

UNIVERSIDAD ANTONIO NARIÑO

MAESTRÍA EN INGENIERÍA DE BIOPROCESOS

FACULTAD DE INGENIERÍA AMBIENTAL Y CIVIL

Fabián Camilo Otálora Tapiero

Diseño de la metodología para la integración de herramientas Open Source aplicadas a la evaluación del comportamiento hidrodinámico de biorreactores usando CFD: Caso de estudio Reactor Metanogénico Escalonado RME

Bogotá D.C.

2021

MAESTRÍA EN INGENIERÍA DE BIOPROCESOS

Fabián Camilo Otálora Tapiero

Diseño de la metodología para la integración de herramientas Open Source aplicadas a la evaluación del comportamiento hidrodinámico de biorreactores usando CFD: Caso de estudio Reactor Metanogénico Escalonado RME

Tesis presentada al programa de Maestría en ingeniería de bioprocesos de la Universidad Antonio Nariño para optar por el título de magister en ingeniería de bioprocesos

Orientador: Prof, PhD. Héctor Javier Luna Wandurraga - UAN

Co-Orientador: Prof, PhD. Bruno Eduardo Lobo Baeta - UFOP

Bogotá D.C.

2021

Tabla de contenido

Lista de imágenes	4
RESUMEN.....	5
ABSTRACT	5
1. INTRODUCCIÓN	6
2. REVISIÓN BIBLIOGRÁFICA.....	8
3. OBJETIVOS	16
3.1. Objetivo general.....	16
3.2. Objetivos específicos	16
4. METODOLOGÍA.....	16
4.1. Mallado	16
4.1.1. Validación de software para mallado	17
4.1.2. Blender	18
4.2. Corriendo la modelación	28
4.2.1. Elección software de modelación	28
4.2.2. Condiciones del solucionador	30
4.2.3. Corrida.....	35
4.2.4. Pos procesamiento.....	37
4.3. Flujo de trabajo	38
4.3.1. Biorreactor RME	38
5. RESULTADOS Y DISCUSIÓN.....	39
5.1. Mallado y pre-procesamiento	39
5.2. Parámetros de la modelación y software de modelación	43
5.3. Pos procesamiento y flujo de trabajo.....	45
5.4. Aporte social	46
6. CONCLUSIONES.....	47
7. SUGERENCIAS PARA FUTURAS INVESTIGACIONES.....	48
8. BIBLIOGRAFIA	48
ANEXOS.....	54
Velocidad	55
Densidad:.....	56

blockMeshDict- Geometría:	57
controlDict	67

Lista de imagenes

Imagen 1: Ecuación de búsqueda bibliográfica.....	10
Imagen 2. Mapa de términos generado a partir de VOSviewer.....	10
Imagen 3. Autores destacados en la investigación de CFD bajo la ecuación de búsqueda aplicada.....	11
Imagen 4. Página de inicio de Blender (GUI).....	21
Imagen 5. Familiarización con Blender.....	22
Imagen 6. Proceso de extrusión y escalado para la construcción de la geometría.	23
Imagen 7. Vistas del reactor construido.....	24
Imagen 8. Asignación de características a las caras del biorreactor.....	24
Imagen 9. Número de celdas y exportación de la geometría.	25
Imagen 10. Script de características de la malla.	26
Imagen 11. Validación del proceso de mallado.....	27
Imagen 12 Flujo de carpetas de configuración de la modelación.....	35
Imagen 13. Script de controlDict.....	36
Imagen 14. Validación de la geometría en pos-procesamiento.....	37
Imagen 15. Esquema de flujo de trabajo desarrollado.	38
Imagen 16. Biorreactor en estudio, ilustrado y real.....	39
Imagen 17. Distribución de las celdas en la geometría del biorreactor.....	43
Imagen 18. Resultado de la modelación: Visualización del llenado del tanque	46
Ilustración 1. Geometría del biorreactor en estudio.....	19
Ilustración 2. Interfaz gráfica de usuario de SwiftBlock	20
Tabla 1. Propiedades de software de diseño de geometría.....	18
Tabla 2. Parámetros del modelo	27
Tabla 3. Principales ecuaciones de la solución del solver.	32
Tabla 4. Elección del Solver.	34
Tabla 5. Comandos de manejo de OpenFOAM.	36
Tabla 6. Resultados de implementación por otros autores de Blender a CFD.	41

RESUMEN

En este trabajo encontrará una propuesta de flujo de trabajo que integra herramientas de código abierto y colaborativo, específicamente Blender, OpenFoam y paraView para modelar el comportamiento hidrodinámico de biorreactores, la validación del flujo de trabajo creado se realizó con un biorreactor metanogenico escalonado que ha sido trabajado con anterioridad a escala laboratorio y desarrollado en alianza de investigación entre la UAN y la UFOP. El planteamiento y la comprensión de las herramientas fue realizado bajo la guía de ejemplos prácticos y finalmente el acople a un caso de estudio, con base en la búsqueda bibliográfica encontramos que este es un novedoso trabajo para el área de modelación hidrodinámica en el contexto de los bioprocesos. Por último se concluye en la viabilidad de aplicar el esquema aquí propuesto para incrementar y facilitar el desarrollo de bioprocesos y las perspectivas de automatizar las simulaciones en investigaciones futuras creando scripts de Python e incluyendo análisis en herramientas como R Studio y Power Bi que complementan todo un portafolio Open Source, que en comparación con software licenciado representa una elevada inversión en licenciamiento y mantenimiento de los software actualizados.

Palabras Clave: Dinámica computacional de Fluidos, CFD, OpenFOAM, Blender, bioprocesos, biorreactores, modelación, simulación, comportamiento hidrodinámico, flujo de trabajo.

ABSTRACT

In this work you will find a workflow proposal that integrates collaborative and open source tools, specifically Blender, OpenFoam and paraView to model the hydrodynamic behavior of bioreactors, the validation of the created workflow was carried out with a stepped methanogenic bioreactor that has been previously worked on a laboratory scale and developed in a research alliance between UAN and UFOP. The approach and understanding of the tools was carried out under the guidance of practical examples and finally the coupling to a case study, based on the bibliographic search we found that this is a novel work for the area of hydrodynamic modeling in the context of the bioprocesses. Finally, it is concluded on the feasibility of applying the scheme proposed here to increase and facilitate the development of bioprocesses and the prospects of automating simulations in future research by creating Python scripts and including analysis in tools such as R Studio and Power Bi that complement a whole Open Source portfolio, which in comparison with licensed software represents a high investment in licensing and maintenance of updated software.

Keywords: Computational Fluid Dynamics, CFD, OpenFOAM, Blender, paraView, bioprocesses, bioreactors, model, simulation, hydrodynamic behavior, workflow.

1. INTRODUCCIÓN

En la última década los bioprocesos han tenido un auge notable debido a la diversificación en sus campos de aplicación, la construcción del conocimiento va alcanzando un nivel de madurez que permite su uso con elevados rangos de confiabilidad en diferentes áreas industriales, grandes centros urbanos y comunidades rurales. El enfoque orientado hacia el tratamiento de residuos se ha diversificado y conceptos como biorrefinerías entraron en juego permitiendo asegurar el saneamiento de efluentes con cargas contaminantes y la obtención de bioproductos como biocombustibles, energía, calor y productos de valor agregado de interés comercial y agrícola (Flores et al., 2019; Gavrilescu, 2014).

La remoción biológica de nutrientes especialmente fósforo y nitrógeno para evitar la eutrofización de cuerpos hídricos es un gran reto especialmente porque el proceso de degradación para cumplir con normatividad requiere en la mayoría de casos tratamiento aerobio que no logra alcanzar tasas de remoción óptimas o una combinación de biotecnologías que consumen elevadas cantidades de energía y requieren amplia disponibilidad de áreas para instalación de equipos y biorreactores junto con el consumo de reactivos si se tienen operaciones unitarias fisicoquímicas en el proceso (Díez-Montero et al., 2015).

La implementación exitosa de los bioprocesos en las biorrefinerías requiere una rigurosa integración de tecnologías, así como es necesario asegurar la menor emisión posible de gases efecto invernadero en los procesos y la presentación de análisis de ciclo de vida que garanticen la mejor ruta de aprovechamiento de residuos y maximización de beneficios (Nizami et al., 2017). Un componente imprescindible dentro de los bioprocesos son los biorreactores que llevan a cabo la descontaminación de efluentes mediante remoción de materia orgánica y compuestos específicos manipulables y de poco rango contaminante,

La Dinámica Computacional de Fluidos (CFD) facilita la evaluación de las configuraciones de biorreactores y sistemas de tratamiento, así como permite simular rendimientos en diferentes geometrías, regímenes de operación, distribuciones físicas, tiempos de retención, patrones de mezcla, perfiles de presión y temperatura en sustratos variados (Allonneau et al., 2015; Blanco-Aguilera, Lara, Barajas, Tejero, & Díez-Montero, 2020; Koerich et al., 2020; Marroquín-Fandiño et al., 2020; Rivas et al., 2020a). A partir de experimentos de menor complejidad como ensayos de trazadores se obtiene la materia prima para modelos de CFD y su validación mediante la distribución del tiempo de residencia hidráulica (RTD) y regímenes de mezcla, adicionalmente, con ciertas sustancias es posible determinar el comportamiento cinético del sistema (Blanco-Aguilera, Lara, Barajas, Tejero, & Díez-Montero, 2020a; Levenspiel, 2012).

La simulación y modelación de bioprocesos se ha perfilado como la solución de mayor eficiencia en términos económicos y de precisión en los resultados ya que permite reproducir virtualmente comportamientos basados en datos confiables y diversos, (Marroquín-Fandiño et al., 2020) obtuvieron avances destacables en el prototipado y desarrollo de equipos de bajo costo que permiten realizar múltiples experimentos de manera simultánea con obtención de datos en línea a escala de mililitros facilitando la implementación de tecnología en zonas apartadas o laboratorios de presupuestos limitados ya que hay menor costo en reactivos.

El propósito de esta implementación tecnológica es lograr un flujo de trabajo de alta eficiencia, para ello es indispensable utilizar los mejores software disponibles en cada etapa de la modelación, el análisis hidrodinámico integrando RTD y CFD ha sido modelado en múltiples software como ANSYS (Marton et al., 2006) o COMSOL (Marroquín-Fandiño et al., 2020) los cuales corresponden a potentes herramientas informáticas pre programadas de acceso limitado dado su elevado costo de adquisición de licencias y necesidad de constante actualización, entre tanto el software OpenFOAM cumple con la misma rigurosidad informática, pero se destaca al ser una herramienta open source que utiliza código abierto y se acopla a otros programas como Blender o Salome para la creación y visualización de mallado y comportamiento dinámico en las etapas de pre-procesamiento y post-procesamiento. (Blanco-Aguilera, Lara, Barajas, Tejero, & Diez-Montero, 2020) trabajaron en el desarrollo de un novedoso reactor multiambiente caracterizado por su capacidad de remover nutrientes de aguas residuales domésticas con un sistema de separación por concentración de oxígeno en el interior del sistema, este desarrollo fue validado en OpenFOAM con datos de comportamiento hidrodinámico obtenidos mediante test de trazadores.

El objetivo de esta tesis fue integrar múltiples herramientas *Open Source* para crear un flujo de trabajo que permita obtener todos los beneficios del análisis de comportamiento hidrodinámico mediante la dinámica computacional de fluidos (CFD) en el área de los biorreactores. Previamente (Otálora-Tapiero, 2018; Pérez Peña & Velandia Blanco, 2020) construyeron y operaron dos biorreactores, uno ASTBR de 1800 ml y un RME de 2900 ml, respectivamente, sobre los cuales se hicieron pruebas de trazadores con el sistema desarrollado por (Marroquín-Fandiño et al., 2020), el propósito de aplicar este flujo de trabajo es validar que sistemas como los dos mencionados anteriormente puedan aprovechar la ventaja de simular una variedad de características físicas y condiciones operativas que brindan un análisis integral del desempeño de los biorreactores antes de finalizar los diseños de sistemas a escala de producción comercial e implementar bioreactores enfocados a residuos específicos o mezclas complejas facilitando las labores de transmisión y apropiación de tecnología en zonas rurales o de recursos limitados así como ayuda a optimizar las aplicaciones industriales (Marcato et al., 2021; Noll & Henkel, 2020).

