N$ -ésimo bloque (el +2 es debido a que se agregan dos capas de nodos arriba y debajo –a los que llamaremos límite superior e inferior, respectivamente– del bloque para comunicación con los demás procesos según se muestra en la misma Figura 11), donde cada núcleo corre uno de estos bloques en forma simultánea con los demás núcleos, esta manera de particionar el cómputo es bastante simple, relativamente fácil de programar y tiene un radio de comunicación a cómputo<sup>55</sup>,

$$O\left(\frac{2L_X \times L_Z}{\left(L_X \times L_Z \times \left\lfloor \frac{L_Y}{N} \right\rfloor\right)}\right) = O\left(\left\lfloor \frac{L_Y}{N} \right\rfloor^{-1}\right) \text{ para los primeros } N-1 \text{ bloques y}$$

$$O\left(\left(\left\lfloor \frac{L_Y}{N} \right\rfloor + L_Y \text{ mod } N\right)^{-1}\right) \text{ para el } N\text{-ésimo bloque, lo cual se considera}$$

aceptable ya que para nuestro caso  $L_Y \gg N$ .

Asimismo, cada uno de los bloques se transfiere para su ejecución en el GPU dentro de cada nodo de procesamiento de MPI. La comunicación entre los nodos de MPI se realiza de manera asíncrona empleando las llamadas de

<sup>54</sup> Aquí se habla de nodos del estencil del modelo de Boltzmann, diferénciese de los nodos (procesos) de MPI.

<sup>55</sup> Se define como la cantidad de comunicación (en bytes) dividida por el tiempo de cómputo y que puede ser una medida del impacto que causa la dicha cantidad de comunicación en el tiempo de cómputo global, ver referencia (Culler, Pal Singh, & Gupta, 1999).

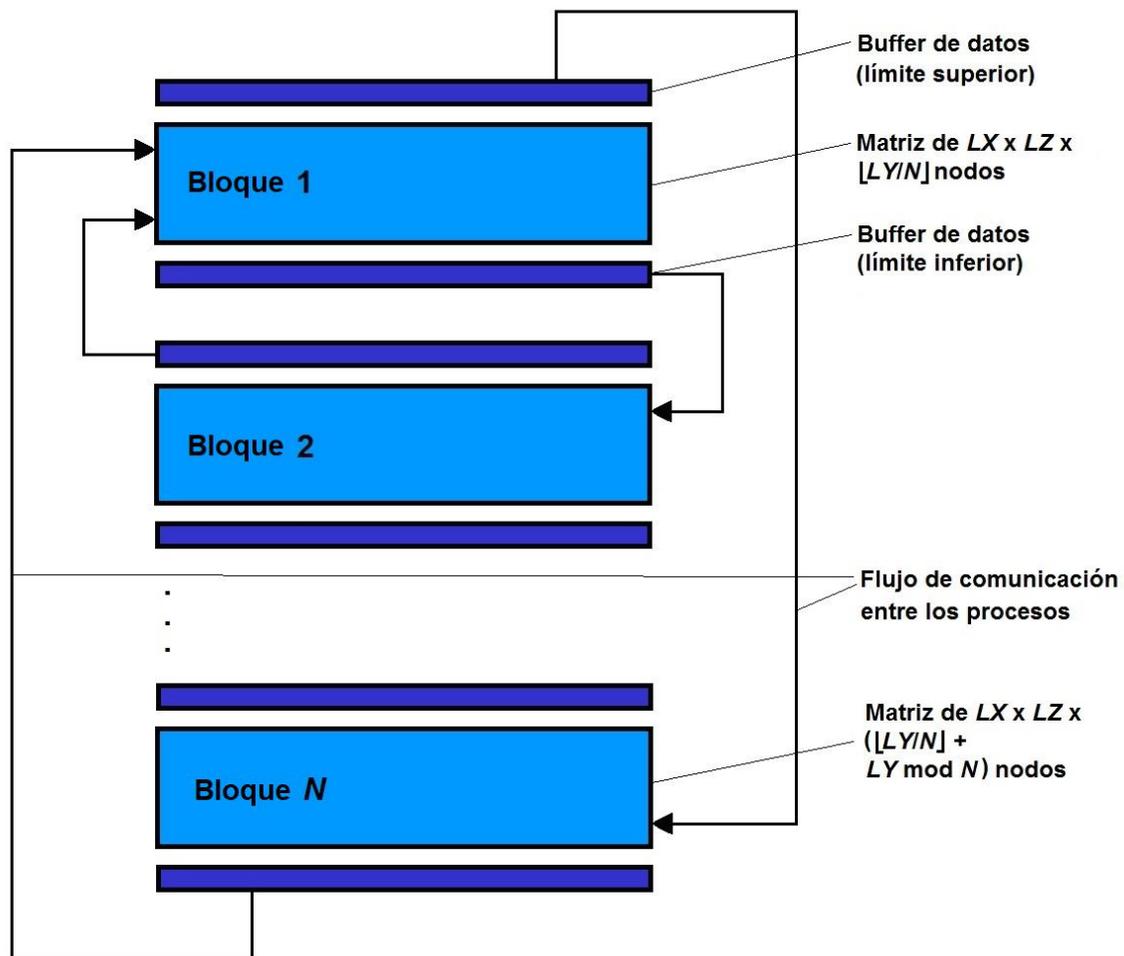


Figura 11. Flujo de comunicación entre los bloques de nodos que componen a la red de Boltzmann de la simulación.

envío / recepción correspondientes utilizándose además una topología en anillo para dicha comunicación<sup>56</sup> de la manera siguiente: se divide la comunicación en envío/recepción del límite superior y envío / recepción del límite inferior del bloque de datos donde en cada parte un nodo inicia o “dispara” la transmisión / recepción enviando el límite superior o inferior según sea el caso al nodo destino. Una vez terminada la transmisión / recepción, el nodo destino anterior envía su límite superior o inferior al siguiente nodo destino, y así se continua hasta que la transmisión / recepción del límite superior o el límite inferior de los bloques de datos finaliza con la

<sup>56</sup> Esta topología se heredó de versiones antiguas del programa informático con la idea de dar una mayor solidez a la comunicación entre los nodos MPI y dado que ha mostrado ser estable en la práctica es que se ha seguido utilizando hasta el presente.

recepción del límite superior o inferior de datos correspondiente por parte del nodo que inició el proceso.

En la Figura 12 se presenta el algoritmo del método de las redes de Boltzmann tal y como está implementado en el programa computacional, el cual es básicamente el mismo tanto para el autómata que controla el flujo del plasma sanguíneo como para el que controla la difusión y advección del agente terapéutico. Nótese en la Figura 12 que existe una diferencia importante en la implementación final de ambos autómatas ya que el cómputo del paso de propagación sólo se realiza en el GPU para el caso del plasma sanguíneo en tanto que para el caso de la droga dicho cómputo se realiza en el CPU del servidor “host”.

Asimismo para el primer caso únicamente se transfieren entre el GPU y el servidor “host” los nodos de comunicación de los bloques de datos en tanto que para el segundo caso se transfiere todo el bloque de datos del arreglo temporal de funciones de distribución entre el GPU y el servidor “host” para con ello actualizar el arreglo principal de las funciones de distribución para su utilización en el siguiente ciclo de iteración.

Esto se realizó así por que, como ya se mencionó, éste es un trabajo de exploración en el cual se probaron éstas dos maneras de intercambiar información de comunicación entre los bloques de datos aplicando las técnicas más recomendadas de optimización por el fabricante de las tarjetas gráficas para cada caso. Los resultados de las observaciones realizadas para las dos formas de transferencia de memoria anteriores se muestran en la sección de métricas del siguiente capítulo.

El algoritmo de la Figura 12 se encuentra anidado en otro más general que es el que controla el ciclo principal de cómputo mediante el cual se actualiza la posición del centro de masa de las partículas suspendidas mediante el método de Heún y que se ilustra en la Figura 13.

Finalmente se menciona que aunque la interfaz de programación del usuario de CUDA proporciona la facilidad para dimensionar el “grid” de hilos del GPU en 3D, se decidió barrer capas de hilos en 2D dentro de los “kernels” a ejecutar en el GPU principalmente por simplicidad pero también por que cuando se creó una versión de prueba en la que se manejaban los arreglos de

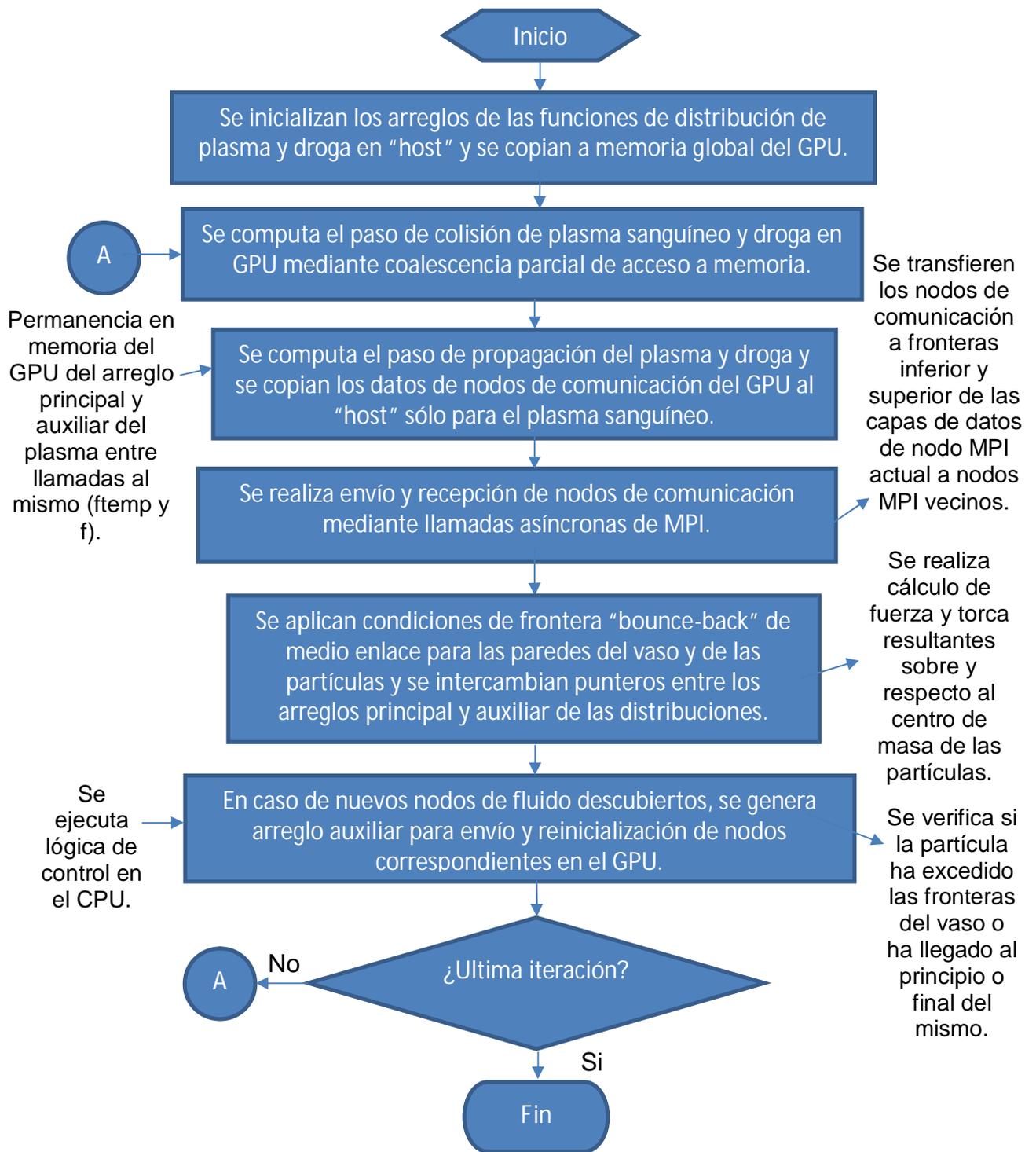


Figura 12. Diagrama de flujo de algoritmo que computa el método de redes de Boltzmann para el plasma sanguíneo y el agente terapéutico.

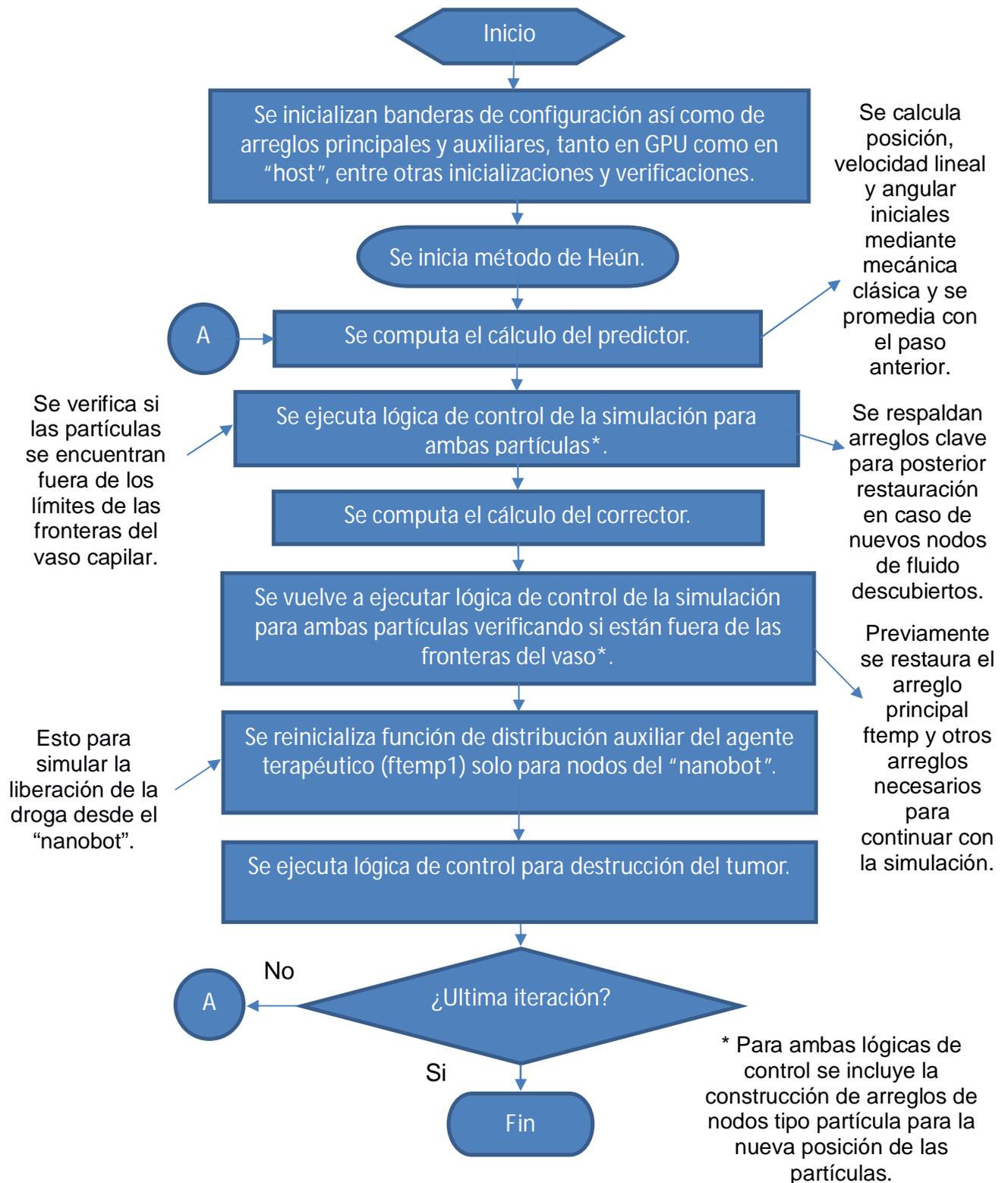


Figura 13. Diagrama de flujo de algoritmo para cómputo de variables hidrodinámicas mediante el método de Heún sin iteración.

las funciones de distribución principal y auxiliar y se dimensionaba el “grid” de hilos en 3D, no se notó una notable diferencia en la eficiencia de las dos versiones. Esto puede deberse a que la manera en que se organizó la memoria dentro de dichos arreglos es como un arreglo en 1D.