2. REVISIÓN BIBLIOGRÁFICA

Con el fin de establecer el contexto de la investigación se realizó una búsqueda bibliométrica en la base de datos de Scopus ya que este tipo de estudios son herramientas de alta efectividad para evaluar integralmente la producción científica mediante la interacción de múltiples indicadores, como journals, autores, colaboraciones, países o ubicaciones geográficas, instituciones, palabras clave o conceptos relevantes; a partir de estos marcos de referencia se pueden determinar comportamientos, analizar tendencias y perspectivas de innovación en áreas de interés como en este estudio donde la dinámica computacional de fluidos es el eje de investigación para tomar decisiones enfocadas a la optimización de tiempos y recursos tanto económicos como investigativos (Liu et al., 2019; Saikia et al., 2020). El desarrollo de este tipo de búsquedas se enfoca principalmente en determinaciones cuantitativas, el explorar nuevas tendencias se permite dar un enfoque orientado al análisis cualitativo para determinar la incidencia de estos estudios sobre la evaluación en investigación, desarrollo e innovación (Herther, 2009).

Se utilizó la revisión bibliográfica para obtener ventajas de los desarrollos previos y asegurar suficiencia investigativa en una amplia gama de campos que inciden sobre los objetivos de investigación (Ellegaard & Wallin, 2015) el área de CFD ha sido desarrollada principalmente y con elevada efectividad en aplicaciones de ingeniería mecánica por esto es importante integrar análisis bibliométricos para identificar oportunidades de aplicación y optimización en otras áreas como la biotecnología y bioprocesos. Otro atributo de estos análisis es la posibilidad de inferir el grado de madurez de una tecnología lo cual resulta muy práctico para este estudio y la integración de la información contenida en los textos analizados. Pretorius et al., 2008 plantea la posibilidad de analizar la combinación entre bibliometría y modelos para encontrar tendencias en CFD y ayudar a que el desarrollo tecnológico sea utilizado por investigadores para evaluar los enfoques y perspectivas de la gestión de la tecnológica.

La simulación CFD ha revolucionado por completo la mecánica de fluidos. Hoy en día, el diseño de aeronaves sin costosos prototipos para pruebas, predicción de la viabilidad de campos de aerogeneradores, estudios de turbinas, simulaciones de sistemas de suministro de agua potable y muchas otras aplicaciones son posibles con una computadora de alto rendimiento (Boccardo, Sethi, et al., 2019; Buwa & Ranade, 2002; Kansal et al., 2015; Koerich et al., 2020; Kumawat et al., 2019). Sin embargo, a pesar de las ventajas que ofrece esta técnica, también tiene varias limitaciones a considerar. El software disponible hoy en día permite que la simulación CFD sea cómoda y fácil de usar, con interfaces gráficas elaboradas y una amplia gama de opciones de posprocesamiento (Blender, paraView, etc).

No obstante, esto puede llevar a empresas y usuarios individuales a pensar que el análisis CFD es una herramienta infalible y a utilizarlo de forma despreocupada, ya que una simulación realista solo está en manos de profesionales (Boccardo et al., 2015). Además, un cálculo CFD real es un proceso iterativo, extremadamente cuidadoso y revisado, siempre se

ha apoyado en la medida de lo posible con experimentos y pruebas para confirmar la veracidad del mismo. Actualmente, existe una amplia gama de software para simulaciones CFD, que cubre diferentes campos de la mecánica de fluidos (Mohiuddin & Mathkour, 2015). Por ello, en este capítulo se hará una revisión de las distintas opciones disponibles, analizando sus ventajas y desventajas y sus principales aplicaciones centrándose en las dos más importantes: ANSYS, ampliamente utilizado por las empresas, y OpenFOAM, un software CFD gratuito y de código abierto, paquete, que tiene una capacidad similar pero ventajosa sobre software licenciados.

Este trabajo se enfocó en el uso de herramientas de acceso libre siempre que fuera posible ya que de esta manera se garantiza la divulgación y apropiación de conocimientos. La principal fuente de información fue la base de datos bibliográfica [Scopus](#) ya que es uno de los principales navegadores científicos, desarrollada por Elsevier en el año 2004 y dada su disponibilidad y elevada confiabilidad al integrar múltiples áreas del conocimiento como ciencias sociales, ciencias naturales, artes y humanidades entre muchas otras fue seleccionada para el proceso de recolección de información mediante la construcción de una ecuación de búsqueda. La amplia variedad de documentos que están a disponibles para la consulta garantiza un conjunto robusto de datos para el análisis integrando términos clave, citas, autores y distribuciones geográficas. SciVal es una herramienta desarrollada por Elsevier que integra una serie de módulos para el análisis de datos que pueden exportados de Scopus fue utilizada en este estudio para ayudar a determinar comportamientos de la investigación orientando la toma de decisiones hacia la evidencia en investigación científica. Sus principales módulos son:

- Módulo de vista general (Overview): Indica el aspecto general de la investigación tomando información de las publicaciones, citas y colaboraciones de los documentos exportados.
- Módulo de evaluación comparativa (Benchmarking): Mediante el uso de métricas y estadísticas se realiza una comparación objetiva de la investigación según con respecto a otras instituciones, países, etc.
- Módulo de colaboración (Collaboration): Permite identificar oportunidades de colaboración con pares que estén desarrollando investigaciones en la misma área de interés.
- Módulo de tendencias (Trends): Brinda un panorama general del comportamiento del campo de investigación teniendo en cuenta instituciones, autores, journals y regiones, facilitando la determinación de temas en auge o disminución en investigación en ciertas áreas que han alcanzado su punto de madurez tecnológica o se están convirtiendo en ejes estratégicos.

[VOSviewer](#) fue otra herramienta utilizada en el desarrollo del componente bibliométrico de este estudio, es un programa de acceso libre, desarrollado por Nees Jan van Eck y Ludo Waltman de la Universidad de Leiden, enfocado en analizar y visualizar la literatura

A partir del análisis bibliométrico se identificó que los países con mayores publicaciones bajo la ecuación de búsqueda antes mencionada son Brasil, Alemania, USA, Reino Unido, Canadá, con diversos autores que han creado redes de colaboración internas como se puede observar en la Imagen 3 pero con colaboración externa mínima por lo que se puede plantear que a futuro este campo de investigación seguirá creciendo al plantear colaboraciones e investigaciones conjuntas.

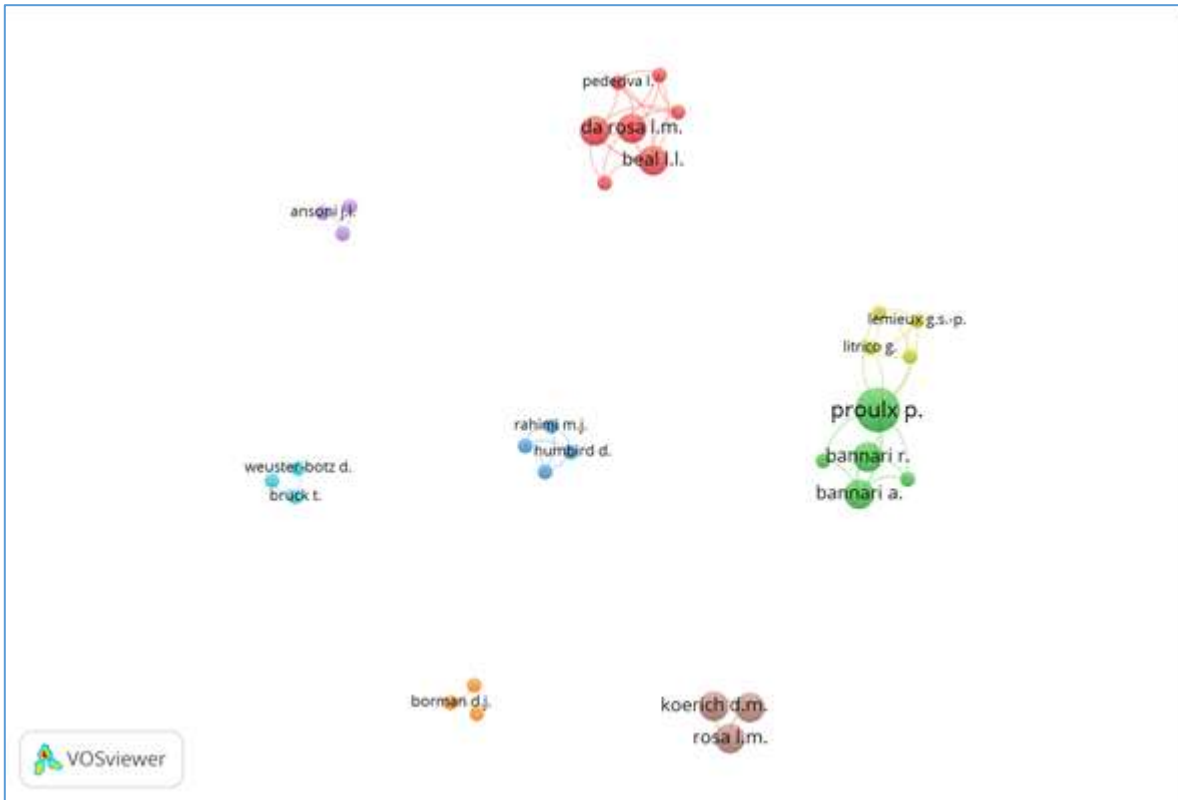


Imagen 3. Autores destacados en la investigación de CFD bajo la ecuación de búsqueda aplicada.

Con la integración de SciVAL se pudo establecer las áreas de influencia del proyecto que son principalmente las ciencias de la computación, matemáticas e ingeniería química pero estudios recientes están aportando al desarrollo de CFD aplicado a ciencias ambientales, bioquímica y algunas ciencias de la rama de la medicina lo cual direcciona una diversificación de la aplicación de la modelación, La investigación en CFD tuvo una fuerte consolidación en los años 90s donde los planteamientos matemáticos previamente desarrollados fueron integrados con las ciencias de la computación y así problemas de ecuaciones complejas sin solución exacta pudieron ser aproximados bajo modelos solucionados por software, posteriormente en los años 2000 se terminaron de perfeccionar software especializados como ANSYS, COMSOL; openFOAM o PALABOS que siguen en constante actualización pero durante la última década esta área de CFD se ha venido diversificando permitiendo el crecimiento y la integración de conceptos base planteados en décadas pasadas.



En el contexto de la investigación se sabe que los sistemas de tratamiento de aguas residuales vienen en un constante proceso evolutivo permitiendo a estos posicionarse como una industria clave en la sociedad actual ya que con el desarrollo de nuevas metodologías y equipos en la rama de los bioprocesos es posible integrar diferentes tipos de residuos para generar co y sub productos de amplio valor agregado, como bioetanol, biogás, biohythane, H_2 , ácidos grasos, sustancias de elevado valor proteico, compuestos remediadores de suelos entre muchas otras biomoléculas (Kilts et al., 2009; Passos et al., 2018; Pavlostathis & Giraldo-Gomez, 1991). El concepto de biorrefinerías ha evolucionado entorno de estos modelos de generación de bioproductos en un sistema cerrado donde todo es aprovechado y los residuos son vistos como materia prima en el cual la digestión anaerobia (DA) genera múltiples beneficios en términos de degradación de residuos, cierre de ciclos de vida, recuperación energética y generación de valor agregado como una solución a la necesidad de implementar nuevos sistemas de tratamiento de residuos, uno de los requerimientos clave para seguir desarrollando está tecnología es la optimización en tiempos experimentales y la modelación en software de CFD plantea reducir estos tiempos experimentales al estudiar las relaciones hidrodinámicas para lograr un óptimo acoplamiento de los procesos (Xiang et al., 2020).

La DA es un proceso fundamental donde se realizan pasos de suma importancia para el resto de operaciones como la producción de ácidos grasos volátiles o la obtención primaria de biogás, por tanto, se necesitan biorreactores y condiciones operacionales de alta

eficiencia que incrementen en su máximo posible los rendimientos, dentro de los procesos operacionales. Para optimizar la DA se utiliza la separación de las fases acidogénica y metanogénica con el fin de brindar condiciones específicas a cada grupo de microorganismos, existen diferentes trabajos experimentales que respaldan estos sistemas de tratamiento, Khabibullin et al., 2019 estudió el comportamiento de biorreactores tipo ASTBR con volúmenes a escala laboratorio de 2.64L para la acidogénesis y 2L, 1.52L y 0.74L para metanogénesis logrando resultados destacables en la producción de biogás con rendimientos de 0.34 L·g/DQO removida y demostró que la mayor remoción en términos de DQO se logró gracias al diseño del biorreactor basado en la tasa orgánica para lo cual se debe fijar en primer lugar la tasa de flujo específica de agua tratada, la cual se halla con la relación entre el caudal de entrada y el volumen del reactor también se comprueba que el rendimiento específico de biogás producido a partir de la materia orgánica consumida es independiente de la tasa de carga orgánica aplicada, ya que éste varía ligeramente con el cambio de la tasa orgánica específica lo que brinda una guía metodológica en el proceso de diseño de biorreactores con alta eficiencia en remoción de materia orgánica.

Los ácidos grasos volátiles (AGV) son materia prima para los procesos metabólicos de los microorganismos en la cadena de digestión anaerobia y se debe controlar su producción operacionalmente, Fuess et al., 2017 enfatiza en la gestión de los tiempos de retención demostrando que biorreactores de configuración ASTBR y UASB soportan el aumento de la carga orgánica de entre 15 y 30 kg DQOm³d⁻¹ siempre que se mantengan periodos de retención menores a 24h para evitar saturación e inhibición y garantizar remociones cercanas al 80%. La experimentación con modificaciones físicas en los biorreactores ha llevado a grandes avances en términos de rendimiento y optimización, un importante desarrollo han sido los lechos inmovilizadores o matrices de soporte ubicados al interior de los biorreactores donde se fijan grupos de microorganismos lo que garantiza mayor contacto del fluido y mejores relaciones entre las especies microbianas. Las variaciones en la configuración pueden incrementar significativamente el potencial energético hasta en un 150% ya que los gases energéticos producto de la digestión anaerobia (H₂ y CH₄) pueden concentrarse en mayor proporción con estas variaciones hidrodinámicas para el tratamiento de residuos de la industria azucarera como la vinaza y el bagazo de caña de azúcar (Fuess et al., 2017; Solé-Bundó et al., 2019).

Las variables hidrodinámicas fomentan los fenómenos de transporte de masa y calor al interior del biorreactor así como determinan la distribución de nutrientes y materia orgánica durante la DA (Gelves et al., 2013), como se comentó en la sección anterior, de manera experimental se pueden obtener datos como el TRH que permiten modelar el comportamiento hidrodinámico al interior de los biorreactores mediante ensayos de trazadores (Levenspiel, 2012), el cálculo de TRH teórico puede ser validado para determinar fallas u obstrucciones en el flujo, por ejemplo Otálora-Tapiero, 2018 encontró que en un reactor tipo ASTBR diseñado para un TRH teórico de 8h hubo un desfase del 9% en este tiempo durante su operación experimental, esta fluctuación es normal en la operación de

sistemas continuos y está dada por la variabilidad en los métodos operacionales, principalmente la velocidad de la bomba de alimentación, formación de biomasa en los canales de conducción y sedimentación de biomasa en la parte baja del biorreactor.

El comportamiento multifásico está determinado por la interacción entre las fases líquida y gaseosa al interior del biorreactor y este puede ser clasificado como mezcla completa o flujo pistón donde se debe determinar el número de reactores en serie que lo componen, así como la existencia de cortos circuitos, caminos preferenciales o volúmenes muertos (Carvalho et al., 2008; Krychowska et al., 2020; Levenspiel, 2012; Paladino & Neviani, 2020; Pavlostathis & Giraldo-Gomez, 1991). Las adecuaciones físicas también han permitido desarrollar reactores novedosos enfocados por ejemplo en la remoción de nutrientes donde los requerimientos de los microorganismos son aún más específicos que en la remoción de materia orgánica por ello un biorreactor multi-ambiente (AnoxAn) desarrollado por Blanco-Aguilera, Lara, Barajas, Tejero, & Diez-Montero, 2020 cuenta con un perfil vertical interior donde hay una diferencia considerable de concentración de oxígeno creando microambientes que separan los microorganismos diferenciando sus requerimientos de O_2 para alcanzar mejores tasas de remoción de nutrientes específicamente.

La modelación en CFD permite disminuir tiempos experimentales y evidenciar oportunidades de mejora, para ellos en las últimas décadas se han ofertado múltiples softwares, algunos tienen aplicaciones específicas como NUMECA (Pouffary et al., 2008) que se enfoca en modelaciones aeroespaciales, ANSYS que tiene una aplicación de amplio rango y es licenciado, por otro lado, OpenFOAM es un software de código abierto que cuenta con múltiples solvers capaces de resolver problemas integrando fluidos líquidos y gaseosos con elevada confiabilidad, dada la complejidad y variabilidad del comportamiento de los biorreactores. Desde los años 80 se ha trabajado en la búsqueda de soluciones analíticas a estos sistemas con muy pocos casos de éxito, es por esto que el análisis numérico en CFD permite determinar soluciones aproximadas con alto grado de precisión de las ecuaciones en derivadas parciales no lineales que describen el comportamiento de los reactores multifásicos (A. Bakker & A. Van den Akker, 1994; Galerkin et al., 2017; Rivas et al., 2020b).

Los principales métodos de solución de modelos CFD son de diferencia finita, volumen y elementos que se utilizan para aproximar la solución utilizando métodos numéricos. Ninguno de estos proporciona soluciones precisas. La diferencia entre los tres es el modelo matemático y ecuaciones de gobierno que se utilizan para el cálculo (Peeters, 2016). Dependiendo de la precisión de la solución que desee, puede utilizar cualquiera de las tres. FDM proporciona la aproximación menos aceptable para un costo de cálculo mínimo, mientras que FVM proporciona la mejor aproximación con un alto costo computacional. La mayoría de los simuladores utilizan FEM ya que proporciona una aproximación aceptable con cálculos más bajos que el volumen finito.

Uno de estos métodos es el FVM (método de volumen finito), utilizado en la discretización de las ecuaciones diferenciales de Navier-Stokes, que gobiernan el comportamiento de los fluidos en la naturaleza. Además, este método proporciona la base para el análisis CFD (Dinámica de fluidos computacional), que es una rama de la mecánica de fluidos que utiliza el análisis numérico para resolver problemas que involucran flujos de fluidos.

El modelo de turbulencia es característico de los fluidos y están divididos en los siguientes en orden de complejidad creciente:

- Modelos algebraicos (ecuación cero): longitud de mezcla (modelo de primer orden).
- Modelos de una ecuación: k-modelo, μ t-modelo (modelo de primer orden).
- Modelos de estrés algebraico: ASM (modelo de segundo orden).
- Modelos de tensión de Reynolds: RSM (modelo de segundo orden).
- Modelos de ecuación cero

OpenFOAM proporciona tipos de simulación para flujo de turbulencia, RAS y LES. Como sugiere el nombre, la simulación RAS se basa en los modelos basados en RANS.

La construcción del modelo tiene etapas importantes que deben acoplarse correctamente para lograr modelos robustos y simulaciones confiables, la configuración del modelo numérico requiere las ecuaciones de gobernanza que describen matemáticamente el comportamiento de los fluidos en las diferentes zonas al interior del reactor, comúnmente se utiliza análisis de flujo transiente solucionando las ecuaciones de Navier Stokes en promedio con Reynolds (RANS) para flujos turbulentos e incompresibles dada su relativa simplicidad y bajo costo computacional, utilizan factores de continuidad, conservación de momento integrando presión y velocidad de fluidos (H K Versteeg. W Malalasekera, 2016). Las geometrías se realizan en software de mallado y se requiere lograr una buena resolución y baja saturación computacional por lo que es común mezclar celdas hexaédricas y poliédricas y aquí es donde se reafirma la importancia de trabajar con software Open Source de elevada capacidad como Blender. Por último, la validación puede realizarse con experimentos de distribución de tiempo de residencia (RTD) que también indican si se encuentran cortos circuitos en el reactor, caminos preferenciales y/o volúmenes muertos (Blanco-Aguilera, Lara, Barajas, Tejero, & Diez-Montero, 2020; Ranganathan & Sivaraman, 2011).

La posibilidad de integrar múltiples herramientas informáticas para potenciar la simulación en CFD permite aprovechar las bondades de cada software, por ejemplo Krychowska et al., 2020, utilizó datos obtenidos experimentalmente de RTD para validar una simulación desarrollada con CFD donde se buscó generar un modelo compartimental (CM) compuesto por una parte estructural que desarrolla la arquitectura matemática y una parte paramétrica que contiene los parámetros extrínsecos (variables de respuesta) del modelo. Para este estudio se utilizó el reactor prefabricado BioFlo 415, en el desarrollo de CFD el software ANSYS CFX fue el utilizado, durante el desarrollo de los modelos matemáticos se

empleó MATLAB y SIMULINK y la geometría fue creada en AutoCAD, lo cual respalda la viabilidad de acoplar múltiples herramientas para estos estudios. Con los resultados de perfiles de velocidad y patrones de flujo se facilita el desarrollo de modelos cinéticos o metabólicos para determinar reacciones y reducir tiempos y costos de los bioprocesos orientado hacia una aplicación industrial (Boccardo et al., 2015).

Con base en la argumentación bibliográfica y el sustento teórico se plantea la siguiente pregunta de investigación que con este trabajo se pretendió responder **¿Es posible integrar múltiples herramientas para modelar y simular el comportamiento hidrodinámico de biorreactores de manera eficiente y garantizando su replicabilidad?**

3. OBJETIVOS

3.1. Objetivo general

Establecer un flujo de trabajo de código abierto con enfoque en la evaluación del comportamiento hidrodinámico de biorreactores aplicando dinámica computacional de fluidos (CFD).

3.2. Objetivos específicos

- Establecer las herramientas Open Source de mallado para la aplicación en el área de CFD con enfoque en biorreactores.
- Complementar el pre-procesamiento, los requerimientos para modelación y pos procesamiento.
- Crear el flujo de trabajo que integre las herramientas viabilizando la aplicación de esta tecnología a biorreactores.

4. METODOLOGÍA

Este capítulo describe el proceso que se siguió para desarrollar el flujo de trabajo orientado a la evaluación hidrodinámica mediante la integración de herramientas Open Source, para ello se estableció que la metodología de CFD requiere 3 etapas esenciales dentro de los procesos de simulación donde en cada una de ellas se utilizan herramientas de gran efectividad como Blender, Open Foam y Para View.

4.1. Mallado

Como fue descrito en las secciones anteriores el proceso de mallado es el primer paso dentro de la metodología de modelación y cada aspecto debe ser cuidadosamente seleccionado ya que aquí se sientan las bases del todo el desarrollo del modelo y posterior simulación.

Este es el primer paso del pre-procesamiento, el cual consiste en la definición de un modelo representativo de un problema de flujo para posteriormente exportar a un programa CFD por medio de una interfaz amigable para el operador y la posterior transformación de esta

entrada en una forma adecuada para su uso por parte del solucionador. Las actividades del usuario en la etapa de pre-procesamiento implican:

1. Definir la geometría de la región de interés.
2. Definir las condiciones de modelación de variables como masa, longitud, tiempo, temperatura, concentración, corriente e intensidad lumínica, dependiendo del tipo de modelación se aplican unas u otras.
3. Definir las ecuaciones que gobiernan el modelo y en base a esto se selecciona el solver que se requiere para solucionar la modelación.

Considerando la complejidad e importancia del proceso de mallado en este trabajo fue necesario recurrir al proceso de ensayo/error para la construcción de la malla, el proceso de generación de la malla es simplificado ya que el dominio es subdividido en bloques más sencillo, en este caso la determinación topológica permitió una malla hexaédrica.

4.1.1. Validación de software para mallado

Dentro de la industria de geometrización digital y mallado existen múltiples herramientas de diseño para el proceso de mallado como lo son:

- [AutoCAD](#);
- [SolidWorks](#);
- [SketchUp](#);
- [Salome Meca](#);
- [Blender](#).

Así como se mostró en las secciones anteriores dos de los principales software usados son Salome Meca que por defecto viene ligado a OpenFOAM ya que es una herramienta simple con capacidades que suplen algunos requerimientos dentro del proceso como se muestra en la

El grupo de trabajo ya tiene experiencia en herramientas de diseño, especialmente en Blender dadas las ventajas mencionadas anteriormente, hemos prototipado, renderizado y simulado geometrías de biorreactores, partes específicas de bombas peristálticas y sensores de medición y automatización.