Sin embargo para optimizaciones futuras de manejo de memoria en el GPU cuando en trabajos posteriores se cuente con un mayor número de partículas suspendidas (“nanobots”, células de sangre roja o inclusive otro tipo de células) se preferirá utilizar los arreglos que participan en el cálculo del método de redes de Boltzmann en una dimensión y que sólo consten de los nodos del estencil de Boltzmann que interactúan entre si según se ha mencionado en los trabajos ya presentados (Bernaschi, Fatica, Mechionna, Succi & Kaxiras, 2010). Manejando de forma especial aquellos arreglos que interactúan con las fronteras del sistemas y con las partículas suspendidas.

### **3.4 Optimización.**

Para mejorar la eficiencia de nuestro sistema se hicieron las siguientes mejoras a los “kernels” llamados para su ejecución en el GPU:

- ✓ Se realizan los accesos a la memoria global del dispositivo (GPU) de forma parcialmente coalescente, se han organizado los arreglos utilizados en el cómputo del paso de colisión y propagación del método de las redes de Boltzmann de forma que los accesos de lectura / escritura a dicha memoria sean contiguos, es decir a localidades contiguas de memoria. Esto para minimizar el número de transacciones de acceso a la memoria global del GPU y optimizar así el uso del ancho de banda de transferencia de memoria entre el GPU y la máquina “host” según recomienda el fabricante (NVIDIA Corporation, 2015). Pero esta organización es parcial debido a que no está alineada, es decir la dirección inicial de memoria no es múltiplo de los segmentos de la memoria del dispositivo para su lectura / escritura mediante las transacciones de acceso a la memoria realizadas por el mismo dispositivo. Sin embargo esta pequeña deficiencia en el intento de optimizar la utilización de la memoria global del dispositivo se ve paliada por el caché de segundo nivel, L2, que minimiza el número de

transacciones parcialmente coalescentes y que está disponible a partir de los dispositivos con capacidad 2.0 en adelante<sup>57</sup>.

- ✓ Se definen las dimensiones del “grid” de hilos que se transfieren como parámetros al “kernel” que se ejecuta en el GPU de forma que se cargue el número máximo de bloques permitidos por multiprocesador y se optimice con ello el uso de los mismos.
- ✓ Se realiza “padding” de capas de datos adicionales en los arreglos principal y auxiliar a utilizar en el MRB de forma que las filas del “grid” de hilos que se definen en el GPU según el punto anterior sean múltiplos de la dimensión correspondiente (para nuestro caso sería la dimensión Y) de los bloques definidos para dicho “grid” de hilos. Esto para mejorar aún más el performance según se recomienda en (Kirk & Hwu, 2013)
- ✓ Se optimiza la carga de datos a la memoria y el cómputo del “kernel” de CUDA dividiendo estos en pedazos de cómputo y de carga para ejecutarlos concurrentemente mediante “streams” (conjunto de instrucciones que realizan las acciones de carga y cómputo mencionadas) de CUDA (NVIDIA Corporation, 2015), aprovechando que las tarjetas gráficas utilizadas en el presente proyecto proporcionan esta funcionalidad. Siendo la transferencia de memoria entre la máquina “host” y el GPU para el cómputo del método de las redes de Boltzmann en el caso de la difusión-advección del agente terapéutico la más beneficiada con la aplicación de ésta técnica de optimización.
- ✓ Uso de memoria no paginable del servidor “host” (NVIDIA Corporation, 2015), con lo cual no es necesario copiar los datos desde o hacia el dispositivo GPU, (de hecho éste tipo de memoria se optimiza también para el uso de “streams” mencionada anteriormente), entre otras correcciones y mejoras.

Aunque el estado actual de la paralización de nuestro sistema no está totalmente optimizado, el performance esperado es bueno ya que para la lattice promedio de los escenarios con que se realizaron las pruebas que se muestran en el siguiente capítulo, el sistema con una célula de sangre roja, un

---

<sup>57</sup> Se refiere a la arquitectura especial con la que el fabricante indica mejoras en varios aspectos de diseño de las nuevas generaciones de tarjetas gráficas fabricadas (NVIDIA Corporation, 2015). Para las tarjetas gráficas utilizadas en el presente proyecto es posible utilizar el caché de segundo nivel, L2.

“nanobot” de diámetro 15 veces más pequeño que dicha célula de sangre roja, un tumor y una lattice de  $56 \times 99 \times 56$ . Por lo que si consideramos los siguientes valores de volumen en unidades de lattice,

- $\text{Vol}_{\text{RBC}} = 19,839.21 \text{ ul}^3$
- $\text{Vol}_{\text{Tumor}} = 7,853.98 \text{ ul}^3$
- $\text{Vol}_{\text{Nodos pared}} = 30,037.39 \text{ ul}^3$
- $\text{Vol}_{\text{Sistema}} = 196,608 \text{ ul}^3$

no se ocupa más del  $(\text{Vol}_{\text{RBC}} + \text{Vol}_{\text{Tumor}} + \text{Vol}_{\text{Nodos Pared}})/(\text{Vol}_{\text{Sistema}}) \times 100 = 29.363 \%$  del total del dominio de nodos totales del sistema si se considera el peor caso con una tarjeta gráfica asignada a cada nodo MPI: el dominio del sistema particionado en sólo dos bloques de datos, asociando cada bloque a un nodo MPI, y con la RBC y el tumor completamente dentro de uno de los bloques.

# Capítulo 4.

## 4.1 Resultados y Discusión.

Con base en una minuciosa revisión de la literatura se ha encontrado que no existe aún un reporte de cómo se comporta una sola nanopartícula inmersa en el flujo sanguíneo a escala microcirculatoria. Lo cual, como se nota claramente, es de suma importancia no sólo para el diseño y optimización de aspectos varios de fabricación de nuevos dispositivos a escala nanométrica sino también para la creación y mejora quirúrgica de nuevas estrategias para posicionar la liberación de drogas terapéuticas en los sitios exactos de destrucción de cuerpos malignos tales como los tumores o edemas ateroscleróticos.

Lo anterior tomando en cuenta aspectos tan desafiantes (y amplios en su profundidad y estudio interdisciplinar) como el complicado mecanismo que rige la microhidrodinámica de flujo sanguíneo en un capilar que como ya se mencionó con anterioridad es un aspecto clave en la efectividad de esta clase de terapias altamente prometedoras en la aplicación de la nanomedicina a las enfermedades de tipo circulatorio.

Y es precisamente en este punto donde nuestra contribución puede ayudar al avance en el entendimiento detallado de los fenómenos y procesos que ocurren cuando se intenta realizar una terapia a una escala que de otra forma no sería confiable o altamente costosa y que según los estudios previos realizados es una herramienta realista, eficiente e innovadora de investigación médica.

En especial se menciona que en el estudio que se ha realizado hasta el momento referente al transporte de “nanocarriers” se observa que los modelos

reportados no consideran el fluido del capilar u otra partícula suspendida en el mismo por lo que ésta es de manera específica nuestra contribución al incluir una RBC y un tumor que va destruyéndose de manera paulatina según un criterio establecido para la cantidad de droga presente en el flujo circundante alrededor del mismo (Rodríguez–Romo & Ortigoza–Dominguez, 2019), y que como ya se mencionó es importante para el diseño de sistemas de liberación controlada de drogas a escala mesoscópica y microscópica (Mouffouk, Challa, Alrefae & Sini (2017)).

La nanopartícula entra en el flujo sanguíneo ya sea por inyección intravenosa por medio de un catéter o por absorción a través del colon. Antes de que podamos analizar la forma en que se libera la droga desde las nanopartículas se debe conocer la distribución de las mismas alrededor del cuerpo, lo cual afortunadamente ya se ha realizado, ver referencias (Cattaneo & Zunino (2014)) y (Grobler, Sierra & Viljoen (2016)).

Nuestro modelo sólo se enfoca en una pequeña sección de un capilar que contiene un tumor, dicha sección esta rodeada de tejido y el transporte de la droga se realiza por difusión y advección dentro de dicho capilar.

Nuestra segunda innovación se refiere a la arquitectura de cómputo utilizada para simular nuestro modelo. El único trabajo encontrado en la literatura relacionado con lo anterior se refiere a un algoritmo paralelizado para simular la distribución de la concentración de “nanocarriers” en un vaso capilar. Esto mediante el paradigma OpenMP-MPI (Hournkumnuard, Dolwithayakul & Chantrapornchai (2013)). En este artículo los autores estudian el campo magnético utilizado para guiar, tracear, o concentrar los “nanocarriers” en sitios específicos dentro del cuerpo humano, de aquí que el método es llamado liberación dirigida de droga por magnetismo. Nuestra arquitectura de software es claramente más eficiente.

Se ejecutó el programa computacional para un número de 100, 1000, 5000 y 10,000 iteraciones (las cuales corresponden a un intervalo de tiempo real de la simulación de  $t = 1.56 \times 10^{-6}$  s,  $1.56 \times 10^{-5}$  s,  $7.81 \times 10^{-5}$  s y  $1.56 \times 10^{-4}$  s, respectivamente). En la Figura 14 se muestra el perfil de velocidades para el plasma sanguíneo para una lattice de  $56 \times 62 \times 56$  y para un plano en eje X–Y correspondiente a un valor en el eje ortogonal al mismo de 21 ul.

La relativamente rápida destrucción del tumor se debe a que la concentración adimensional inicial es de  $3.8028 \text{ um/ul}^3$ , el doble de la

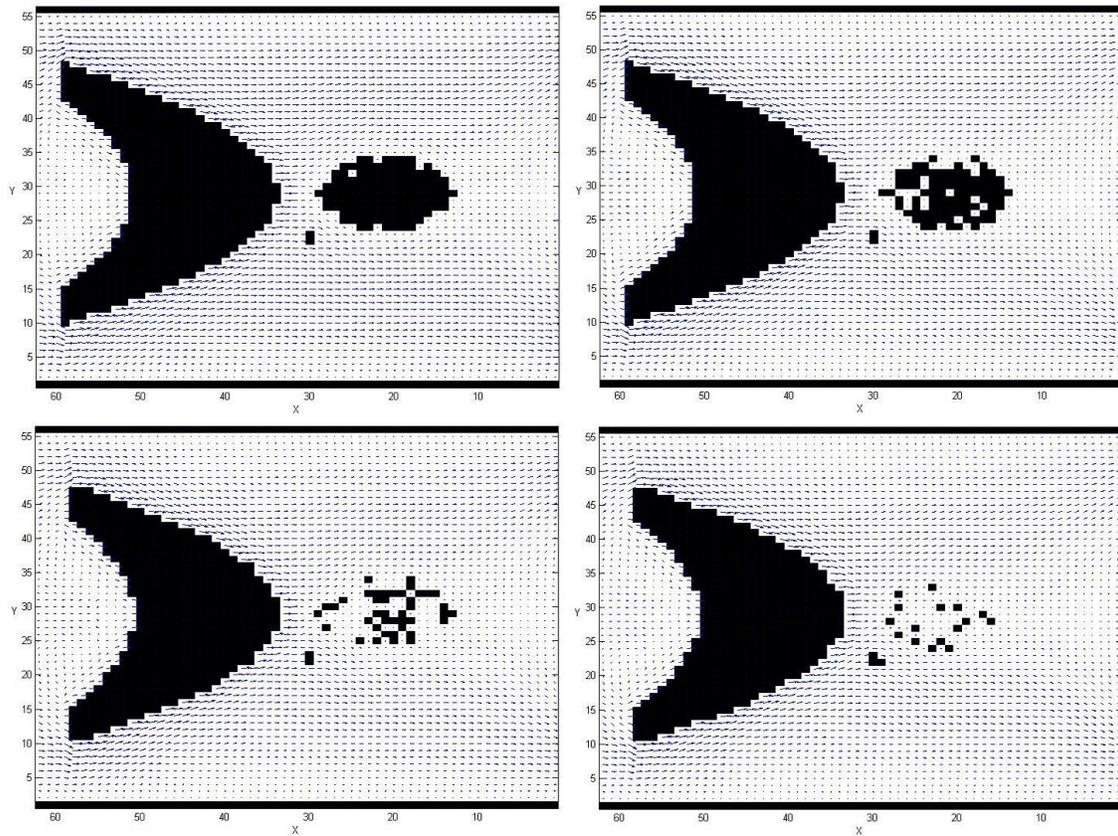


Figura 14. Perfiles de velocidad en un sistema de 56 X 62 X 56 nodos para, de izquierda a derecha, 100, 1000, 5000 y 10,000 iteraciones, respectivamente.

concentración crítica para el plasma. En este estado la droga ha saturado el plasma y se puede comenzar la simulación ya que, de otra manera, tomaría mucho tiempo ver el efecto en el tumor de la droga liberada solamente desde el “nanobot”. Esta es una consecuencia del efecto interescala de tiempo en nuestro modelo. De hecho este es el umbral utilizado por el autómata que controla la destrucción del tumor, este valor se conoce como la concentración crítica de Paclitaxel en el plasma (que para nuestro caso se seleccionó de  $1.9014 \text{ } \mu\text{m}^3/\text{ul}$ ). Esta concentración crítica deberá obtenerse mediante experimentación.