Dado que blender emite la modelación de una manera visual e interactiva en tiempo real, facilitó el armado de la estructura del biorreactor, adicionalmente durante el proceso de establecimiento de metodología se encontró SwiftBlock como herramienta complementaria que simplifica la exportación.

Tabla 1 y Blender, cuyo uso es novedoso y pocos estudios publicados lo integran

El grupo de trabajo ya tiene experiencia en herramientas de diseño, especialmente en Blender dadas las ventajas mencionadas anteriormente, hemos prototipado, renderizado y simulado geometrías de biorreactores, partes específicas de bombas peristálticas y sensores de medición y automatización.

Dado que blender emite la modelación de una manera visual e interactiva en tiempo real, facilitó el armado de la estructura del biorreactor, adicionalmente durante el proceso de establecimiento de metodología se encontró [SwiftBlock](#) como herramienta complementaria que simplifica la exportación.

Tabla 1. Propiedades de software de diseño de geometría.

	
Modelación gráfica y visualización 3D.	Es un software para simulación numérica con capacidad de pre y pos procesamiento.
Open Source.	Se basa en una arquitectura abierta y flexible.
Especializado en animación.	Tiene bajos niveles de optimización de rendimiento.
Generar imágenes fotorrealistas (Renderización)	
Texturización.	
Capacidad de integración con herramientas desarrolladas a la medida del problema (SwiftBlock).	

4.1.2. Blender

Este software especializado de la industria del diseño gráfico es usado para discretizar la geometría en elementos hexaédricos, el reactor fue diseñado a escala real con las medidas tomadas del construido en el laboratorio, para la elaboración de la geometría se utilizó el Software libre de código abierto [Blender](#) en su versión 2.90.0, aprovechando sus bondades en el diseño gráfico 3D. Es compatible con la totalidad de la canalización 3D: modelado, montaje, animación, simulación, renderizado, composición y seguimiento de movimiento, edición de vídeo y animación 2D.

La base de la malla fue creada a partir de la adición de un bloque y el escalado del mismo con la medida del tanque #3 de un volumen útil de 2653 ml como se muestra en la Ilustración 1, posteriormente se utiliza el modo edición para extruir los otros dos tanques, la entrada y salida. Para realizar la transición entre tanques la función de escalar en planos

seleccionados que se activa con la tecla S, por defecto se debe dejar una mínima inclinación en las transiciones que para este caso fueron de 5mm ya que SwiftBlock debe reconocer este desnivel para permitir el paso del fluido entre tanques al momento de correr la modelación.

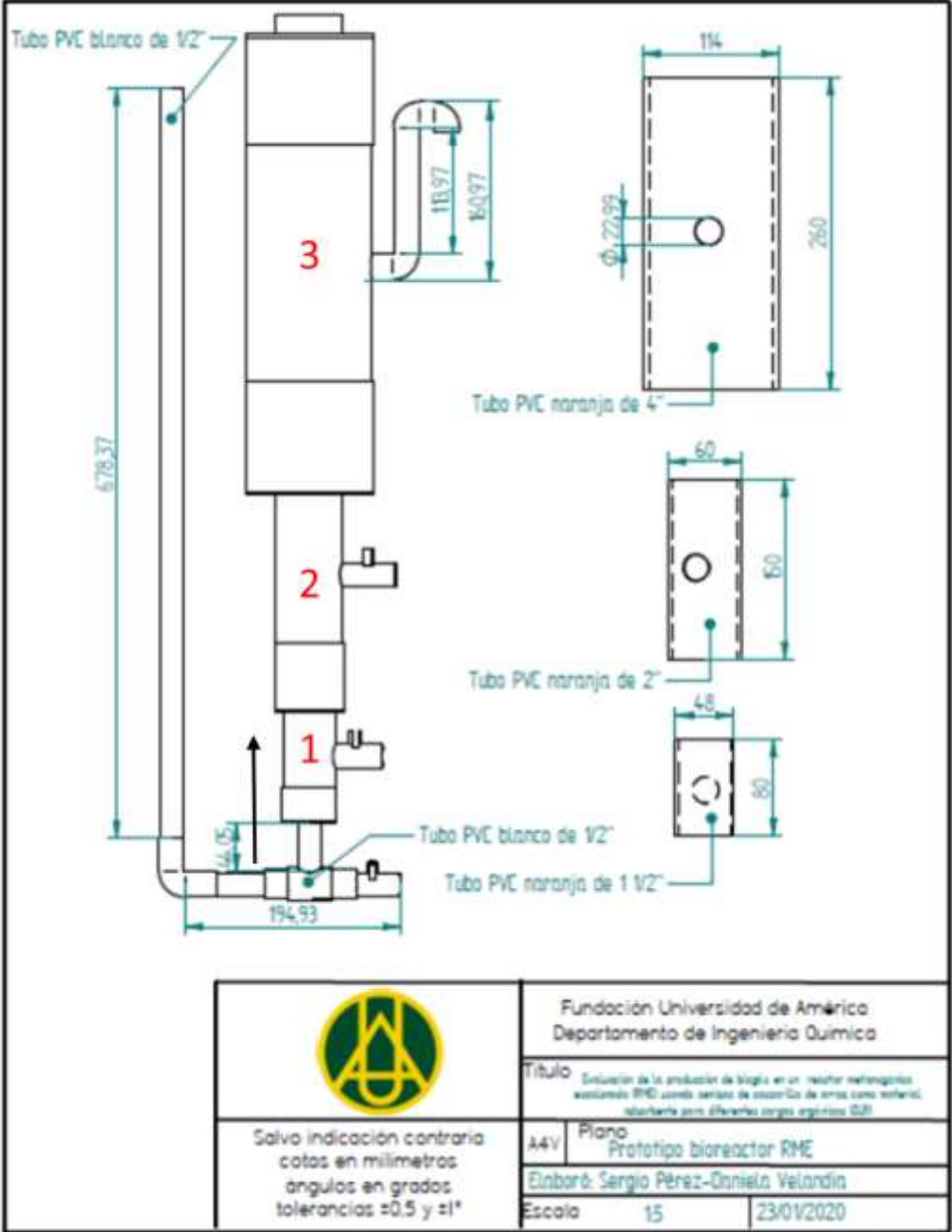


Ilustración 1. Geometría del biorreactor en estudio.

SwiftBlock

Para aprovechar las capacidades de Blender se necesitó desarrollar una mayor comprensión del proceso de la construcción de las geometrías, durante la investigación de herramientas

se encontró que Nogenmyr, K.J., Folkersma, 2020 desarrollaron un complemento llamado [SwiftBlock](#) que se añade a Blender y permite exportar las geometrías de manera práctica para obtener la plantilla necesaria en la modelación. Mediante la transcripción de Vertices, Bloques y Parches que se guarda en la carpeta system y posteriormente de genera el Internal Mesh en OpenFOAM que permite modelar en las celdas internas de la malla previamente diseñada, la ventaja de esta herramienta sobre otras como SnappyHexMesh es que garantiza la creación de mallas estructuradas con control de capas límite, permite la creación de mallas con base ortogonal con células alargadas o estiradas para la modelación, aquí se pueden clasificar de manera controlada las mallas hexaédricas en el interior o el exterior de la geometría y permite integración con formas rectangulares, cilíndricas o

esféricas (Ilustración 2).

Esta herramienta cuenta con los siguientes módulos:

Block Method Settings:

Este panel contiene configuraciones generales y botones de herramientas.

Edge Settings:

Este panel se utiliza para establecer valores de parámetros y aplicarlos en los bordes seleccionados.

Edge Goups:

Este panel permite la agrupación de bordes e integración de la geometría.

Projections:

Este panel contiene configuraciones para proyectar bordes a superficies en otros objetos de malla.

Boundary Patches:

El panel Parches de límite se usa para especificar caras de límite y sus tipos utilizando el sistema de

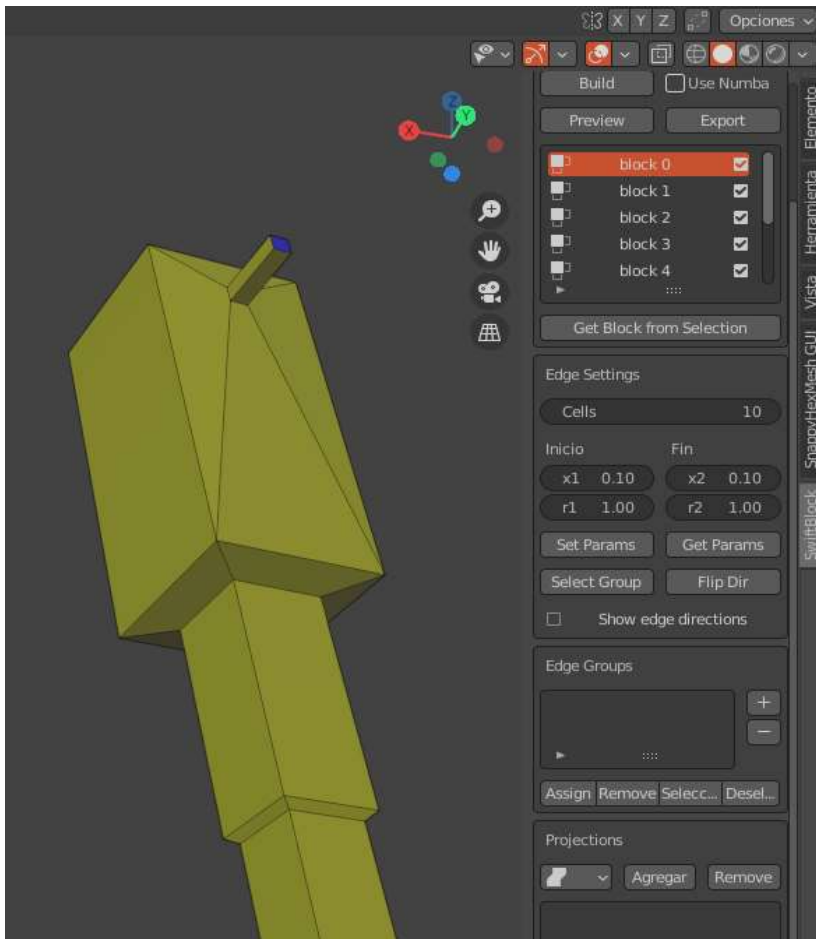
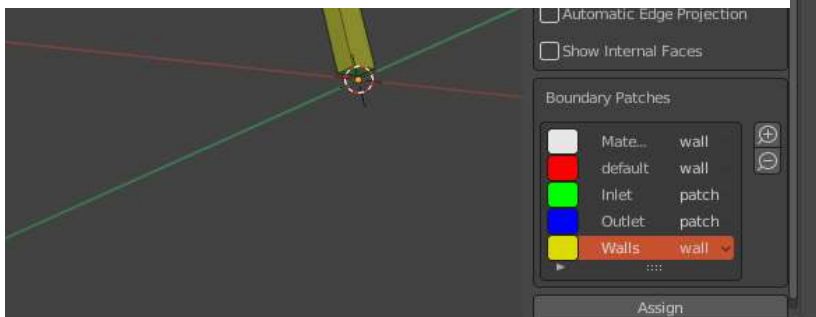


Ilustración 2. Interfaz gráfica de usuario de SwiftBlock



material Blender. Los parches se muestran como una lista. Inicialmente, todas las caras se agregan al parche de límite predeterminado.

A continuación se detalla el proceso de mallado en un paso a paso:

Paso 1: Reconocimiento previo.

Para comenzar se recomienda hacer tutoriales breves para ganar familiarización con el software, una vez instalado se abre su interfaz de acuerdo a la Imagen 4

Se sugieren los siguientes videos:

- [Introducción al modelado.](#)
- [Extrude Blender 2.80 Fundamentals.](#)
- Blender Beginner Tutorial [Part 1](#), [Part 2](#), [Part 3](#), [Part 4](#), [Part 5](#), [Part 6](#), [Part 7](#).