El lento movimiento de la RBC en la dirección del flujo es debido a la muy baja velocidad del fluido del plasma.  $U_{\text{MAX}}$  en el centro del capilar es de  $7 \times 10^{-4} \text{ m/s}$  (consultar el apéndice B).

Con el fin de observar más claramente los perfiles de velocidad para el plasma sanguíneo se muestra en la Figura 15 el perfil de velocidades para una

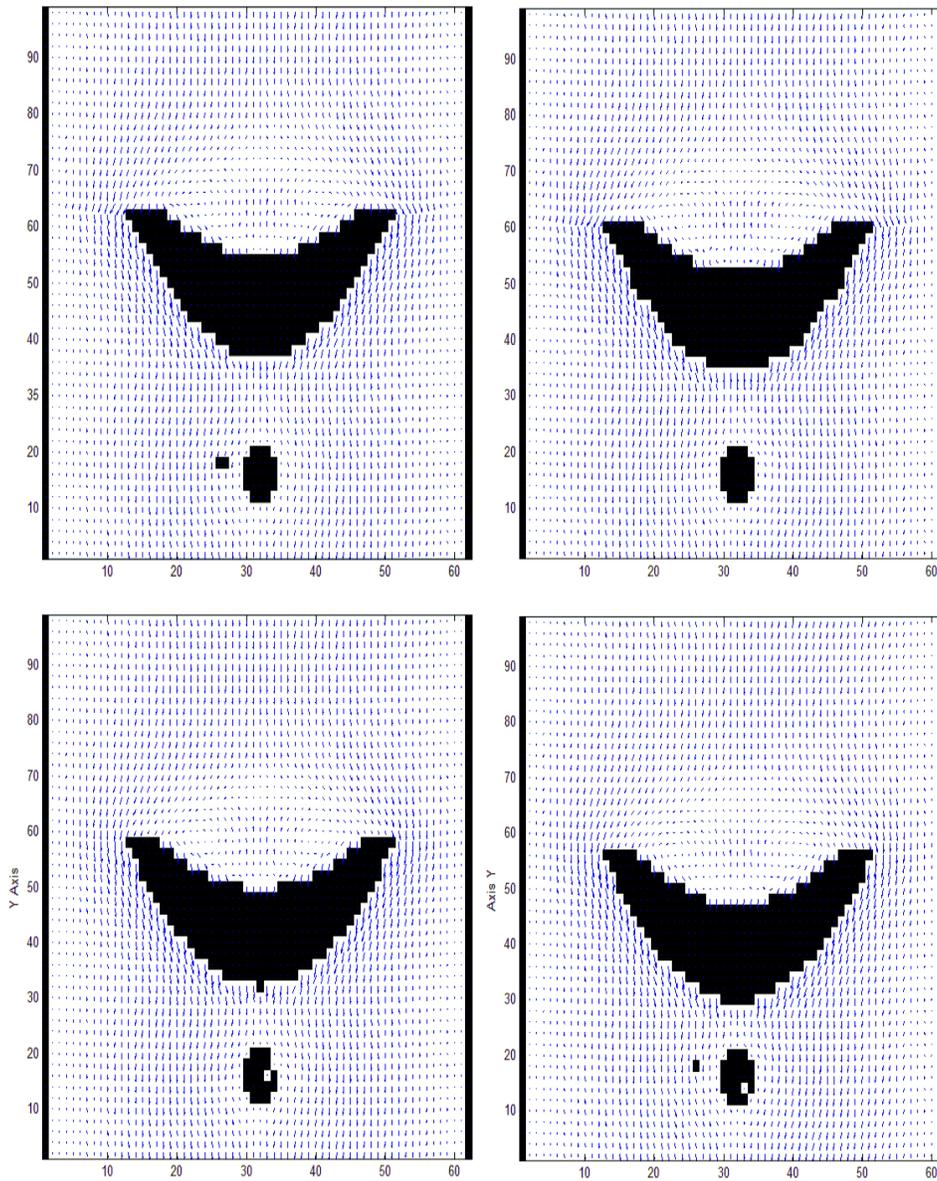


Figura 15. Perfiles de velocidad para un sistema de nodos para, de izquierda a derecha, 10,000, 20,000, 30,000 y 40,000 iteraciones, respectivamente.

lattice de  $62 \times 99 \times 62$  y con un campo de fuerza paralelo al eje axial del vaso capilar de magnitud  $F = 0.00185 \text{ pN}$ . La posición inicial del “nanobot” es de  $(25.345 \text{ ul}, 18.0 \text{ ul}, 25.345 \text{ ul})$  y la posición inicial de la RBC es de  $(31.0 \text{ ul}, 54.0 \text{ ul}, 31.0 \text{ ul})$ . La forma elipsoidal del tumor tiene semieje de  $5.0 \text{ ul}$  en el eje X,  $10.0 \text{ ul}$  en el eje Y y  $29.0 \text{ ul}$  en el eje Z, y el centro del tumor se localiza a  $15.0 \text{ ul}$  del extremo más cercano del capilar. Lo anterior para 10,000, 20,000, 30,000 y 40,000 iteraciones (las cuales corresponden a un intervalo de tiempo real de la simulación de  $t = 1.56 \times 10^{-4} \text{ s}$ ,  $3.13 \times 10^{-4} \text{ s}$ ,  $4.69 \times 10^{-4} \text{ s}$  y  $6.25 \times 10^{-4} \text{ s}$ , respectivamente).

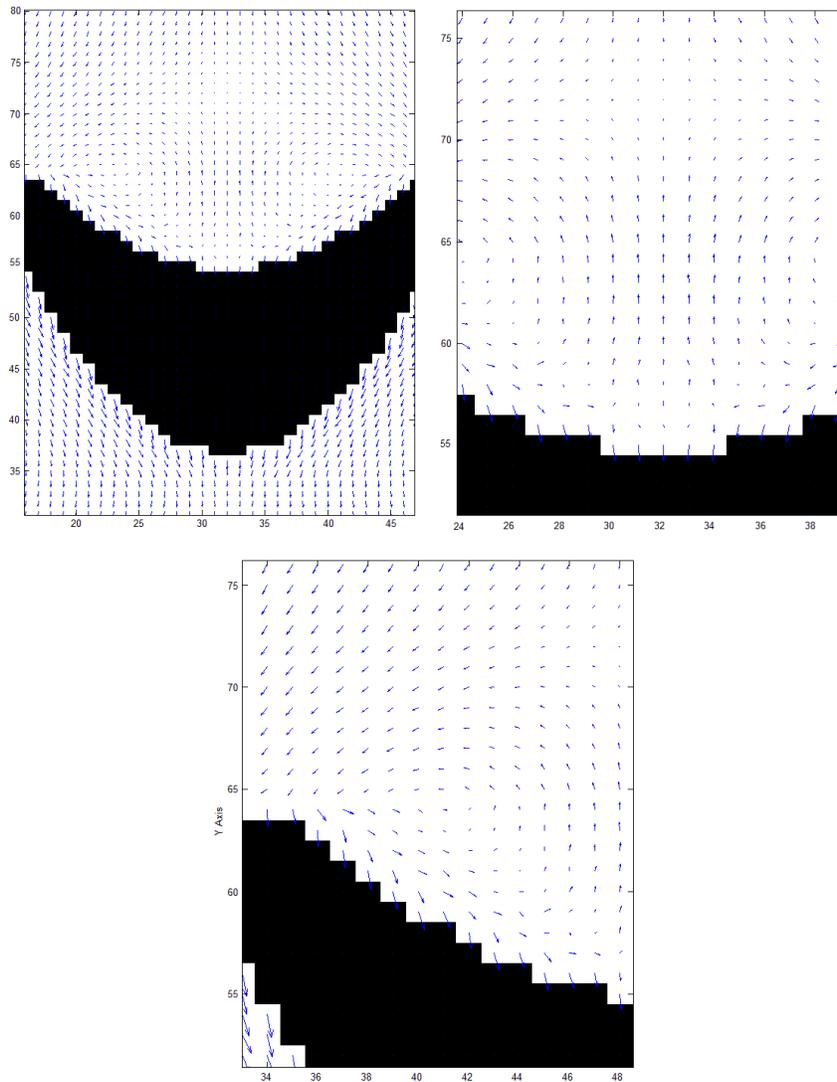


Figura 16. Sucesivos acercamientos de la Figura 15. Remolinos de plasma en la parte posterior de la RBC.

En la Figura 16 se muestran diferentes escalas de acercamiento de la parte de la RBC que se encuentra a contracorriente del flujo sanguíneo (y que se llamará la parte posterior de la misma) donde se observan claramente remolinos que se forman debido al movimiento mismo de la célula y que afectan la dinámica global de nuestro sistema, especialmente en los perfiles de concentración al interior del capilar debido a la advección del plasma sanguíneo. El fenómeno anterior es una de las principales contribuciones de nuestro trabajo pues hasta donde sabemos no se había reportado antes (Rodríguez-Romo & Ortigoza-Dominguez, 2019).

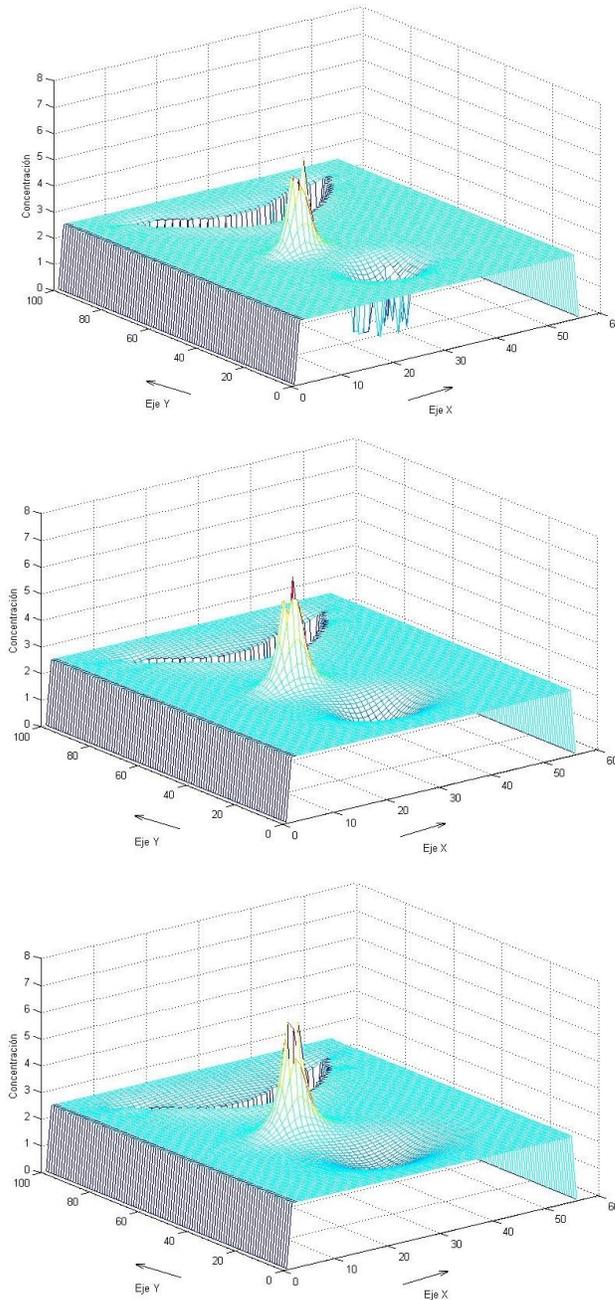


Figura 17. Perfil de concentración de Paclitaxel que muestra la liberación de droga desde el nanobot correspondiente a la mitad de la concentración inicial ( $1.2 \mu\text{m}/\mu\text{l}^3$ ). De arriba abajo las gráficas corresponden a 10,000, 20,000 y 30,000 iteraciones.

En la Figura 15 además de los remolinos que se forman en la parte posterior de la RBC se observa como es la distorsión del flujo entre la parte frontal de la RBC y el tumor. Asimismo se logra observar la distorsión de las líneas de flujo alrededor del “nanobot” y como éste desaparece en las 20,000 y 30,000 iteraciones de ejecución y vuelve a aparecer en las 40,000 iteraciones

de ejecución en este problema de tres cuerpos. Esto debido muy probablemente al mayor campo de fuerza que se ejerce sobre el “nanobot” y que lo hace “oscilar” en el intervalo de tiempo considerado.

Estudiar el perfil de velocidades es importante ya que estamos convencidos que el efecto de la advección puede influir de manera notoria en el transporte de la droga presente en el fluido plasmático y por lo tanto es un factor a tomarse en cuenta para mejorar la estrategia terapéutica de aplicación y localización de la droga.

Para un primer criterio de destrucción del tumor como el que se muestra en la Figura 14, el autómatas simplemente detecta si el promedio de la superficie de concentración de Paclitaxel ha alcanzado el umbral establecido para tal efecto. Si éste es el caso entonces un nodo o una cierta porción de nodos seleccionados al azar son removidos de la superficie del tumor y son transformados en nodos de plasma sanguíneo.

En un segundo criterio de destrucción del tumor se modifica el anterior criterio “normal” de destrucción del tumor de forma que se ralentiza la cantidad de nodos removidos del tumor para que, por ejemplo se remuevan cada 10 o 100 iteraciones. Esto se hizo con la intención de observar el proceso de destrucción del tumor para intervalos de tiempo más grandes ( $> 10,000$  iteraciones) y que se aplican a los experimentos correspondientes de las figuras 18 y 19 descritos un poco más adelante.

Las concentraciones adimensionales de Paclitaxel en la evolución del plasma se muestran en las Figuras 17 y 18. En la Figura 17 se muestran tres gráficos tridimensionales de esta evolución para 10,000, 20,000 y 30,000 iteraciones para una lattice de  $56 \times 99 \times 56$  y para una concentración inicial de la droga en el plasma sanguíneo de  $2.4 \text{ um/ul}^3$ . Las posiciones ocupadas por la RBC con forma de paracaídas y el tumor son oscuras debido a que en estas localidades la concentración es nula. Se modificó la liberación del “nanobot” para observar mejor la influencia de la droga liberada desde el mismo en el sistema. En la misma Figura 17 se puede ver también como se comporta el transporte de Paclitaxel debido a la difusión y la advección.

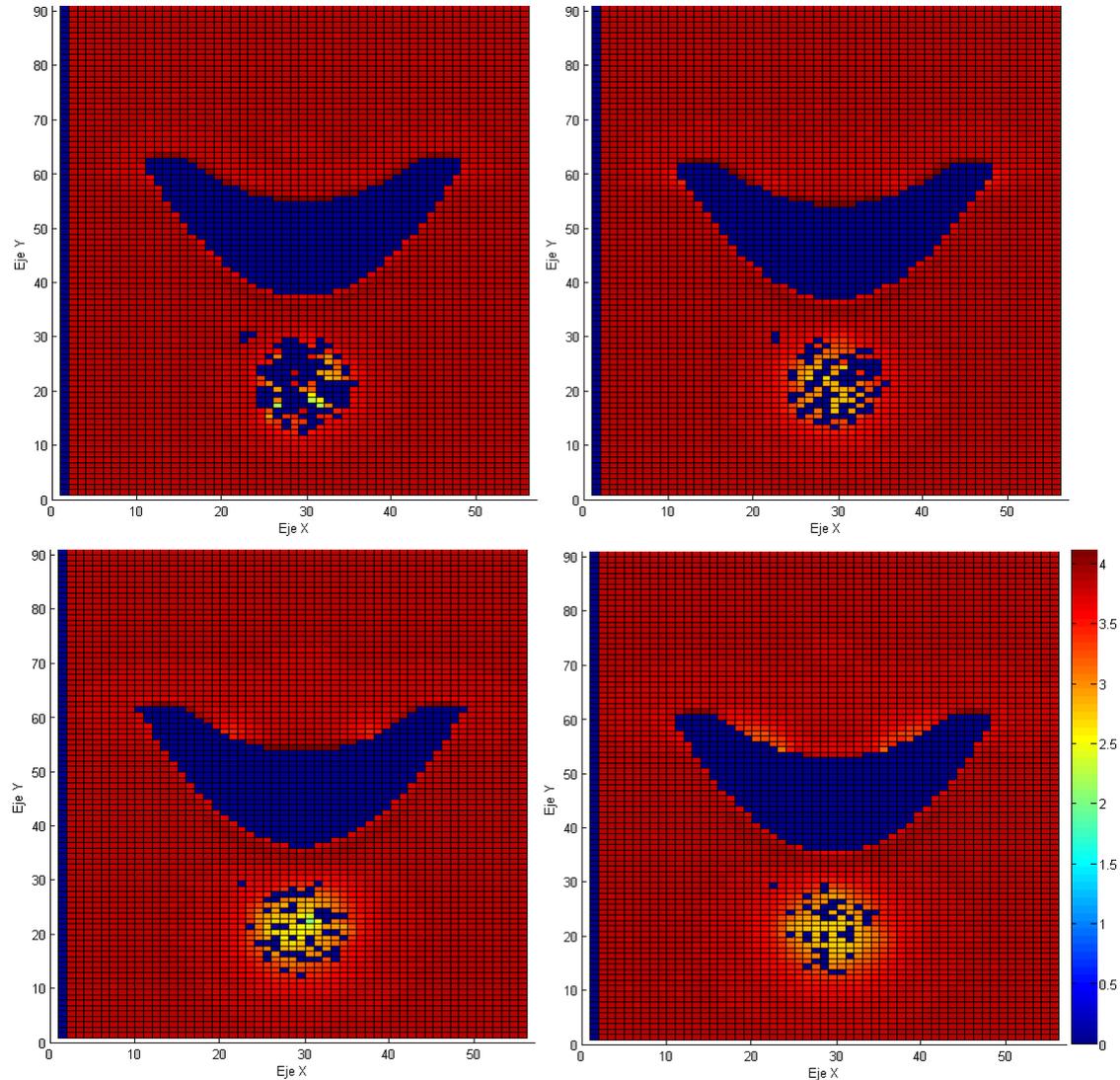


Figura 18. Perfil de variación de concentración adimensional de Paclitaxel para un sistema vaso capilar-RBC-tumor. De izquierda a derecha las gráficas corresponden a  $1.56 \times 10^{-4}$ s,  $3.12 \times 10^{-4}$ s,  $4.68 \times 10^{-4}$ s y  $6.24 \times 10^{-4}$ s.

Por otra parte, si seleccionamos un plano para el valor en el eje Z de 21 ul (posición del “nanobot” en ese plano) para la misma lattice de  $56 \times 91 \times 56$  y una concentración inicial de  $3.8 \text{ ul}^3$ , se obtienen los perfiles de concentración que se muestran en la Figura 18 para diferentes intervalos de tiempo.

Aún cuando no es muy notorio, hay un decremento global de la concentración de la droga debido a que las fronteras del capilar están abiertas. Nótese también la (aunque leve) baja concentración del Paclitaxel en la parte posterior de la RBC (que se debe probablemente a la continua aceleración de

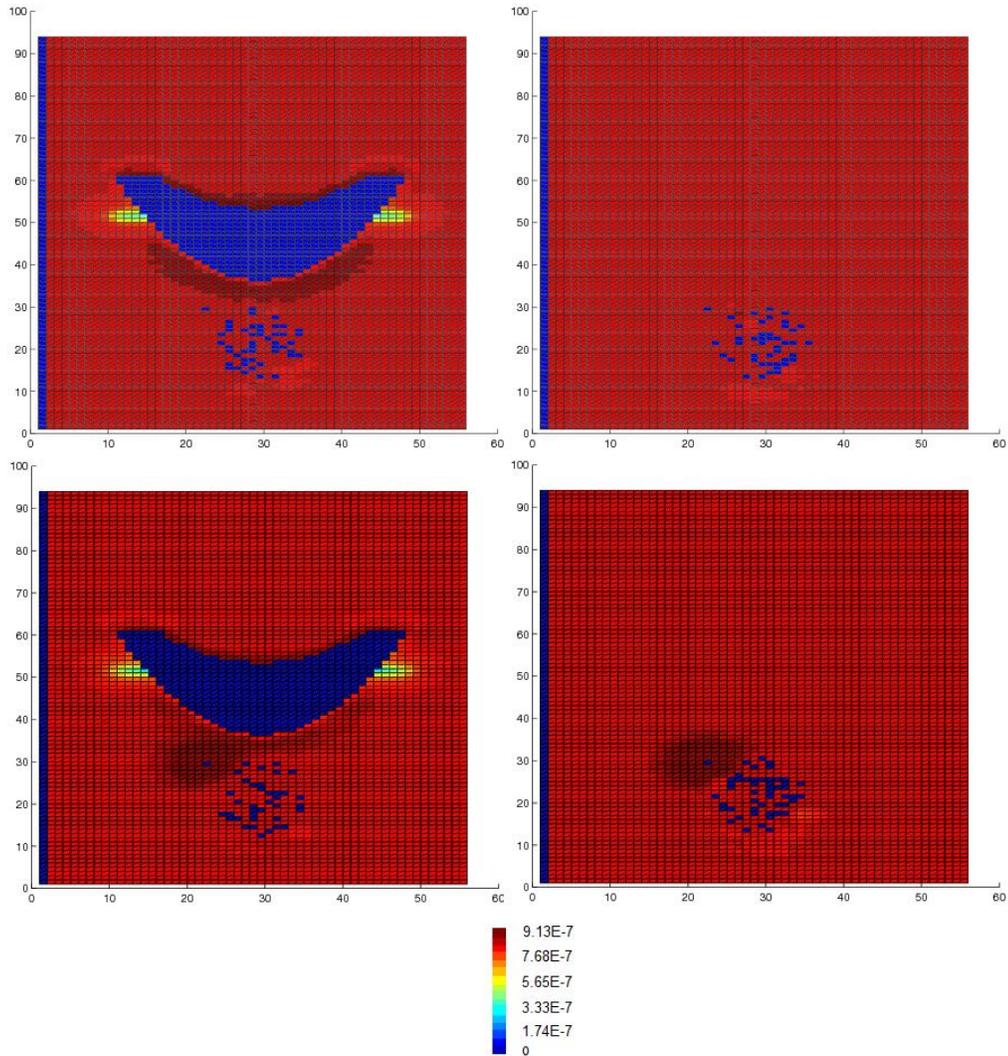


Figura 19. Diferentes casos para experimento de conducta dirigida del sistema, de izquierda a derecha las dos primeras gráficas corresponden a la destrucción del tumor con y sin la presencia de la RBC para una liberación real de droga desde el “nanobot” y las dos últimas gráficas corresponden a la destrucción del tumor con y sin la presencia de la RBC para una liberación modificada de droga desde el “nanobot”.

la célula) así como en la región donde se encontraba el tumor. También nótese que en la parte frontal de la RBC se forma una fina “franja” de alta concentración que tiende a acumularse también debido al movimiento de la célula y que se cree es causa de una mayor tasa de destrucción del tumor debido a la advección de la droga hacia el mismo, como se verá a continuación. Esto último es otra de las principales contribuciones de nuestro trabajo pues hasta donde sabemos no ha habido reportes similares. Se hace notar también que los puntos oscuros en la región donde se encontraba el tumor son los remanentes de éste en el plasma saturado con el Paclitaxel.

Punto	Var (%)	Punto	Var (%)
(28,29,21)	0.80	(21,19,21)	-0.88
(27,30,21)	1.87	(22,15,21)	8.50
(26,29,21)	2.34	(23,13,21)	12.83
(25,29,21)	7.49	(24,12,21)	9.75
(24,28,21)	9.59	(25,11,21)	12.99
(23,27,21)	11.10	(26,11,21)	10.30
(22,26,21)	-4.00	(27,10,21)	14.79
(22,25,21)	8.55		

Tabla 1. Coordenadas de los puntos correspondientes al contorno resaltado del tumor como se muestra en la Figura 20. La variación porcentual positiva indica la presencia de una mayor concentración de droga para el caso en que la RBC está presente en el sistema.

Finalmente se muestra a continuación un ejemplo de como es posible “dirigir” el comportamiento del sistema dependiendo de la manipulación de algunos parámetros físicos y realizando modificaciones en el proceso del cual se tiene control, en este caso del proceso de destrucción del tumor. Esto con el fin de resaltar la evolución de un fenómeno específico del sistema bajo las circunstancias internas del mismo creadas por dichas modificaciones.

Para ello se ejecutó el sistema para 30,000 iteraciones pero inicializando el sistema con una concentración de  $8.48 \times 10^{-7}$  um/ul<sup>3</sup>, modificando la liberación de droga desde el “nanobot” de forma que sea notoria la influencia de la liberación de la primera desde dicho “nanobot”. Se seleccionó este intervalo de tiempo (iteraciones) para obtener resultados en base a una evolución suficiente del sistema y obtener así obtener resultados los más confiables posible.

El experimento se dividió en dos partes principales: en la primera se incluye una RBC y se ejecuta el programa para una liberación real de la droga desde el “nanobot”. Para una lattice de  $56 \times 91 \times 56$  esto corresponde a la primera y segunda gráficas de la Figura 19.

En la segunda parte del experimento se realiza la ejecución del programa bajo las mismas condiciones iniciales pero modificando la liberación de droga desde el “nanobot” de forma que sea notoria dicha liberación en el plasma sanguíneo, para así observar su influencia en el proceso de destrucción del tumor. Esto también incluyendo y sin incluir una

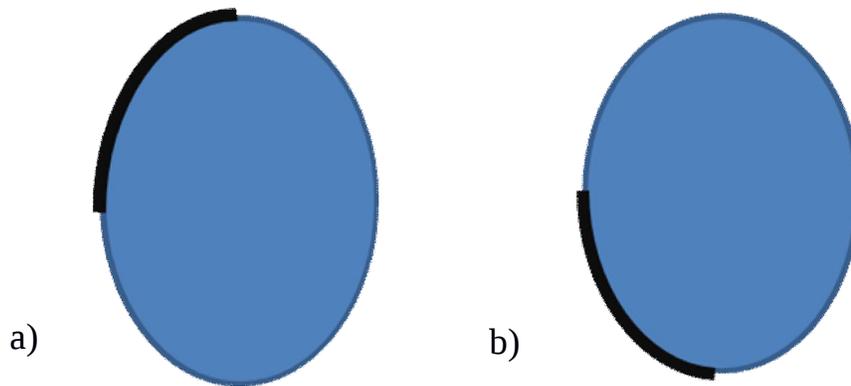


Figura 20. Sección transversal del tumor con forma de semielipsoide de revolución donde los contornos resaltados corresponden al primer par y segundo par de columnas de la Tabla 1, respectivamente. En a) corresponden a los puntos frontales a la RBC, en b) corresponden a los puntos opuestos a la RBC

RBC en el sistema. Lo anterior corresponde a la tercera y cuarta gráficas de la Figura 19.

Adicionalmente se muestra en la Tabla 1 una medida cuantitativa de la diferencia de concentración de droga acumulada alrededor del tumor para el caso de la liberación modificada de la droga desde el “nanobot”, tomando y sin tomar en cuenta la presencia de la RBC para diferentes puntos del mismo tal como se muestra en la Figura 20. Para ello se calcula la variación porcentual de concentración en el tumor (variable etiquetada como “Var” en la Tabla 1) para ambos casos tomando como referencia el caso cuando está presente la RBC. Lo anteriormente mencionado correspondería a la segunda y cuarta gráficas de la Figura 19.

Nótese la gran mayoría de puntos con variación porcentual positiva que indica la mayor presencia de concentración de droga para el caso en que la RBC está presente en el sistema.