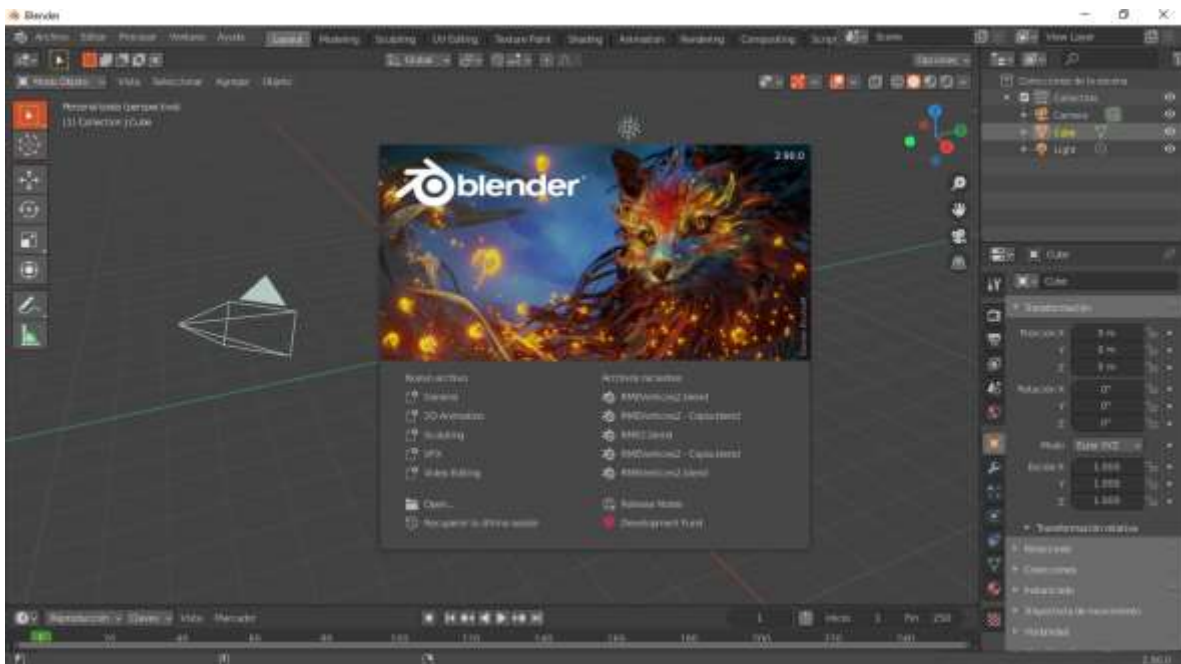


Imagen 4. Página de inicio de Blender (GUI).

Paso 2: Configuraciones base.

En el panel de “Colecciones de la escena” del lado derecho se deben eliminar las entidades creadas por defecto de “Camera” y “Light”.

Posteriormente en el mismo panel se seleccionó la configuración de propiedades (A) como se muestra en la Imagen 5 y por facilidad en el proceso de escalado a medida se asignaron las unidades de centímetros (B) en las propiedades de longitud.

Es importante destacar que durante el proceso se manejaron las vistas de sombreado y rayos x que se encuentran como se muestra en la imagen (c) las cuales permiten observar

las paredes y caras de la geometría durante la construcción. Finalmente en este paso es importante manejar los modos edición y objeto que permiten la activación y desactivación de propiedades.

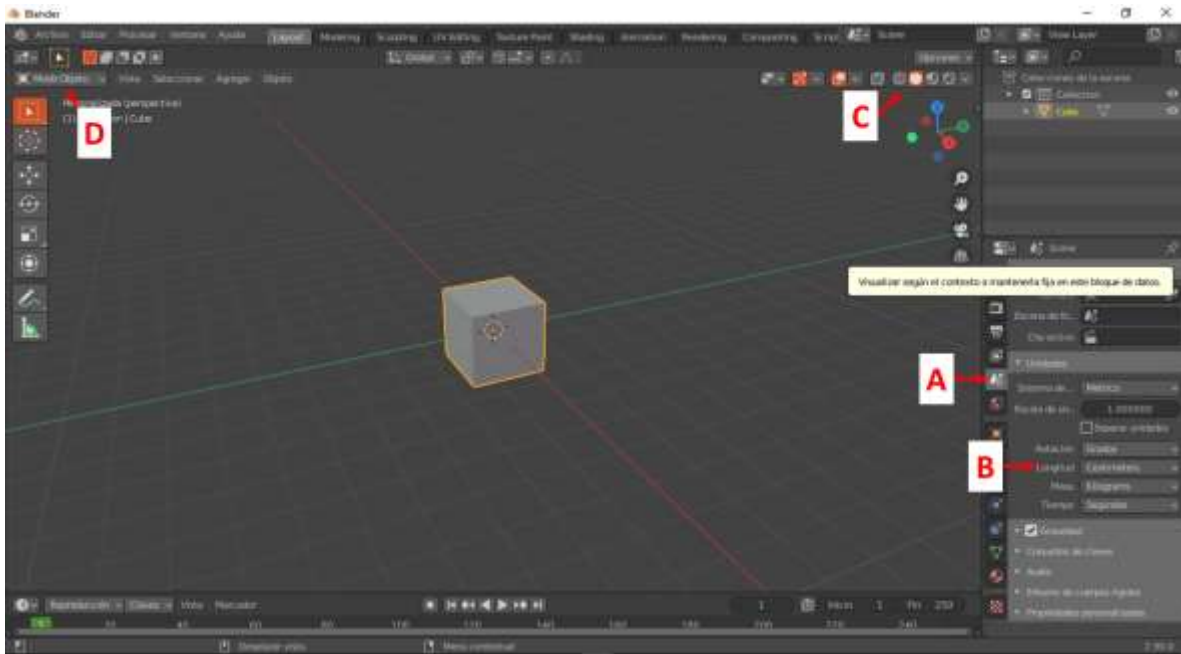


Imagen 5. Familiarización con Blender.

Paso 3: Construcción de tanques.

En este paso se realiza el proceso de extrusión y reajuste de medidas.

- I. En el modo edición se activa la selección por caras, y se selecciona la cara inferior del tanque base.
- II. Posteriormente con ayuda del teclado numérico se elige la vista en 2D con el tanque sobre el plano z y se oprime la letra “E” que activa el proceso de extrusión, con el mouse se hala hacia abajo en un primera extrusión de 1cm, como se muestra en la Imagen 6 (A).
- III. Posteriormente con la misma cara seleccionada (Imagen 6 B) se procede a oprimir la tecla “S” que activa la función de escalado y con el clic izquierdo del mouse oprimido se ajusta el nuevo grosor del tanque inferior.

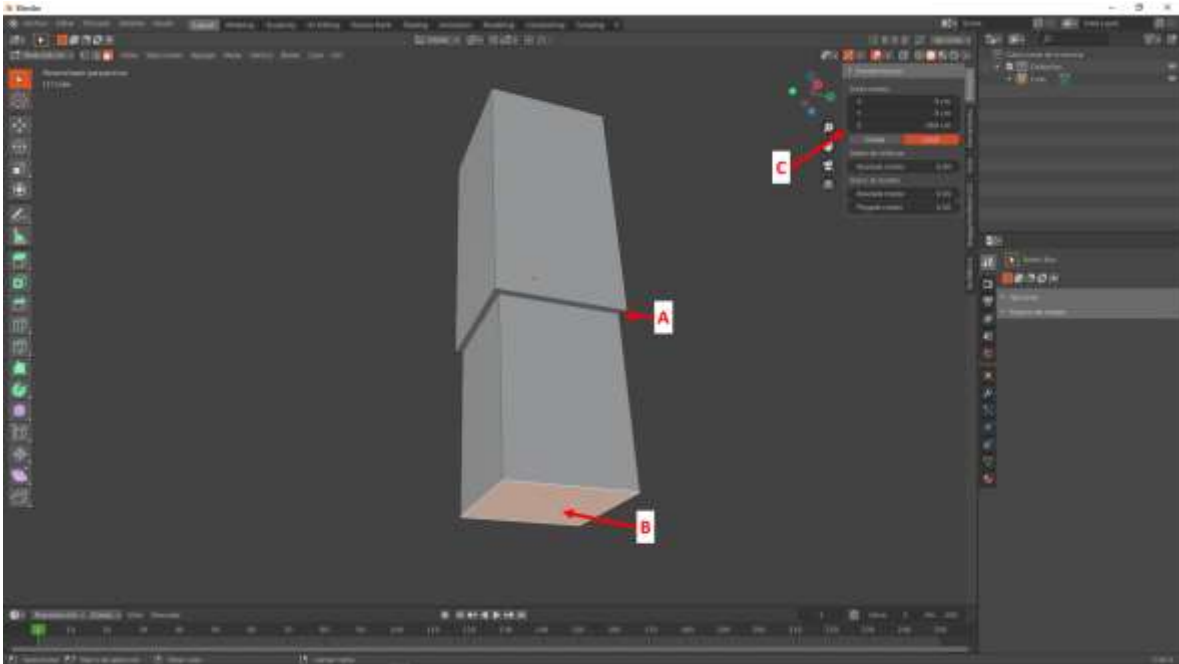


Imagen 6. Proceso de extrusión y escalado para la construcción de la geometría.

- IV. A continuación se selecciona la cara con el nuevo grosor y se repite el paso II sucesivamente hasta lograr los contenedores necesarios, en este caso 3 tanques y el tubo de entrada y salida del biorreactor.
- V. A partir del tanque superior que debe ser ajustado en medidas se realiza una leve extrusión de 1cm hacia la parte inferior oprimiendo la tecla
- VI. Como las medidas son ajustadas con el mouse estas carecen de precisión, para solucionar esto y ajustar a la escala exacta se utiliza la selección por vértices y las medidas de ajustan en el panel de transformación como se muestra en la imagen XX (C).

De esta manera se logra que el reactor cumpla con las medidas establecidas a escala real, dado las características inherentes de creación de la malla hexaédrica en este proyecto se asumió una geometría de tanques rectangulares con el fin de lograr el acople entre los nodos al momento de la subdivisión de celdas. En la Imagen 7 se muestra cómo queda la geometría del reactor después de seguir los pasos indicados anteriormente, desde el interior donde se garantiza el acople entre tanques y desde el exterior.

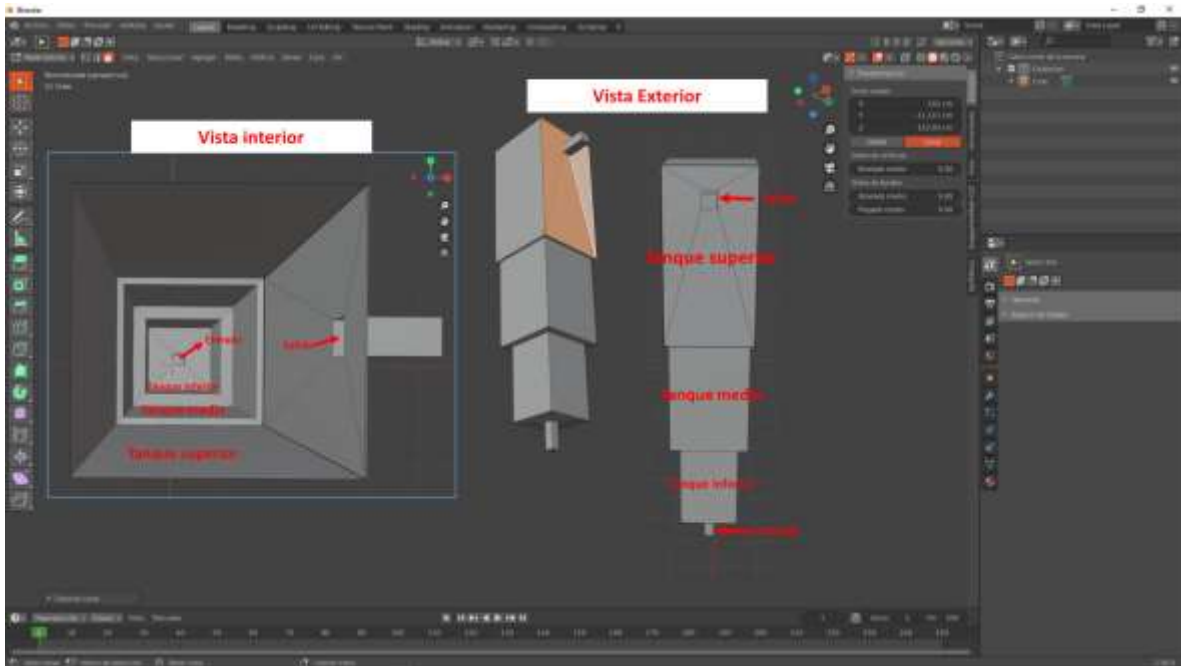


Imagen 7. Vistas del reactor construido.