De lo anterior se observa que el caso que al menos en teoría presenta más daño al tumor es el caso que corresponde a la tercera gráfica de la Figura 19, comprobándose así como el efecto de advección sobre la concentración causado por el movimiento de la RBC en el plasma y que altera los perfiles de flujo sanguíneo tiene una influencia definitiva en la cantidad de droga que se distribuye alrededor del tumor y que al menos teóricamente afecta la tasa de destrucción del mismo.

## 4.2 Métricas.

La métrica de la paralelización se realizó comparando el tiempo de ejecución del programa ya paralelizado contra el tiempo de ejecución del programa secuencial base para una prueba especialmente diseñada para éste propósito (mismas dimensiones de los componentes del modelo geométrico excepto por el largo del capilar, el cual será de 450  $\mu$ l). Esto se hizo así debido a que con dimensiones menores de la lattice no se aprecia el potencial de la arquitectura combinada, que es cuando sólo se computan nodos de fluido utilizando una organización de la memoria adaptada para tal propósito, consultar referencias (Obrecht, Kuznik, Tourancheau & Roux, 2011), (Mazzeo & Coveney, 2008). Se modificó además el número de iteraciones de modo que el tiempo de ejecución no sobrepasara un límite razonable y representativo, se realizaron cinco mediciones y se obtuvo la media aritmética de las mismas como resultado a comparar. Es necesario señalar que sólo se realizaron pruebas de escalabilidad para el programa paralelizado mediante el estándar MPI únicamente.

Para mostrar el potencial de nuestra arquitectura de paralelización se ejecutaron las pruebas de eficiencia y de escalabilidad mostradas a continuación. Los tamaños de las lattice del MRB se seleccionaron forma que se muestre dicho potencial a pesar de las optimizaciones que aún faltan por implementarse en las diferentes versiones de paralelización del programa computacional.

Las pruebas mencionadas a continuación fueron ejecutadas con un procesador tipo AMD® FX-8150 de 3.61 GHz corriendo bajo la plataforma Windows® 8.1. Se seleccionaron diferentes cantidades de procesadores lógicos dependiendo de la cantidad e recursos disponibles, por ejemplo para el caso del paradigma GPU-MPI se ejecutaron las corridas del programa utilizando sólo dos procesadores lógicos, cada uno controlando una de las tarjetas gráficas mencionadas en el Apéndice A<sup>58</sup>.

---

<sup>58</sup> Aunque se realizó un mínimo balance de carga para la distribución de los nodos de los bloques de datos entre los GPUs dependiendo de su potencia computacional si se escogieron los parámetros de dimensionamiento óptimos de los hilos dependiendo de las especificaciones técnicas de cada una de las tarjetas gráficas utilizadas.

	Corrida 1	Corrida 2	Corrida 3	Corrida 4	Corrida 5	Promedio	Speed up
<b>MPI</b>	1304.954	1302.554	1307.176	1314.670	1307.069	1307.285	1.69
<b>CUDA</b>	675.728	671.936	672.044	676.995	673.723	674.085	3.28
<b>CUDA-MPI</b>	366.992	366.867	366.742	366.899	366.605	366.821	6.02
<b>Secuencial</b>	2193.541	2198.778	2237.429	2204.598	2205.138	2207.897	

Tabla 2. Relación de corridas de versiones, promedio de las mismos para 1000 iteraciones así como el “speed up” alcanzado, los tiempos están en s.

Para la versión del programa paralelizado sólo mediante el estándar MPI se seleccionaron cuatro procesadores lógicos para la ejecución de esta versión del programa asignando cada uno a un procesador físico los cuales son justamente con los que cuenta el CPU mencionado. Al igual que para el caso del paradigma GPU-MPI para éste último caso se estableció la afinidad<sup>59</sup> correspondiente en el administrador de tareas de Windows® de forma que el sistema operativo asigne cada procesador lógico a un procesador físico independiente. Debido a razones implícitas de diseño para el caso de la versión paralelizada sólo mediante el paradigma GPU y la versión secuencial del programa las pruebas se ejecutaron solamente en un procesador lógico.

Es necesario mencionar que todas las versiones de los programas creados tienen la misma funcionalidad y están adaptados para que se ejecuten de manera óptima según su implementación. Para la prueba de eficiencia se ejecutaron 5 corridas de las cuatro versiones creadas para 1000 iteraciones utilizando una lattice de  $56 \times 450 \times 56$  y se promedió el tiempo de ejecución del ciclo principal de cálculo. Los resultados obtenidos se muestran en la Tabla 2.

En la misma Tabla 2 se muestran los valores del “speed up” alcanzado, obteniéndose para estos valores eficiencias de 42.25 % y 301 % para las versiones de los paradigmas MPI y GPU-MPI, respectivamente. Estas eficiencias se obtuvieron en este trabajo dividiendo el “speed up” entre el número de procesadores lógicos seleccionado considerando el cómputo como si sólo dependiera de cada procesador lógico.

<sup>59</sup> Esto se refiere a que el sistema operativo tendrá preferencia en planificar las tareas a realizar en los procesadores lógicos seleccionados, lo cual es una manera aproximada de sólo ejecutar dichas tareas en los mismos. Lo anterior para medir de una manera más exacta el “speed up” alcanzado por las versiones paralelizadas del programa para las varias técnicas de paralelización empleadas respecto al programa secuencial.

Se espera que el nivel de eficiencia obtenido no se aminore aun cuando se ejecuten las versiones con varias partículas suspendidas ya que se optimizará, en una versión posterior, el programa paralelizado mediante CUDA-MPI para que, en el cómputo de los paso de colisión y propagación, sólo se realice trabajo efectivo sobre los nodos de fluido ya que se considera que los nodos de las partículas suspendidas son nodos sin fluido interior, para ello podría considerarse algunos modelos ya implementados, como por ejemplo el mencionado en (Bernaschi, Fatica, Melchionna, Succi & Kaxiras, 2010) y algunas ideas mencionadas en (Zhou, Mo, Wu, Zhao, Rui & Cen, 2012).

Adicionalmente se realizaron pruebas de escalabilidad para una versión del programa computacional ejecutándose sólo mediante el estándar MPI pero modificándolo de forma que cada nodo MPI administre sólo aquellos nodos de las partículas suspendidas que se encuentren dentro del dominio de dicho nodo MPI. Esto tomando las consideraciones pertinentes, por ejemplo si hay nodos de partícula suspendida en nodos MPI vecinos, ya que para el cómputo de la devolución del momento al fluido por parte de uno de los nodos MPI es necesario tomar en cuenta los nodos de partícula de los nodos MPI vecinos, lo que implica la generación de nodos de partícula en una capa adicional del bloque de datos perteneciente al dominio de partición de paralelización por parte del nodo MPI que realiza dicho cómputo.

Aunque pueda parecer compleja esta manera de distribuir la carga de trabajo en nuestra implementación paralelizada, los resultados son alentadores ya que se obtiene un máximo de 70 de “speed up” para 45 procesadores tipo Intel<sup>®</sup> Xeon corriendo bajo un sistema operativo CentOS 7. Lo anterior para una lattice de  $203 \times 225 \times 203$  y que se muestra en la Figura 21. Es importante mencionar la gran diferencia en tiempos de ejecución para esta prueba: 3,668.5 s ( $\approx 1$  hr) para el programa versión MPI corriendo con 5 procesadores, 271.7 s (4.5 min) para el programa versión MPI corriendo con 45 procesadores contra 19,148.15 s (5.3 hrs) para el programa corriendo en versión secuencial.

Se cree que algunas de las posibles razones por las cuales se obtuvo esta eficiencia son:

1. Por que como ya se había anticipado antes aunque se puede optimizar más la arquitectura combinada el performance esperado

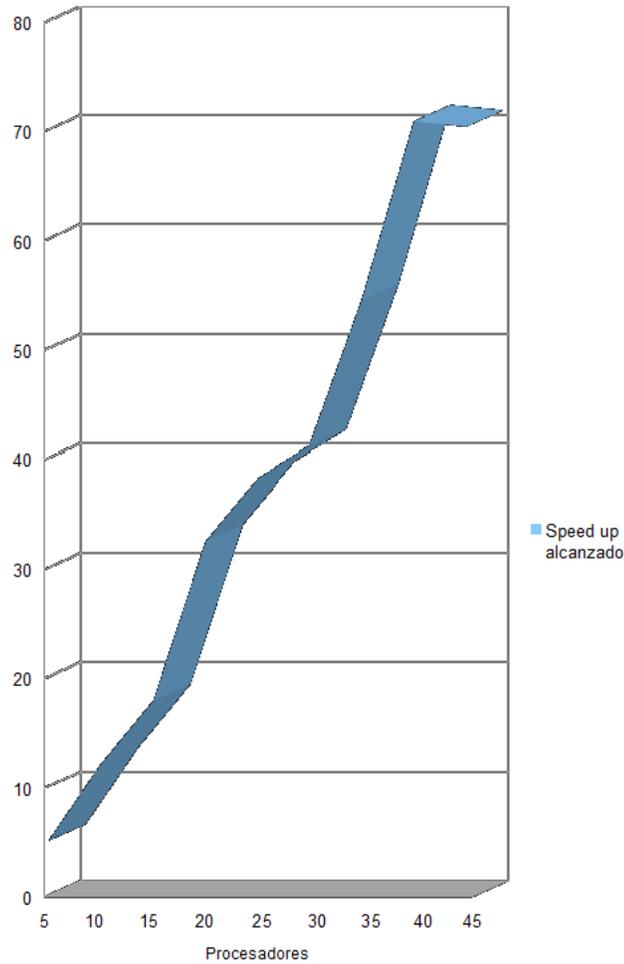


Figura 21. Resultados de escalabilidad del programa versión MPI para una lattice de 203 X 225 X 203.

es bueno y debido al modelo de comunicación utilizado, el cual se trato de mantener los más simple posible y con un tiempo de ejecución  $O(1)$  (véase sección 3.3, Detalle de la Implementación) los poderosos procesadores Intel<sup>®</sup> Xeon dedican la mayor parte de su tiempo de ejecución a realizar los cálculos en sus bloques de datos asignados (dominio) de sólo 5 ul de altura en el eje Y.

2. El hecho de haber logrado que cada nodo MPI sólo generara los nodos de la RBC que se encuentran dentro de su dominio de paralelización (más 2 capas adicionales para la aplicación de las condiciones de frontera sobre la partícula suspendida así como para el cálculo de las componentes cartesianas de la fuerza y la torca ejercidas sobre el centro de masa de la misma) alivió el

cuello de botella computacional que de hecho detectaba la herramienta correspondiente de Visual Studio<sup>®</sup> como la parte del código que más tiempo de proceso del CPU “host” consumía.

Finalmente se indica que aunque en un inicio se estimó la masa del “nanobot” en base a la densidad de partículas poliméricas PLGA mencionada en (Vauthier, Schmidt & Couvreur, 1999), se optó por crear un “nanobot” con una masa de un orden mayor debido a que se presentaban inestabilidades en el método de las redes de Boltzmann para una cantidad de iteraciones mayor de 45,000 iteraciones aproximadamente. Sin embargo esto se espera corregir (aumentar el intervalo de iteraciones donde el sistema está estable) mediante la reincorporación del método de Heún al ciclo principal de cálculo, la inclusión de las modificaciones mencionadas en (Aidun, Lu & Ding, 1998) para la adición de los momentos debido a los nuevos nodos de fluido cubiertos y descubiertos por nodos de las partículas suspendidas debido al movimiento de las mismas entre otras modificaciones y optimizaciones, o incluso la revaloración de partes importantes del modelo conceptual y/o matemático.

### **4.3 Conclusiones.**

- Gracias al método de las redes de Boltzmann es posible simular un sistema biofísico que es útil para aumentar el entendimiento de los complejos fenómenos físicos que suceden al nivel microcirculatorio de un vaso sanguíneo capilar, donde en muchas ocasiones no es posible acceder de manera física en el laboratorio. Gracias a lo anterior es posible sentar las bases para explorar nuevas terapias que una vez perfeccionadas y optimizadas pueden llevar a innovaciones en el campo de la biotecnología que combatan una de las enfermedades mortales más comunes que aquejan hoy en día al mundo occidental como es el cáncer.
- De acuerdo a los resultados obtenidos en las pruebas de eficiencia y escalabilidad se concluye que es posible crear un programa eficiente paralelizado y a bajo costo mediante alguno de los paradigmas disponibles en la actualidad, por ejemplo utilizando una arquitectura combinada compuesta por unidades gráficas de proceso y el estándar

MPI, la cual corriendo bajo una plataforma de programación de uso común como es C++ y teniendo como base hardware de procesamiento convencional y tarjetas gráficas que están prácticamente al alcance de cualquier presupuesto mediano para investigación.

- Aunque en el presente trabajo se utilizó una metodología de programación relativamente simple basada en programación estructurada es muy recomendable una planificación exhaustiva previa si se va a crear un sistema computacional que involucre varios subsistemas (que pueden ser desde algoritmos para simular la deformación de las nanopartículas hasta los mecanismos de acoplamiento entre receptores celulares y las mismas nanopartículas, sin contar con otros fenómenos que se simulen simultáneamente con el flujo sanguíneo, como por ejemplo una reacción química de tipo Michaelis-Menten entre una droga terapéutica y un tumor situado en la pared del vaso sanguíneo, entre varios otros) y/o microsistemas como por ejemplo un autómatas para la destrucción del tumor utilizando el método de Montecarlo para la selección de los nodos a destruir dependiendo de varios parámetros físicos adicionales no considerados en el presente trabajo. Esto debido a que aún cuando el esquema numérico del MRB es en si relativamente simple de implementar sin importar si la geometría para la interacción hidrodinámica sólido-fluido es compleja o si se utiliza una simplificación de la ERB más completa (aunque con un término de multirelajación más complicado), el costo computacional de un proyecto más realista que el presente trabajo y que involucre varios de los subsistemas mencionados puede ser alto debido a que en si el problema del flujo sanguíneo es complejo y si aparte se quiere incorporar una técnica de paralelización para por ejemplo obtener resultados de alta resolución o inclusive respuestas en tiempo real.
- Trabajos como el presente amplía el interés de la comunidad científica involucrada en el desarrollo de herramientas computacionales que colaboren con el área de investigación médica comprometida en combatir los principales males físicos que aquejan al mundo de hoy o simplemente para ampliar nuestro conocimiento en el área de la

microhidrodinámica, la cual promete revelar secretos para colaborar con el mismo fin, mejorar la calidad de vida de la humanidad.