Paso 4: SwiftBlock

Como se especificó anteriormente la herramienta SwiftBlock permite la exportación de la geometría creando los archivos requeridos para la posterior simulación, a continuación se describe el proceso utilizado con esta herramienta en un paso a paso

Paso 1: Asignación de propiedades

La herramienta debe ser instalada como un complemento (Add on) en blender, posteriormente se hace su activación y se abre como lo muestra la Imagen 8 (A).

- I. En modo edición se seleccionan las caras para darles propiedades en la pestaña de "Boundary Patches" (B), en el ejemplo se muestra como Inlet de color verde se le da la propiedad de patch con el símbolo de + que aparece en la Imagen 8 (C) ya que por allí ingresa el fluido y deberá tener condiciones iniciales.
- II. Posteriormente en color lima se le asigna la propiedad de "Walls" (D) a

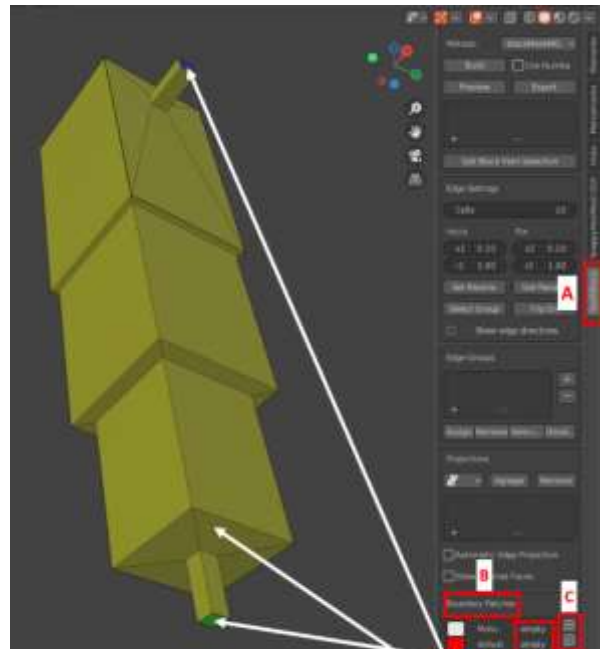


Imagen 8. Asignación de características a las caras del biorreactor.

las paredes de los tanques que contienen el fluido.

- III. Y por último en el paso de asignación la pared de salida de color azul, se configura como patch ya que por allí el fluido sale y se miden las velocidades finales.

Paso 5: Exportación.

Este proceso se hace también dentro del tablero de SwiftBlock, como se indica a continuación; en la Imagen 9 se tienen los 3 paneles principales para el proceso de exportación:

- I. En el panel de “Edge Settings” (A) se configura la resolución de celdas que se desean por cara, en este caso son 10, es de aclarar que si se desea más resolución se deben aumentar las celdas y esto con lleva a un mayor costo computacional ya que las subdivisiones donde se debe iterar el modelo matemático aumentan.
- II. Una vez se indicaron las propiedades y se configuraron las características de la geometría con los pasos anteriores se accede al panel “Block Methods Settings” (B), allí en modo objeto se selecciona todo el reactor.
- III. Después de la selección se le da clic en “Build” lo cual construye los bloques de la geometría que contienen, las caras, los vértices y los nodos que unen toda la estructura, posteriormente se validan y se da clic en “export” guardando el archivo en una carpeta conocida.

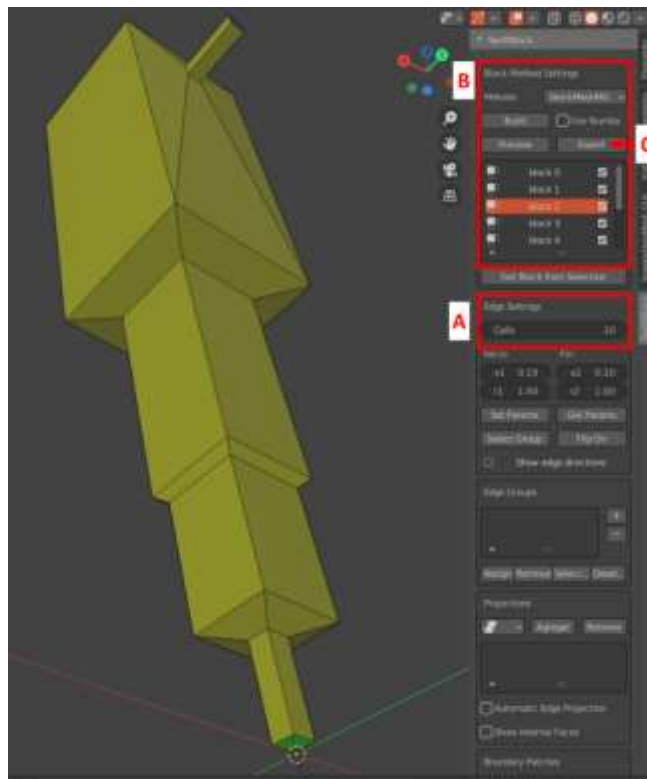


Imagen 9. Número de celdas y exportación de la geometría.

- IV. El resultado de los pasos anteriores será un archivo .txt (Formato Texto) que se recomienda abrir en [NotePad ++](#) dadas sus capacidades de edición de texto, el archivo con el nombre de **BlockMeshDict** contiene la transformación en coordenadas de la geometría creada, como se puede ver en la Imagen 10 para el ejemplo anteriormente descrito del biorreactor RME.

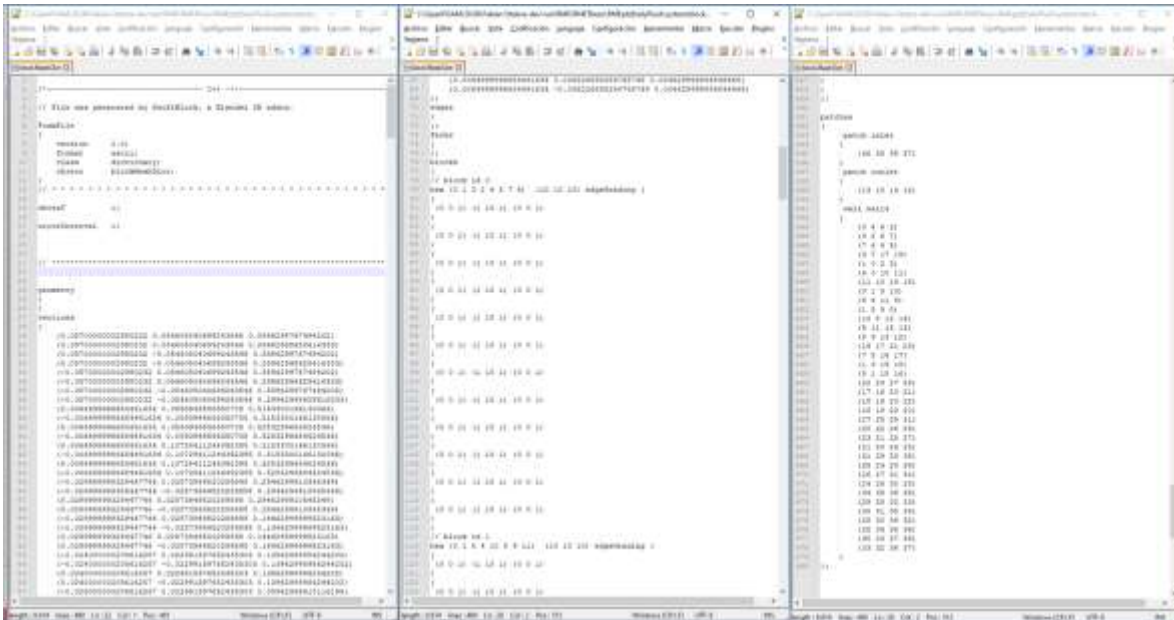


Imagen 10. Script de características de la malla.

Como se puede observar en la Imagen 10 la interfaz de escritura de la geometría es simple en cuanto a interactividad y compleja en cuando a ubicación gráfica de los puntos es ahí donde esta herramienta facilita la exportación y escritura, una vez esto listo se realiza la revisión de la malla escrita con respecto a blender.

Paso 6: Validación.

El proceso de validación se hace mediante la apertura del archivo **BlockMeshDict** en el pos-procesador como se muestra a continuación en la Imagen 11:

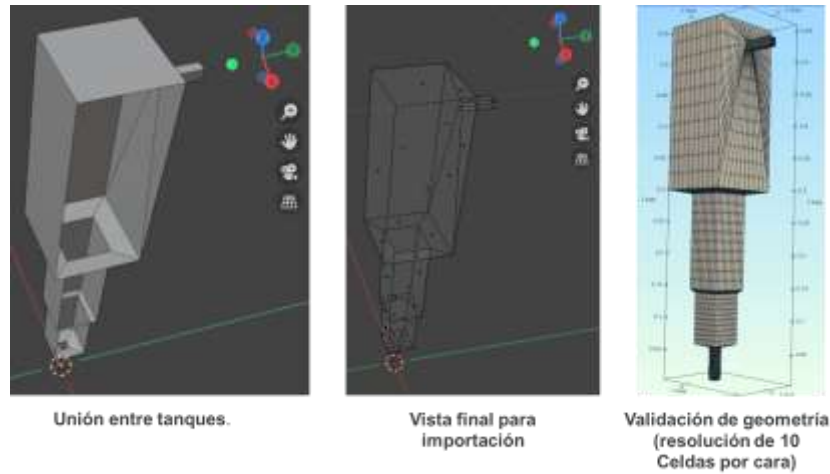


Imagen 11. Validación del proceso de mallado.

Paso 7: Parámetros del modelo

En este paso se deben establecer los parámetros de la modelación, de acuerdo con lo expuesto por diferentes autores en capítulos anteriores es necesario que establecer las variables, el modelo matemático y en base a esto elegir el o los solucionadores que cuentan con la configuración necesaria para lograr la convergencia del modelo.

A partir de la experiencia lograda en anteriores etapas experimentales y trabajos del grupo de investigación como el de Pérez Peña & Velandia Blanco, 2020 quienes estudiaron la hidrodinámica real de reactor aquí analizado se obtuvieron valores de parámetros referenciales que en combinación con parámetros ya probados en la literatura (Jordan & Harasek, 2019) para el solucionador seleccionado se logró una buena convergencia del modelo.

A continuación en la Tabla 2 se muestran las variables utilizadas, su significado dentro de las ecuaciones y los respectivos valores:

Tabla 2. Parámetros del modelo

Variable	Valor
P (Representa la densidad del fluido)	1000 kg·m ⁻³
U (Vector de velocidad)	3 m·s
nut (Viscosidad turbulenta)	0.2 kg·m·s
nuTilda (Representa la viscosidad cinemática)	0.2 m ² ·s ⁻¹
k (Energía cinética turbulenta)	0.2 m ² ·s ⁻³

f (Factor de relajación elíptica)	(s ⁻¹) Para este caso se mantiene en zeroGradient.
Omega (Disipación de turbulencia específica)	Para este caso se da como un valor fijo definido como (440.15 s ⁻¹).
v2 (Velocidad potencial)	Velocidad potencial en (0.25 m ² ·s).
Epsilon (Tasa de disipación de energía cinética turbulenta)	(0.2 m ² ·s ⁻³), que junto con la nuTilda conforman el modelo de turbulencia (k-ε) que es el modelo más común en la dinámica de fluidos computacional (CFD) para simular las características de flujo medio para un flujo turbulento.

Paso 8: Definir los límites y suposiciones del modelo

Otro aspecto importante dentro del proceso de definición es establecer los límites y las suposiciones que se abarracaran en el modelo o la simulación, a continuación se detallan:

- El objetivo de esta metodología es crear un flujo de trabajo mediante la integración de herramientas que pueda ser replicado y adaptado según las necesidades de cada caso y lo que se requiera estudiar en cada reactor, la validación de este estudio fue realizada con un biorreactor RME y dada la complejidad de las interacciones entre fases (líquido – sólido – gas) este trabajo fue enfocado en comprobar el proceso de llenado del biorreactor únicamente para la fase líquida.
- Como se explicó anteriormente en los requerimientos del mallado es necesario garantizar una conexión entre nodos lo cual exigió que en este primer trabajo la geometría se asumiera rectangular y no cilíndrica como el biorreactor RME real.