## Apéndice A – Especificaciones del Hardware Utilizado.

### Especificaciones del servidor:

- Procesador AMD<sup>®</sup> FX-8150, 8 procesadores a 3.61 GHz, 64 bits.
- 8 GB de RAM.
- Tarjeta Madre: Crosshair V Formula, Chipset AMD<sup>®</sup> 990 FX/SB950.
- Bus del Sistema de hasta 5200 MT/s Hypertransport<sup>™</sup> 3.0.
- Sistema Operativo: Microsoft<sup>®</sup> Windows<sup>®</sup> 8.1, 64 bits.

### Especificaciones tarjeta gráfica GeForce<sup>®</sup> GTX480:

- 480 Núcleos de CUDA, tecnología Fermi.
- Compute capability: 2.0.
- Versión del driver de CUDA: 5.0, versión de tiempo de ejecución de CUDA: 5.0.
- 15 Multiprocesadores x 32 núcleos de CUDA/multiprocesador.
- Total de memoria constante: 65536 bytes.
- Total de memoria compartida por bloque: 49152 bytes.
- Número total de registros disponibles por bloque: 32768.
- Tamaño del warp: 32.
- Planificador dual de warps.
- Máximo número de bloques por multiprocesador: 8.
- Máximo número de hilos por multiprocesador: 1536.
- Medidas máximas de cada dimensión de un bloque: 1024 x 1024 x 64.

- Medidas máximas de cada dimensión de un “grid” de hilos: 65535 x 65535 x 65535.
- Total de memoria de video (global): 1536 MB GDDR5.
- Ancho de la interfaz de la memoria: 384 bits.
- Ancho de banda de la memoria: 177.4 GB/s.
- Ejecución de “kernels” y transferencia de memoria concurrentes solamente con un motor de copia disponible.
- Ejecución concurrente de “kernels” con mismo contexto de ejecución.
- Soporte de mapeo de memoria no paginable con el “host”.

#### Especificaciones tarjeta gráfica GeForce<sup>®</sup> GTX660Ti:

- 1344 Núcleos de CUDA, tecnología Kepler.
  - Compute capability: 3.0.
  - Versión del driver de CUDA: 5.0, versión de tiempo de ejecución de CUDA: 5.0.
  - 7 Multiprocesadores x 192 núcleos de CUDA/multiprocesador.
  - Total de memoria constante: 65536 bytes.
  - Total de memoria compartida por bloque: 49152 bytes.
  - Número total de registros disponibles por bloque: 65536.
  - Tamaño del “warp”: 32.
  - Máximo número de bloques por multiprocesador: 16.
  - Máximo número de hilos por multiprocesador: 2048.
  - Medidas máximas de cada dimensión de un bloque: 1024 x 1024 x 64.
- Medidas máximas de cada dimensión de un “grid” de hilos:  
2147483647 x 65535 x 65535.
- Total de memoria de video (global): 2048 MB GDDR5.

- Ancho de la interfaz de la memoria: 192 bits.
- Ancho de banda de la memoria: 144.2 GB/s.
- Ejecución de “kernels” y transferencia de memoria concurrentes solamente con un motor de copia disponible.
- Ejecución concurrente de “kernels” con mismo contexto de ejecución.
- Soporte de mapeo de memoria no paginable con el “host”.

Fuente: [W11].

## Apéndice B – Adimensionalización.

Basándose en (Janoscheck, Toschi & Harting, 2010) se adimensionalizó de la siguiente manera:

1. Aunque el autor propone escoger  $\Delta x = \frac{2}{3} \mu\text{m}$  como una buena resolución para este tipo de simulaciones, se escogió un valor de  $0.1667 \mu\text{m}$  para aumentar aún más la resolución del “nanobot” y que al mismo tiempo se conserve un balance razonable entre el tiempo de simulación y de resolución, también se escogió un  $\tau = 2.525$  según se propone en (Ortigoza, 2013) para minimizar dicho tiempo de simulación sin afectar de manera excesiva la estabilidad del MRB.
2. Para calcular  $\Delta t$  se considera que la viscosidad cinemática del plasma sanguíneo es de  $1.2 \times 10^{-6} \text{ m}^2/\text{s}$  (Migliorini et al., 2002), entonces si multiplicamos  $\Delta t / (\Delta x)^2$  por la misma e igualamos con la fórmula adimensional de la viscosidad cinemática del fluido  $\left( \nu = \frac{(2\tau - 1)}{6} \right)$  según (Wolf-Gladrow, 2000), despejando obtenemos  $\Delta t = 1.5625 \times 10^{-8} \text{ s}$  la cual será equivalente a la unidad de tiempo adimensional paso de tiempo o pt. De manera parecida se procede para calcular la masa de las partículas, la velocidad máxima en el centro del capilar, etc.
3. Para determinar el gradiente de presión en el capilar procedemos como se indica en (Sukop & Thorne, 2010) considerando el gradiente de presión como un campo de fuerza que actúa sobre el plasma, a continuación se determina su magnitud mediante la fórmula,

$$F = \frac{8U_{Avg}\nu}{(d/2)^2}$$

- Para tubos cilíndricos en tercera dimensión y despreciando los efectos de longitud de entrada (consultar referencia (Sukop & Thorne, 2010)), donde  $U_{Avg} (= U_{MAX} / 2)$  es la velocidad promedio del fluido en el centro

del tubo (Sears, 1962),  $\nu$  es la viscosidad cinemática del fluido y  $d$  es el diámetro de la sección transversal del tubo. Para nuestro caso con  $U_{MAX} = 3.281 \times 10^{-5}$  ul/pt (de  $U_{MAX} = 7.0 \times 10^{-4}$  m/s, según el modelo conceptual),  $\nu = 0.675$  ul<sup>2</sup>/pt,  $d = 54$  ul y una densidad del fluido de 1 um/uv (adimensionales ambas unidades), obtenemos  $F = -2.43 \times 10^{-7}$  um·ul/pt<sup>2</sup>.

● Expresando lo anterior en unidades del SI para  $\Delta m = 4.6296 \times 10^{-18}$  kg se tiene,  $F = -(2.43 \times 10^{-7} \text{ um}\cdot\text{ul}/\text{pt}^2)(4.6296 \times 10^{-18} \text{ kg})(0.1667 \times 10^{-6} \text{ m}) / (1.5625 \times 10^{-8} \text{ s})^2 = -7.68 \times 10^{-16} \text{ N} = -0.000768 \text{ pN}$ .

● Para determinar el tiempo de relajación a utilizar en el autómata que controla la difusión y advección del agente terapéutico, esto se hace partiendo de la constante de difusión del Paclitaxel (mencionado en el modelo conceptual) adimensionalizando primero dicha cantidad,  $4.2667 \times 10^{-6} \text{ cm}^2/\text{s} (\Delta t / (\Delta x)^2) = 4.2667 \times 10^{-10} \text{ m}^2/\text{s} (1.5625 \times 10^{-8} \text{ s}) / (0.1667 \times 10^{-6} \text{ m})^2 = 2.4 \times 10^{-4} \text{ ul}^2/\text{pt}$ , enseguida se despeja  $\tau_\sigma$  de la fórmula para determinar el coeficiente de difusión del soluto dado en (Sukop & Thorne, 2010) para el enfoque de soluto pasivo,

$$D_\sigma = \frac{1}{3} \left( \tau_\sigma - \frac{1}{2} \right)$$

obteniéndose  $\tau_\sigma = 3(2.4 \times 10^{-4}) + 1/2 = 0.5007$ .

## Apéndice C – Cálculo del Momento de Inercia de la Célula de Sangre Roja.

Se considera un paraboloides de revolución de altura  $H_2$  y radio  $R_2$  situado en un plano Cartesiano con el punto máximo situado en el origen y el eje central orientado a lo largo de uno de los ejes, por ejemplo el eje Y. El paraboloides tiene una cavidad en su base de altura  $H_1$  y radio  $R_1$ . Para determinar el centro de masa a lo largo de dicho eje se divide al paraboloides en rodajas con forma de discos de radio  $r$  y espesor  $dy$ .

De la ecuación del paraboloides,

$$y = kr^2 \quad (c.1)$$

se sustituye  $r = R_2$ ,  $y = H_2$  y se obtiene,

$$k = \frac{H_2}{R_2^2}$$

por lo que,

$$r = \left( \frac{R_2^2}{H_2} y \right)^{1/2} \quad (c.2)$$

Enseguida determinamos el volumen de cada uno de los semidiscos,

$$dV_1 = \pi r^2 dy = \pi \left( \frac{R_2^2}{H_2} y \right) dy$$

entonces el volumen del paraboloides sin considerar la cavidad paraboloidal será,

$$V_1 = \int_0^{H_2} \pi \left( \frac{R_2^2}{H_2} y \right) dy = \frac{1}{2} \pi R_2^2 H_2 \quad (c.3)$$

y considerando la cavidad paraboloidal,

$$V = \frac{1}{2} \pi (R_2^2 H_2 - R_1^2 H_1) \quad (c.4)$$

pero se sabe que la ordenada del centro de masa es,

$$y_{C.M.} = \frac{\int y dV}{\int dV} \quad (c.5)$$

entonces de la ec. (c.3),

$$dV = \pi \left( \frac{R_2^2}{H_2} y \right) dy - \pi \left( \frac{R_1^2}{H_1} (y - (H_2 - H_1)) \right) dy$$

y por lo tanto,

$$\begin{aligned} \int y dV &= \pi \int_0^{H_2} \left( \frac{R_2^2}{H_2} - \frac{R_1^2}{H_1} \right) y^2 dy + \pi \frac{R_1^2}{H_1} \int_{H_2-H_1}^{H_2} (H_2 - H_1) y dy \\ &= \frac{\pi}{3} \cdot \frac{R_2^2 H_2^3}{H_2} + \pi \frac{R_1^2}{H_1} \cdot \left( -\frac{H_2^3 - (H_2 - H_1)^3}{3} + \frac{H_1 (H_2 - H_1)(2H_2 - H_1)}{2} \right) \\ &= \frac{\pi}{3} R_2^2 H_2^2 + \pi \frac{R_1^2}{6} (-6H_2^2 + 6H_2 H_1 - 2H_1^2 + 6H_2^2 - 9H_2 H_1 + 3H_1^2) \\ &= \frac{\pi}{6} (2R_2^2 H_2^2 + R_1^2 (-3H_2 H_1 + H_1^2)) \end{aligned} \quad (c.6)$$

sustituyendo el resultado anterior y  $V$  en la ec.(c.5) se tiene,

$$y_{C.M.} = \frac{\int y dV}{\int dV} = \frac{\frac{\pi}{6} (2R_2^2 H_2^2 + R_1^2 (-3H_2 H_1 + H_1^2))}{\frac{\pi}{2} (R_2^2 H_2 - R_1^2 H_1)} = \frac{1}{3} \cdot \frac{2R_2^2 H_2^2 - R_1^2 H_1 (3H_2 - H_1)}{R_2^2 H_2 - R_1^2 H_1} \quad (c.7)$$

Ahora bien, para determinar el momento de inercia con respecto al eje  $Y$  se determina primero la masa del paraboloides. Si se considera la cavidad paraboloidal, se tiene,

para el semiparaboloides,

$$dm_2 = \rho \pi \left( \frac{R_2^2}{H_2} y \right) dy$$

para la cavidad paraboloidal,

$$dm_1 = \rho \pi \left( \frac{R_1^2}{H_1} y \right) dy$$

entonces la diferencial de masa para la RBC es,

$$dm = dm_2 - dm_1 = \rho\pi \left( \frac{R_2^2}{H_2} - \frac{R_1^2}{H_1} \right) y dy$$

y el momento de inercia respecto al eje Y será la suma de los momentos de inercia respecto a todas las rodajas,

$$\begin{aligned} I_Y &= \int dI_Y = \int \frac{1}{2} r^2 dm = \frac{1}{2} \rho\pi \left( \frac{R_2^4}{H_2^2} \int_0^{H_2} y^2 dy - \frac{R_1^4}{H_1^2} \int_{H_2-H_1}^{H_2} (y - (H_2 - H_1)) y dy \right) \\ &= \frac{1}{2} \rho\pi \left( \frac{R_2^4}{H_2^2} \cdot \frac{H_2^3}{3} - \frac{R_1^4}{H_1^2} (H_2^3 - (H_2 - H_1)^3) \right) \end{aligned} \quad (c.8)$$

Para expresar el resultado anterior en términos de la masa, se tiene de la ec.(c.4),

$$\rho = \frac{2}{\pi} \cdot \frac{M}{R_2^2 H_2 - R_1^2 H_1}$$

sustituyendo el resultado anterior en la ec. (c.8),

$$I_Y = \frac{M}{6} \cdot \frac{2R_2^4 H_2 + R_1^4 (H_1 - 3H_2)}{R_2^2 H_2 - R_1^2 H_1} \quad (c.9)$$

Finalmente para determinar los momentos de inercia respecto a los ejes restantes, se consideran los ejes  $X'$  y  $Z'$  contenidos en el plano de cada rodaja, se tiene entonces por simetría,

$$I_{X'} = I_{Z'}$$

entonces,

$$dI_{X'} = dI_{Z'}$$

de acuerdo al teorema de los ejes perpendiculares,

$$dI_Y = dI_{X'} + dI_{Z'} = 2dI_{X'}$$

$$dI_{X'} = \frac{1}{2} dI_Y = \frac{1}{4} dmr^2$$

aplicando ahora el teorema de Steiner,

$$dI_x = dI_{x'} + dmy^2 = \frac{1}{2}dI_y + dmy^2$$

si integramos la ecuación anterior y sustituyendo el resultado de la ec.(c.8) y  $dm$  se tiene,

$$dI_x = dI_{x'} + dmy^2 = \frac{1}{2}dI_y + dmy^2 \quad (c.10)$$

$$\begin{aligned} I_x &= \frac{1}{24} \rho \pi (2R_2^4 H_2 + R_1^4 (H_1 - 3H_2)) + \rho \pi \left( \frac{R_2^2}{H_2} \int_0^{H_2} y^3 dy - \frac{R_1^2}{H_1} \int_{H_2-H_1}^{H_2} (y - (H_2 - H_1)) y^2 dy \right) \\ &= \frac{1}{24} \rho \pi (2R_2^4 H_2 + R_1^4 (H_1 - 3H_2)) + \rho \pi \left( \frac{R_2^2 H_2^4}{4H_2} - \frac{R_1^2}{H_1} \left( \frac{H_1}{12} \cdot (6H_2^2 H_1 - 4H_2 H_1^2 + H_1^3) \right) \right) \\ &= \frac{1}{24} \rho \pi (2R_2^4 H_2 + R_1^4 (H_1 - 3H_2) + 6R_2^2 H_2^3 - 2R_1^2 (6H_2^2 H_1 - 4H_2 H_1^2 + H_1^3)) \end{aligned} \quad (c.11)$$

expresando la ecuación anterior en términos de  $M$  se tiene,

$$I_x = I_z = \frac{1}{12} M \frac{2R_2^4 H_2 + R_1^4 (H_1 - 3H_2) + 6R_2^2 H_2^3 - 2R_1^2 H_1 (6H_2^2 - 4H_2 H_1 + H_1^2)}{R_2^2 H_2 - R_1^2 H_1} \quad (c.12)$$

Fuente: [W12].