4.2. Corriendo la modelación

Una vez establecidos los parámetros del modelo, se procede con la corrida, a continuación se explica el proceso de elección del software y como se configuró el proceso de modelación:

4.2.1. Elección software de modelación

De manera paralela al proceso de mallado se realizó la investigación sobre los software disponibles para el proceso de modelación, en la actualidad existen múltiples software y han sido desarrollados con fines de modelación específicos. Los tres principales son:

- COMSOL
- Ansys (Fluente y CFX).
- OpenFOAM

El estudio de la dinámica computacional de fluidos es complejo ya que se deben tener en cuenta múltiples factores, en el proceso de elección del software y el método de solución la comprensión detallada del problema es fundamental, por tanto la definición de las limitaciones y lo que se irá a asumir dentro del modelo permiten lograr con mayor precisión un reflejo cercano del comportamiento real de la unidad modelada.

Como se describió anteriormente las soluciones en CFD están basadas en iteraciones hasta la convergencia, en la parte matemática los métodos de solución más comunes son:

- Método de volúmenes finitos (FVM).
- Método de diferencias finitas (FDM).
- Método de elementos finitos (FEM).

Y a partir de la investigación conceptual de este trabajo se infiere que los métodos FVM y FDM brindan soluciones discretas, mientras que FEM brinda una solución continua (hasta cierto punto), la mayoría de investigadores opta por FVM y FDM ya que tiene menos complejidad al momento de programar que FEM y de manera específica se considera que FVM proporciona mejores propiedades de conservación, dicho esto se hizo un análisis de los software siempre teniendo en cuenta la ventaja que tiene OpenFOAM sobre ANSYS y COMSOL de ser una herramienta de código abierto y fuente libre.

En primer lugar la herramienta COMSOL, que en un principio fue concebida como un software de análisis estructural y sus solucionadores fueron programados con un algoritmo basado en FEM lo cual garantiza aproximaciones aceptables de solución pero en cálculos relativamente simples ya que después de cierto límite iteraciones la solución deja de ser continua con este método.

ANSYS y OpenFOAM utilizan FVM, dada su capacidad de solucionar flujos con características complejas y de amplio requerimiento de iteraciones pese a que este método acarrea un mayor gasto computacional, OpenFOAM utiliza este método para resolver sistemas de ecuaciones diferenciales parciales ligadas a cualquier malla 3. Sus solucionadores de flujo de fluidos se desarrollan dentro de un marco de solución iterativo, robusto, implícito, de presión-velocidad, aunque se aplican técnicas alternativas a otros solucionadores de mecánica continua (FDM), de manera paralela los modelos matemáticos intrínsecos de OpenFOAM integran en bajo nivel los procesos de descomposición de dominios garantizando que los solucionadores se puedan correr en general sin necesidad de codificación específica.

Expuesto lo anterior se decidió integrar al flujo de trabajo OpenFOAM, esta herramienta cuenta con interfaz gráfica de usuario simple, dada su sencillez puede tornarse compleja de entender en los primeros acercamientos por lo cual es recomendable asistir una los siguientes videos que desde la voz de la experiencia pueden facilitar en gran medida el trabajo con OpenFOAM.

- [How to install OpenFOAM and run a simulation in Windows 10 in 2020 – tutorial.](#)
- [How to run your first simulation in OpenFOAM®.](#)
- [OpenFOAM for Beginners win10 4 - Running Tutorial Files with Allrun](#)

El software fue descargado en la [página oficial](#) en su versión v-2016 20.09, se optó por correrlo directamente en windows 10 aunque lo ideal es instalar el complemento que crea una máquina virtual de Linux en Windows ya que el programa es original de este sistema operativo y algunas funciones o comandos operativos pueden variar de un sistema operativo a otro.

4.2.2. Condiciones del solucionador

Con base en las condiciones iniciales del modelo planteado en el pre-procesamiento mostradas en la Tabla 2 se desarrolló la corrida de la modelación.

Para la solución de la modelación se eligen los solucionadores de SimpleFoam que resuelve las ecuaciones de estado estacionario para flujos turbulentos incompresibles, este se utiliza esencialmente en la modelación del llenado del reactor.

La Tabla 4 muestra la comparación de diferentes estudios que han aplicado modelación de fluidos en biorreactores, donde se plantean condiciones muy diversas en los parámetros que definen los solucionadores elegidos.

Se utilizó el solver simpleFOAM con una malla hexaédrica para calcular para el comportamiento del flujo en forma de campo de velocidad y presión y así visualizar la distribución del fluido a lo largo de la geometría

Para simpleFOAM:

1. El solver calcula los campos de presión y velocidad en el tiempo estipulado inicial para la entrada del fluido.
2. Se calculan los campos de presión y velocidad en el tiempo final que fue estipulado en el directorio “controlDict”.

SimpleFOAM

SimpleFoam es un solucionador de estado estable para fluidos con comportamiento de flujo turbulento incompresible, que utiliza el algoritmo SIMPLE (método semi-implícito para ecuaciones vinculadas a presión).

Este solver está basado en un proceso iterativo de solución separada donde cada ecuación que caracteriza el sistema como las ecuaciones de velocidad, presión y las variables de turbulencia son resueltas secuencialmente y la solución de las ecuaciones anteriores se inserta en la ecuación siguiente.

En la ecuación de momento aparece la linealidad de ρ y esta se resuelve calculando los valores de velocidad y presión de la iteración anterior, el solver integra la dependencia de la presión

para evitar un desacoplamiento entre las ecuaciones de momento y presión y, por tanto, la aparición de oscilación de alta frecuencia en la solución.

En primera medida el solver resuelve la ecuación de cantidad de movimiento que como resultados indica un campo de velocidad pero esta no satisface la ecuación de continuidad (presenta divergencias), allí se integran al modelo las las ecuaciones de impulso y continuidad que se utilizan para construir una ecuación para la presión donde el objetivo es obtener un campo de presión que sirve para insertar en la ecuación de momento y genera el campo de velocidad libre de divergencias. Por último, después de corregir el campo de velocidades, se resuelven las ecuaciones de turbulencia y el procedimiento de solución iterativa anterior se repite hasta la convergencia.

Matemáticamente las ecuaciones se representan como se indica en la Tabla 3 y para comprender como se resuelven las ecuaciones de turbulencia en el proceso del solver se recomienda leer los procesos de [Simulación de Promedio de Reynolds \(RAS\)](#).

Tabla 3. Principales ecuaciones de la solución del solver.

<p>Ecuación de cantidad de movimiento (Momentum):</p>	$\frac{\partial (u_{rj} u_i)}{\partial x_j} + \epsilon_{ijk} \omega_i u_j = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \frac{1}{\rho} \frac{\partial}{\partial x_j} (\tau_{ij} + \tau_{t_{ij}})$	<p>\mathbf{u} Representa la velocidad absoluta. \mathbf{u}_r La velocidad relativa $\boldsymbol{\omega}$ La velocidad angular del marco giratorio de referencia τ_{ij} y $\tau_{t_{ij}}$ Son las tensiones viscosa y turbulenta. Dada que la velocidad relativa \mathbf{u}_r aparece en el término de divergencia, el flujo de cara que ϕ aparece en la discretización de volumen finito de la ecuación de cantidad de movimiento debe calcularse con la velocidad relativa.</p>
<p>Ecuación de presión:</p>	$a_P^* u_P^* = -\sum_N a_N u_N^* + b_P + \frac{1-\alpha}{\alpha} a_P u_P^{n-1} - \nabla p_P^{n-1} = \mathbf{H}[\mathbf{u}^*] - \nabla p_P^{n-1}$	<p>Esta surge a partir de la ecuación de momentum en su forma semidiscreta. La ecuación de continuidad es la misma para la velocidad relativa descrita en un marco de referencia sujeto a un movimiento de cuerpo rígido y la velocidad descrita en un marco de referencia inercial. Por esa razón, la siguiente ecuación es válida para la velocidad relativa $\mathbf{u}_{Pr}^* = \mathbf{u}_P^* - \boldsymbol{\Omega} \times \mathbf{r}$ y también para la velocidad absoluta \mathbf{u}_P^*. \mathbf{r} Es el vector desde el centro de rotación hasta la posición de interés. Por esta razón se obtiene que las ecuaciones se resuelvan en el caso de un sistema de referencia en movimiento simplemente agregando el</p>

		término $-\boldsymbol{\Omega} \times \mathbf{r}$ en ambos lados de la ecuación y se continúa con la solución de la ecuación.
Ajuste de las oscilaciones de la presión	$\frac{\partial(\rho u u)}{\partial x} = -\frac{\partial p}{\partial x}$	La forma habitual de evitar el llamado efecto "tablero de ajedrez" es interpolar las velocidades en las caras mediante la interpolación de Rhie y Chow. El efecto tablero de ajedrez tiene su origen en la ausencia de términos de presión vecinos en la ecuación de momento debido a la discretización del término del gradiente de presión en el marco del método de volumen finito usando una disposición de variables colocadas. La misma ausencia de términos de presión vecinos se obtiene cuando el término laplaciano se discretiza por interpolación lineal del gradiente de presión en los centros de las caras. En la implementación del solucionador simpleFoam, la interpolación de Rhie y Chow no se realiza de forma explícita.
El solucionador emplea el algoritmo SIMPLE para resolver la ecuación de continuidad:	$\nabla \cdot \mathbf{u} = 0$ $\nabla \cdot (\mathbf{u} \otimes \mathbf{u}) - \nabla \cdot \mathbf{R} = -\nabla p + \mathbf{S}_u$	<ul style="list-style-type: none"> u = Velocidad p = Presión Cinemática R = Tensor S = Fuente de momentum u <p>Este algoritmo requiere la entrada de Presión Cinemática (m²/s²) y velocidad (m/s).</p>

		Los modelos físicos que se implementan son el de turbulencia y de volúmenes finitos.
--	--	--

Tabla 4. Elección del Solver.

Modelo	Parámetro				
	1	2	3	4	5
	Velocidad de flujo regida por el # de Reynolds	Variación de la Viscosidad y esfuerzo cortante	Cambios de densidad con respecto al tiempo	Unidireccional o multi	Solvers
Background	Laminar	Newtoniano	Incompresible	Unidireccional	icoFoam : solver Transciente para flujo laminar incompresible de fluidos newtonianos. pisoFoam : solver transciente para flujo turbulento incompresible, utilizando el algoritmo PISO.
Blanco-Aguilera, R., Lara, J. L., Barajas, G., Tejero, I., & Diez-Montero, R. (2020).	Turbulento	Newtoniano	Incompresible	Multidireccional	simpleFoam : solucionador de estado estable para flujos turbulentos incompresibles.
Najarnezhadmashhadi, A., Wärnå, J., Eränen, K., Trajano, H. L., Murzin, D., & Salmi, T. (2021)	Turbulento	No newtoniano	Compresible	Multidireccional	Solucionadores de ecuaciones diferenciales ordinarias (ODE). multifaseEulerFoam : solver para un sistema de múltiples fases de fluido compresible, incluida la transferencia de
Koerich D.M., Lopes G.C., Rosa L.M. (2020)	Turbulento	Newtoniano	Incompresible	Unidireccional	Based on a 3D, transient, Eulerian–Eulerian two-phase approach coupled with a turbulence model. In the two-phase Eulerian solver , the pressure–velocity coupling was computed by using the PIMPLE algorithm.
Severin T.S., Brück T., Weuster-Botz D. (2019)	Laminar	Newtoniano	Incompresible	Multidireccional	Eulerian solver interFoam for a two phase system with the fluids water and air at 20°C. For validation simulations with passive species transport, adaptations were made to the solver according
Modelo utilizado	Turbulento	Newtoniano	Incompresible	Unidireccional	Solver simpleFoam de estado estacionario para flujos turbulentos incompresibles scalarTransportFoam Solucionador de ecuaciones de transporte escalar pasivo

4.2.3. Corrida

Después de haber configurado y los datos y la información, en las carpetas directorio “0”, “constant”, y “system” el siguiente paso es correr el modelo en el software. Dada la poca interactividad de la interfaz de OpenFOAM la forma más sencilla de ejecutar la modelación es correr el caso en primer plano, de esta manera primero se ingresa al directorio del caso y se escribe simpleFOAM en la terminal del sistema.