## Bibliografía.

- Ahmadian, M. T., Firoozbakhsh, K., & Hasanian, M. (2012). Simulation of Red Blood Cell Motion in Microvessels Using Modified Moving Particle Semi-Implicit Method. *Scientia Iranica*, **19**, 113-118.
- Aidun, C. K., & Lu, Y. (1995). Lattice Boltzmann Simulation of Solid Particles Suspended in a Fluid. *Journal of Statistical Physics*, **81**, 49-61.
- Aidun, C. K., Lu, Y., & Ding, E.-J. (1998). Direct analysis of particulate suspensions with inertia using the discrete Boltzmann equation. *Journal of Fluid Mechanics*, **373**, 287-311.
- Aldi, N., Buratto, C., Casari, N., Dainese, D., Mazzanti, V., Mollica, F., Munari, E., Occari, M., Pinelli, M., Randi, S., & Spina, P. R. (2017). Experimental and Numerical Analysis of a Non-Newtonian Fluids Processing Pump. *Energy Procedia*, **72<sup>nd</sup> Conference of the Italian Thermal Machines Engineering Association ATI2017, Lecce, Italy, 6-8 September 2017**. **126**, 762-769.
- Artoli, A. M., Hoekstra, A. G., & Sloot, P. M. A. (2006). Mesoscopic Simulations of Systolic Flow in the Human Abdominal Aorta. *Journal of Biomechanics*, **39**, 873-884.
- Axner, L., Bernsdorf, J., Zeiser, T., Lammers, P., Linxweiler, J., & Hoekstra, A. G. (2008). Performance evaluation of a parallel sparse lattice Boltzmann solver. *Journal of Computational Physics*, **227**, 4895-4911.
- Bagchi, P. (2007). Mesoscale Simulation of Blood Flow in Small Vessels. *Biophysical Journal*, **92**, 1858-1877.
- Banik, B. L., Fattahi, P., & Brown, J. L. (2016). Polymeric Nanoparticles: The Future of Nanomedicine. *Wiley Interdisciplinary Reviews: Nanomedicine and Nanobiotechnology*, **8**, 271-299.
- Behrooz, P. (1999). *Introduction to Parallel Processing, Algorithms and Architectures* (1<sup>st</sup> ed.). Kluwer Academic Publishers.
- Bent, J., Hutchings, L. R., Richards, R. W., Gough, T., Spares, R., Coates, P. D., Grillo, I., Harlen, O. G., Read, D. J., Graham, R. S., Likhtman, A. E., Groves, D. J., Nicholson, T. M., & McLeish, T. C. B. (2003). Neutron-Mapping Polymer Flow: Scattering, Flow Visualization, and Molecular Theory. *Science*, **301**, 1691-1695.
- Bernaschi, M., Fatica, M., Melchionna, S., Succi, S., & Kaxiras, E. (2010). A flexible High-Performance Lattice Boltzmann GPU code for the Simulations of Fluid Flows in Complex Geometries. *Concurrency and Computation: Practice and Experience*, **22**, 1-14.
- Bernaschi, M., Melchionna, S., Succi, S., Fyta, M., Kaxiras, E., & Sircar, J. K. (2009). MUPHY: A Parallel MULTI PHYSics/Scale Code for High Performance Bio-Fluidic Simulations. *Computer Physics Communications*, **180**, 1495-1502.

- Bhatnagar, P. L., Gross, E. P., & Krook, M. (1954). A Model for Collision Processes in Gases. I. Small Amplitude Processes in Charged and Neutral One-Component Systems. *Physical Review*, **94**, 511-525.
- Bird, ARMSTRONG, & HASSAGER. (1987). *Dynamics of Polymeric Liquids. Fluid Mechanics Vol 1* (Volume 1 edition ed.). New York: John Wiley & Sons.
- Biferale, L., Mantovani, F., Pivanti, M., Pozzati, F., Sbragaglia, M., Scagliarini, A., Schifano, S., F., Toschi, F., & Tripiccione, R. (2012). A Multi-GPU Implementation of a D2Q37 Lattice Boltzmann Code. *Lecture Notes in Computer Science*, **7203**, 640-650.
- Biswas, O., & Sen, A. (2016). *Nanorobot-the expected ever reliable future asset in diagnosis, treatment and therapy. Foundations and Frontiers in Computer, Communication and Electrical Engineering: Proceedings of the 3rd International Conference C2E2, Mankundu, West Bengal, India, 15<sup>th</sup>-16<sup>th</sup> January, 2016*. (1 edition ed.). (A. Acharyya, Ed.) CRC Press.
- Buxton, G. A. (2008). The Fate of a Polymer Nanoparticle Subject to Flow-Induced Shear Stresses. *Europhysics Letter*, **84**, Article Number 26006.
- Campelo, F., & Hernandez-Machado, A. (5 de 2006). Dynamic model and stationary shapes of fluid vesicles. *The European Physical Journal E*, **20**, 37-45.
- Cattaneo, L., & Zunino, P. (2014). A computational Model of Drug Delivery through Microcirculation to compare Different Tumor Treatments. *International Journal for Numerical Methods in Biomedical Engineering*, **30**, 1347-1371.
- Cercignani, C. (1988). *The boltzmann equation and its applications*. New York : Springer Verlag, c1988.
- Chandran, K. B., Udaykumar, H. S., & Reinhardt, J. M. (Edits.). (2010). *Image-Based Computational Modeling of the Human Circulatory and Pulmonary Systems: Methods and Applications*. New York: Springer.
- Chen, D. T., Wen, Q., Janmey, P. A., Crocker, J. C., & Yodh, A. G. (2010). Rheology of Soft Materials. *Annual Review of Condensed Matter Physics*, **1**, 301-322.
- Cinelli, M., Coles, S. R., Sadik, O., Karn, B., & Kirwan, K. (2016). A Framework of Criteria for the Sustainability Assessment of Nanoproducts. *Journal of Cleaner Production*, **126**, 277-287.
- Clausen, J. R., Reasor, D. A., & Aidun, C. K. (2010). Parallel Performance of a Lattice-Boltzmann/Finite Element Cellular Blood Flow Solver on the IBM Blue Gene/P Architecture. *Computer Physics Communications*, **181**, 1013-1020.
- Creel, C. J., Lovich, M. A., & Edelman, E. R. (2000). Arterial paclitaxel distribution and deposition. *Circulation Research*, **86**, 879-884.
- Cremasco, M. A., & Wang, L. N.-H. (2012). Estimation of partition, free and specific diffusion coefficients of paclitaxel and taxanes in a fixed bed by

- moment analysis: experimental, modeling and simulation studies. *Acta scientiarum. Technology*, **34**, 33-40.
- Crowl, L. M., & Fogelson, A. L. (2010). Computational model of whole blood exhibiting lateral platelet motion induced by red blood cells. *International Journal for Numerical Methods in Biomedical Engineering*, **26**, 471-487.
- Culler, D. E., Singh, J. P., & Gupta, A. (1999). *Parallel Computer Architecture: A Hardware/Software Approach*. Morgan Kaufmann Publishers, Incorporated.
- Darema, F. (2001). *SPMD model: past, present and future, Recent Advances in Parallel Virtual Machine and Message Passing Interface: 8th European PVM/MPI Users' Group Meeting, Santorini/Thera, Greece, September 23-26, 2001*. Journal of Statistical Physics.
- Ding, E.-J., & Aidun, C. K. (2003). Extension of the Lattice-Boltzmann Method for Direct Simulation of Suspended Particles Near Contact. *Journal of Statistical Physics*, **112**, 685-708.
- Djohari, H., & Dormidontova, E. E. (2009). Kinetics of Nanoparticle Targeting by Dissipative Particle Dynamics Simulations. *Biomacromolecules*, **10**, 3089-3097.
- Doddi, S. K., & Bagchi, P. (2009). Three-dimensional computational modeling of multiple deformable cells flowing in microvessels. *Physical Review E*, **79**, 046318.
- Donega, C. d. (Ed.). (2014). *Nanoparticles, Workhorses of Nanoscience*. Springer.
- Dupin, M. M., Halliday, I., Care, C. M., Alboul, L., & Munn, L. L. (2007, 6). Modeling the flow of dense suspensions of deformable particles in three dimensions. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, **75**, 066707.
- Fåhræus, R., & Lindqvist, T. (1931). The Viscosity of the Blood in Narrow Capillary Tubes. *American Journal of Physiology -- Legacy Content*, **96**, 562-568.
- Fedosov, D. A., Dao, M., Em Karniadakis, G., & Suresh, S. (2014). Computational Biorheology of Human Blood Flow in Health and Disease. *Annals of Biomedical Engineering*, **42**, 368-387.
- Fedosov, D. A., Noguchi, H., & Gompper, G. (2014). Multiscale Modeling of Blood Flow: From Single Cells to Blood Rheology. *Biomechanic Modelling Mechanobiology*, **13**, 239-258.
- Felicetti, L., Femminella, M., Reali, G., & Liò, P. (2016). Applications of Molecular Communications to Medicine: A survey. *Nano Communication Networks*, **7**, 27-45.
- Fontana, G., Licciardi, M., Mansueto, S., Schillaci, D. & Giammona, G. (2001). Amoxicillin-Loaded Polyethylcyanoacrylate Nanoparticles: Influence of PEG Coating on the Particle Size, Drug Release Rate and Phagocytic Uptake. *Biomaterials*, **22**, 2857-2865.
- Freund, J. B. (2014). Numerical Simulations of Flowing Blood Cells. *Annual Review of Fluid Mechanics*, **46**, 67-95.

- Gekle, S. (2016). Strongly Accelerated Margination of Active Particles in Blood Flow. *Biophysical Journal*, **110**, 514-520.
- Gohil, T., McGregor, R. H., Szczerba, D., Burckhardt, K., Muralidhar, K., & Székely, G. (2011). Simulation of oscillatory flow in an aortic bifurcation using FVM and FEM: A comparative study of implementation strategies. *International Journal for Numerical Methods in Fluids*, **66**, 1037-1067.
- Grobler, A., Sierra, Z. P., & Viljoen, H. J. (2016). Modelling Nanoparticle Delivery of TB Drugs to Granulomas. *Journal of Theoretical Biology*, **388**, 85-95.
- Hournkumnuard, K., Dolwithayakul, C., & Chantrapornchai, C. (2013). *Parallel Simulation of Magnetic Targeting of Nano-Carriers in Capillary using OpenMP and MPI*. 2013 International Computer Science and Engineering Conference (ICSEC), Nakorn Pathom, 4<sup>th</sup>-6<sup>th</sup> September, 2013, 47-52.
- Resnick, R., Halliday, D., & Kenneth, S. K. (2002). *Física Volumen 1* (4<sup>a</sup> ed. En español). Compañía Editorial Continental.
- He, X., & Luo, L.-S. (1997). Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. *Physical Review E*, **56**, 6811-6817.
- Helfrich, W. (1973). Elastic properties of lipid bilayers: theory and possible experiments. *Zeitschrift Fur Naturforschung. Teil C: Biochemie, Biophysik, Biologie, Virologie*, **28**, 693-703.
- Hong, P.-Y., Huang, L.-M., Lin, L.-S., & Lin, C.-A. (2015). Scalable Multi-Relaxation-Time Lattice Boltzmann Simulations on Multi-GPU Cluster. *Computers and Fluids*, **110**, 1-8.
- Hossain, S. S., Hossainy, S. F., Bazilevs, Y., Calo, V. M., & Hughes, T. J. (2012). Mathematical modeling of coupled drug and drug-encapsulated nanoparticle transport in patient-specific coronary artery walls. *Computational Mechanics*, **49**, 213-242.
- Hwang, C.-W., & Edelman, E. R. (2002). Arterial Ultrastructure Influences Transport of Locally Delivered Drugs. *Circulation Research*, **90**, 826-832.
- Janoschek, F., Toschi, F., & Harting, J. (2010). Simplified Particulate Model for Coarse-Grained Hemodynamics Simulations. *Physical Review E*, **82**, 056710.
- Jendrejack, R. M., Pablo, J. J., & Graham, M. D. (2002). Stochastic simulations of DNA in flow: Dynamics and the effects of hydrodynamic interactions. *The Journal of Chemical Physics*, **116**, 7752-7759.
- Jo, H., Park, Y., Eunkyung Kim, S., & Lee, H. (2016). Exploring the Intellectual Structure of Nanoscience and Nanotechnology: Journal Citation Network Analysis. *Journal of Nanoparticle Research*, **18**, 21.
- Jonathan B., F. (2014). Numerical Simulation of Flowing Blood Cells. *Annual Review of Fluid Mechanics*, **46**, 67-95.
- Kirk, D. V., Hwu, & W. W. (2013). *Programming Massively Parallel Processors, A Hands-on Approach* (2<sup>nd</sup> ed.). Elsevier, Incorporated.
- Koumoutsakos, P., Pivkin, I., & Milde, F. (2013). The Fluid Mechanics of Cancer and Its Therapy. *Annual Review of Fluid Mechanics*, **45**, 325-355.

- Koulouzis, S., Cushing, R., Karasavvas, K. A., Belloum, A., & Bubak, M. (2012). Enabling Web Services to Consume and Produce Large Datasets. *IEEE Internet Computing, Institute of Electrical and Electronic Engineers*, *16*, 52-60.
- Krüger, T., Varnik, F., & Raabe, D. (2011). Efficient and accurate simulations of deformable particles immersed in a fluid using a combined immersed boundary lattice Boltzmann finite element method. *Computers & Mathematics with Applications*, *61*, 3485-3505.
- Ladd, A. J., & Verberg, R. (2001). Lattice-Boltzmann Simulations of Particle-Fluid Suspensions. *Journal of Statistical Physics*, *104*, 1191-1251.
- Lauralee, S. (2011). *Fisiología Humana, de las Células a los Sistemas*. Cengage Learning.
- Laurendeau, N. M. (2005). *Statistical Thermodynamics, Fundamental and Applications*. Cambridge University Press.
- Lázaro, G. R., Hernández-Machado, A., & Pagonabarraga, I. (2014). Rheology of red blood cells under flow in highly confined microchannels. II. Effect of focusing and confinement. *Soft Matter*, *10*, 7207-7217.
- Lázaro, G. R., Hernández-Machado, A., & Pagonabarraga, I. (2014). Rheology of red blood cells under flow in highly confined microchannels: I. effect of elasticity. *Soft Matter*, *10*, 7195-7206.
- NVIDIA Corporation. (2014). *CUDA C Programming Guide v. 6.5*.
- MacMECCAN, R. M., Clausen, J. R., Neitzel, G. P., & Aidun, C. K. (2009). Simulating deformable particle suspensions using a coupled lattice-Boltzmann and finite-element method. *Journal of Fluid Mechanics*, *618*, 13-39.
- Mazzeo, M. D., & Coveney, P. V. (2008). HemeLB: A high performance parallel lattice-Boltzmann code for large scale fluid flow in complex geometries. *Computer Physics Communications*, *178*, 894-914.
- McNamara, G. R., & Zanetti, G. (1988). Use of the Boltzmann Equation to Simulate Lattice-Gas Automata. *Physical Review Letters*, *61*, 2332-2335.
- McWhirter, J. L., Noguchi, H., & Gompper, G. (2009). Flow-induced clustering and alignment of vesicles and red blood cells in microcapillaries. *Proceedings of the National Academy of Sciences*, *106*, 6039-6043.
- Melchionna, S. (2011). A model for red blood cells in simulations of large-scale blood flows. *Macromolecular Theory and Simulations*, *20*, 548-561.
- Migliorini, C., Quian, Y., Chen, H., Brown, E. B., Jain, R. K. & Munn, L. L. (2002). Red Blood Cells Augment Leukocyte Rolling in a Virtual Blood Vessel. *Biophysical Journal*, *83*, 1834-1841.
- Moreno, N., Vignal, P., Li, J., & Calo, V. M. (2013). Multiscale Modeling of Blood Flow: Coupling Finite Elements with Smoothed Dissipative Particle Dynamics. *Procedia Computer Science*, *18*, 2565-2574.
- Mouffouk, F., Challa, D. P., Alrefae, T., & Sini, M. (2017). Modelling and Simulation of the Motion of Nanoparticles in Cylindrical Capillaries allowing

- Particle-to-Wall Interactions. *Mathematical Methods in the Applied Sciences*, **40**, 3111-3128.
- Müller, K., Fedosov, D. A., & Gompper, G. (2014). Margination of micro- and nano-particles in blood flow and its effect on drug delivery. *Scientific Reports*, **4**, 4871.
- Narayanan, S., Binulal, N. S., Mony, U., Manzoor, K., Nair, S., & Menon, D. (2010). Folate targeted polymeric 'green' nanotherapy for cancer. *Nanotechnology*, **21**, 285107.
- O'Connor, J. (2016). Computational Fluid Dynamics in the Microcirculation and Microfluidics – What Role can the Lattice Boltzmann Method Play? *Integrative Biology*, **8**, 589-602.
- Ortigoza Dominguez, R. F. (2013). *Paralelización del Algoritmo de Retículos de Boltzmann en el Estudio de la Dinámica de una Célula de Sangre Roja en una Arteriola*. Universidad Nacional Autónoma de México, Ciudad Universitaria, D.F., México.
- Pacheco, P. S. (1996). *Parallel Programming with MPI* (1<sup>st</sup> ed.). Morgan Kaufman.
- Pan, C., Prins, J. F., & Miller, C. T. (2004). A high-performance lattice Boltzmann implementation to model flow in porous media. *Computer Physics Communications*, **158**, 89-105.
- Pathria, R. K., Beale, Paul D. (2011). *Statistical Mechanics*, Elsevier.
- Peng, Z., Li, X., Pivkin, I. V., Dao, M., Karniadakis, G. E., & Suresh, S. (2013). Lipid bilayer and cytoskeletal interactions in a red blood cell. *Proceedings of the National Academy of Sciences*, **110**, 13356-13361.
- Pozrikidis, C. (2005). Axisymmetric motion of a file of red blood cells through capillaries. *Physics of Fluids*, **17**, 031503.
- Prasad Rao, J., & Geckeler, K. E. (2011). Polymer Nanoparticles: Preparation Techniques and Size-Control Parameters. *Progress in Polymer Science*, **36**, 887-913.
- Premnath, K. N., & Abraham, J. (2007). Three-Dimensional Multi-Relaxation Time (MRT) Lattice-Boltzmann Models for Multiphase Flow. *Journal of Computational Physics*, **224**, 539-559.
- Qian, Y. H., D'Humières, D., & Lallemand, P. (1992). Lattice BGK Models for Navier-Stokes Equation. *EPL (Europhysics Letters)*, **17**, 479.
- Quinn, M. J. (2003). *Parallel Programming in C with MPI and OpenMP* (1<sup>st</sup> ed.). Mc Graw Hill Education.
- Rodríguez-Romo, S., Ortigoza-Dominguez, R. F. (2019). Model and Simulation of Paclitaxel Released Performed by a Nanobot Inside a Capillary, *Physica A: Statistical Mechanics and its Applications*, **518**, 190-209.
- Röhm, D. (2011). *Lattice Boltzmann Simulations on GPUs*. Diploma Thesis, Universität Stuttgart. Institute for Computational Physics, Stuttgart.
- Román Román, E. I. (2007). *Estudio de Plumas Térmicas usando la Técnica de la Ecuación de Boltzmann en Redes*. Tesis de Maestría en Ingeniería, Universidad Nacional Autónoma de México, México.

- Rowhanimanesh, A., & Akbarzadeh Totonchi, M. R. (2013). Autonomous Drug-Encapsulated Nanoparticles: Towards a Novel Non-Invasive Approach to Prevent. *Iranian Journal of Medical Physics*, **10**, 118-132.
- Rubinstein, M., & Colby, R. H. (2003). *Polymer Physics* (1 edition ed.). Oxford ; New York: Oxford University Press.
- Sahoo, S. K., Parveen, S., & Panda, J. J. (2007). The Present and Future of Nanotechnology in Human Health Care. *Nanomedicine: Nanotechnology, Biology and Medicine*, **3**, 20-31.
- Sears, F. W. (1962). *Mecánica, Calor y Sonido*. Madrid, Spain: Ed. Aguilar.
- Shan, X., & Chen, H. (1993). Lattice Boltzmann model for simulating flows with multiple phases and components. *Physical Review E*, **47**, 1815-1819.
- Shen, Z., Coupier, G., Kaoui, B., Polack, B., Harting, J., Misbah, C., & Podgorski, T. (2016). Inversion of hematocrit partition at microfluidic bifurcations. *Microvascular Research*, **105**, 40-46.
- Shi, X., Wang, S., & Zhang, S. (2013). Numerical simulation of the transient shape of the red blood cell in microcapillary flow. *Journal of Fluids and Structures*, **36**, 174-183.
- Shi, X., Zhang, S., & Wang, S.-l. (2014). Numerical simulation of hemodynamic interactions of red blood cells in micro-capillary flow. *Journal of Hydrodynamics, Ser. B*, **26**, 178-186.
- Silbernagl, S., & Despopoulos, A. (2011). *Fisiología, Texto y Atlas*. Ed. Médica Panamericana.
- Soudah, E., Ng, E. Y., Loong, T. H., Bordone, M., Pua, U., & Narayanan, S. (2013). CFD Modelling of Abdominal Aortic Aneurysm on Hemodynamic Loads Using a Realistic Geometry with CT. *Computational and Mathematical Methods in Medicine*.
- Stensholt, S. (2009). *Lattice Boltzmann Methods for use in Irregular Channels, Theoretical Foundations, Simulation and Working Code*. Verlag Dr. Müller.
- Succi, S. (2001). *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*. Oxford : New York: OUP Oxford.
- Sukop, M. C., & Thorne, D. T. (2010). *Lattice Boltzmann Modeling: An Introduction for Geoscientists and Engineers* (1<sup>st</sup> ed.). Springer Publishing Company, Incorporated.
- Tan, J., Thomas, A., & Liu, Y. (2011). Influence of Red Blood Cells on Nanoparticle Targeted Delivery in Microcirculation. *Soft matter*, **8**, 1934-1946.
- Tirado-Ramos, A., Tsouloupas, G., Dikaiakos, M., & Sloot, P. (2005). Grid Resource Selection by Application Benchmarking for Computational Haemodynamics Applications. *International Conference on Computational Science*, 534-543.
- Vahala, G., Keating, B., Soe, M., Yopez, J., Vahala, L., Carter, J., & Ziegeler, S. (2008). MHD turbulence studies using lattice Boltzmann algorithms. *Communications in Computational Physics*, **4**, 624-646.

- Vance, M. E., Kuiken, T., Vejerano, E. P., McGinnis, S. P., Hochella Jr., M. F., Rejeski, D., & Hull, M. S. (2015). Nanotechnology in the Real World: Redeveloping the Nanomaterial Consumer Products Inventory. *Beilstein Journal of Nanotechnology*, **6**, 1769-1780.
- Vautier, C., Schmidt, C., & Couvreur, P. (1999). Measurement of the Density of Polymeric Nanoparticulate Drug Carriers by Isopycnic Centrifugation. *Journal of Nanoparticle Research*, **1**, 411-418.
- Velázquez Ortega, J. L. (2009). *Simulación de Fluidos no Newtonianos en Presencia de Medios Porosos*. Tesis Doctoral en Ciencias, Universidad Nacional Autónoma de México, México.
- Ververg, R., Alexeev, A., & Balaazs, A. C. (2006). Modeling the Release of Nanoparticles from Mobile Microcapsules. *The Journal of Chemical Physics*, **125**.
- Winkler, R. G., Fedosov, D. A., & Gompper, G. (2014). Dynamical and rheological properties of soft colloid suspensions. *Current Opinion in Colloid & Interface Science*, **19**, 594-610.
- Wolf-Gladrow, D. A. (2000). *Lattice-Gas Cellular Automata and Lattice Boltzmann Models: an Introduction*. New York: Springer Verlag.
- Xiong, Q., Li, B., Xu, J., Fang, X., Wang, X., Wang, L., XiangFeng, H., & Ge, W. (2012). Efficient Parallel Implementation of the Lattice Boltzmann Method on Large Clusters of Graphic Processing Units. *Chinese Science Bulletin*, **57**, 707-715.
- Yin, X., Thomas, T., & Zhang, J. (2013). Multiple red blood cell flows through microvascular bifurcations: Cell free layer, cell trajectory, and hematocrit separation. *Microvascular Research*, **89**, 47-56.
- Zeiser, T., Hager, G., & Wellein, G. (2009). Vector Computers in a World of Commodity Clusters, Massively Parallel Systems and Many-Core Many-Threaded CPUs: Recent Experience Based on an Advanced Lattice Boltzmann Flow Solver, in: Wolfgang, D. B. K., Nagel, E., Resh, M. M. (Eds.). *High Performance Computer in Science and Engineering '08*, 333-347.
- Zhou, H., Mo, G., Wu, F., Zhao, J., Rui, M., & Cen, K. (2012). GPU Implementation of Lattice Boltzmann Method for Flows with Curved Boundaries. *Computer Methods in Applied Mechanics and Engineering*, **225-228**, 65-73.

## Páginas Web Consultadas.

- [W1]. <http://www.rae.es/>, fecha de consulta: 13 de diciembre de 2018.
- [W2]. [http://sgpwe.izt.uam.mx/files/users/uami/cfmc/Bioquimica\\_III/in\\_vivo\\_e\\_in\\_vitro.pdf](http://sgpwe.izt.uam.mx/files/users/uami/cfmc/Bioquimica_III/in_vivo_e_in_vitro.pdf), fecha de consulta: 28 de enero de 2017.
- [W3]. <http://ocw.uv.es/ciencias/2/1-2/112733mats70.pdf>, fecha de consulta: 13 de diciembre de 2018.
- [W4]. <http://www.mpich.org/documentation/guides/>, fecha de consulta: 13 de diciembre de 2018.
- [W5]. <https://computing.llnl.gov/tutorials/mpi/>, fecha de consulta: 28 de enero de 2017.
- [W6]. <http://www.mcs.anl.gov/research/projects/mpi/>, fecha de consulta: 28 de enero de 2017.
- [W7]. [https://technet.microsoft.com/en-us/library/ee783547\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/ee783547(v=ws.10).aspx), fecha de consulta: 28 de enero de 2017.
- [W8]. <https://www.open-mpi.org/>, fecha de consulta: 13 de diciembre de 2018.
- [W9]. <https://en.wikipedia.org/wiki/SOAP>, fecha de consulta: 28 de enero de 2017.
- [W10]. <https://www.w3.org/TR/soap12-part1/>, fecha de consulta: 28 de enero de 2017.
- [W11]. <http://www.nvidia.com/page/home.html>, fecha de consulta: 28 de enero de 2017.
- [W12]. [http://personales.unican.es/junqueraj/JavierJunquera\\_files/Fisica-1/momentos\\_de\\_inercia.pdf](http://personales.unican.es/junqueraj/JavierJunquera_files/Fisica-1/momentos_de_inercia.pdf), fecha de consulta: 16 de marzo de 2017.