Para esta tesis se utilizó un computador ASUS modelo X555Gde procesador de 64 bits AMD A10-8700P y tarjeta de video Radeon R6 con 10 núcleos, 4 de CPU y 6GPU corriendo a una velocidad de 1.80 GHz, memoria RAM de 8 GB.

En la Imagen 12 se presentan las carpetas de las que consta el modelo completo con el solucionador simpleFOAM, en la carpeta ‘0’ están las variables iniciales que reflejan el comportamiento experimental del fluido en el biorreactor, en la carpeta ‘constant’ se encuentran las propiedades de transporte y turbulencia que conservamos iguales a lo estipulado por el solver y en ‘Polymesh’ está la malla, finalmente en la carpeta ‘System’ están las condiciones de la corrida del modelo (**controlDict** Imagen 13) y el archivo blockMeshDict se crea una vez construida la malla en el GUI de openFOAM.

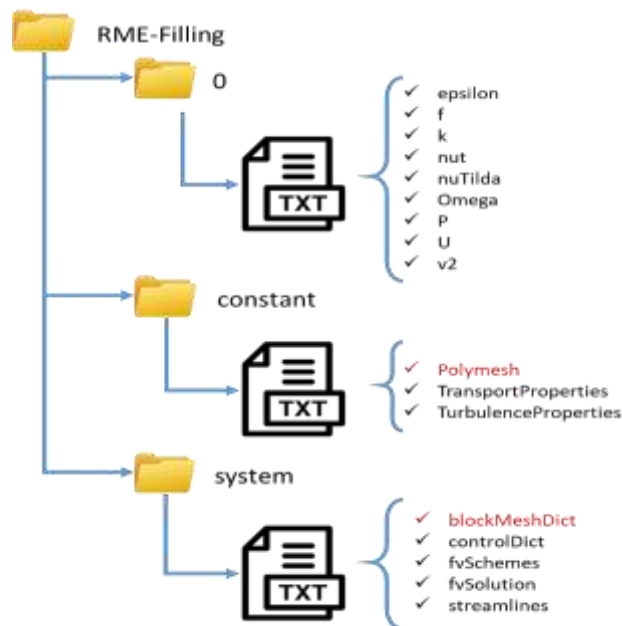


Imagen 12 Flujo de carpetas de configuración de la modelación.

Como se mencionó anteriormente la interfaz de usuario es básica y la configuración de la modelación debe ser hecha modificando archivos .txt, pero para manipular el programa y navegar en la terminal son necesarios una serie de comandos descritos en la tabla Tabla 5, con ellos se ingresa y se gestiona el movimiento entre carpetas para buscar la carpeta que contiene el modelo y allí ejecutar el nombre del solver para resolver el problema de fluido.

4.2.4. Pos procesamiento

Para este paso se eligió paraView ya que es el software que más se usa en en la aplicación de openFOAM, este tiene ventajas ya que ha sido construido pensando en brindar una adecuada visulización del fluido y la geometría también permite examinar y analizar los resultados de simulaciones en paralelo por tanto puede optimizar el rendimiento de las CPUs del equipo y está orientado a una visualización rápida de archivos con una gran cantidad de datos.

Para iniciar el posprocesamiento de paraFoam, se escribe “paraFoam” en la ventana de terminal desde dentro del directorio de casos esto abre la ventana de paraView que muestra la solución del caso dado gráficamente, hay un panel en el lado izquierdo de la ventana paraview que se usa para controlar el caso dado, este panel esta copuesto por:

- Navegador de canalización, que enumera todos los módulos abiertos en paraView;
- Panel de propiedades, que contiene las selecciones de entrada, como tiempos, regiones y campos para el caso dado;
- Panel de visualización, que se utiliza para controlar la representación visual del módulo seleccionado;
- Panel de información, que proporciona información sobre la geometría y el tamaño de la malla.

En esta tesis, el resultado final de los campos de presión y velocidad se muestra gráficamente en toda la geometría para la simulación con el solver simpleFoam, en la Imagen 14 se puede evidenciar como el solver muestra la estructura interna de la geometría y el comportamiento de llenado.

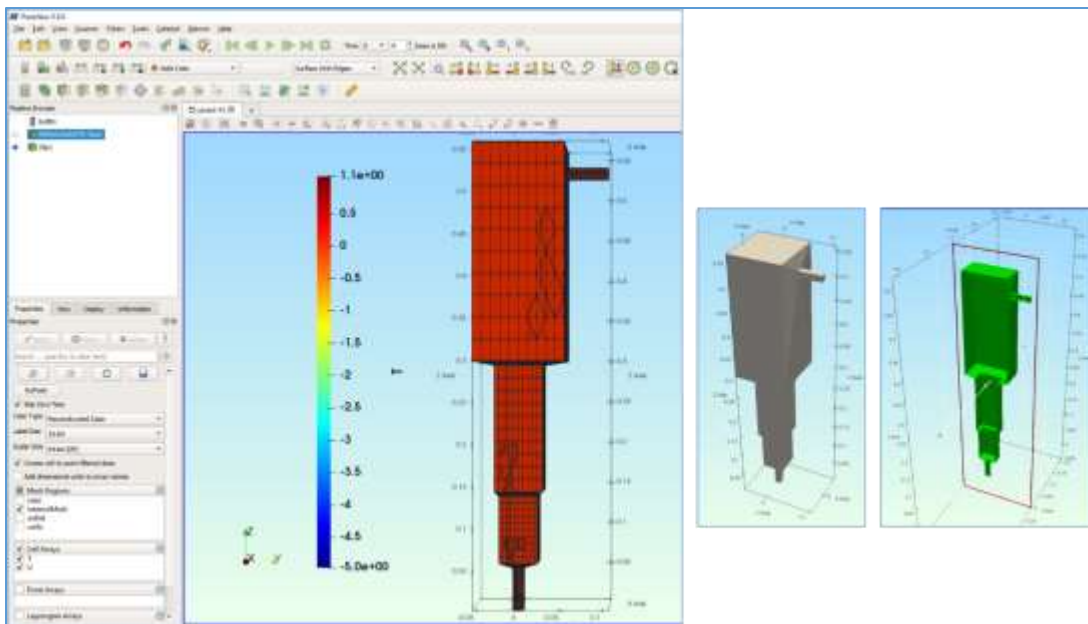


Imagen 14. Validación de la geometría en pos-procesamiento.

4.3. Flujo de trabajo

Finalmente una vez validados los 3 pasos y software implicados en el desarrollo de la modelación se obtuvo el flujo de trabajo deseado (Imagen 15) en Open Source, este apartado recopila brevemente las herramientas y su aplicación al caso de estudio.

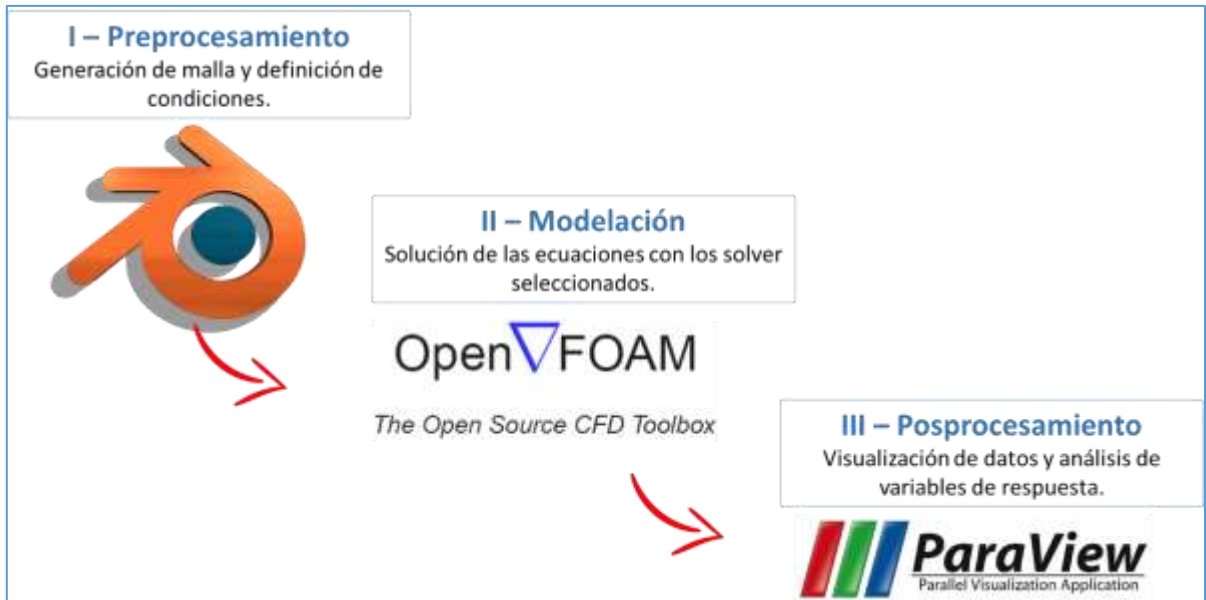


Imagen 15. Esquema de flujo de trabajo desarrollado.

4.3.1. Biorreactor RME

La configuración del biorreactor modelado Imagen 16 Imagen 16 consiste en tres tanques de diámetro variable, como lo detalla (Pérez Peña & Velandia Blanco, 2020), por lo que denomina reactor metanogénico escalonado, su flujo es ascendente y una de los principales postulados de esta configuración es incrementar la mezcla al interior ya que en la unión entre tanques se pretenden crear zonas de agitación que a su vez disminuyen la velocidad del líquido. El biorreactor pretende almacenar en el primer tanque la mayor cantidad de material biológico primario, es decir el inóculo y a raíz de lo mencionado anteriormente también se pretende disminuir el lavado y conservar una mejor calidad de comunidades microbiana al interior.

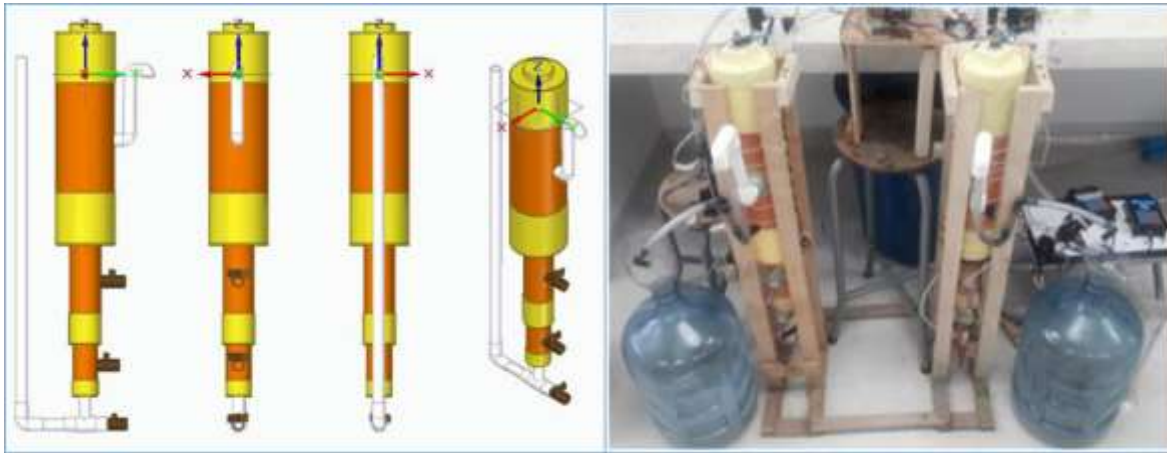


Imagen 16. Biorreactor en estudio, ilustrado y real.

5. RESULTADOS Y DISCUSIÓN

En este capítulo se muestran los resultados obtenidos desde el proceso de selección de herramientas hasta el análisis final de integración con la aplicación al caso de estudio del proceso de llenado del biorreactor RME, como se ha descrito anteriormente el objetivo de este trabajo fue establecer un flujo de trabajo mediante la integración de herramientas de simulación de código abierto que permiten la comprensión de un sistema de bioprocesos dinámico para la modelación y simulación de biorreactores en un entorno virtual. Por esto se evaluaron las características de las herramientas, sus aplicaciones junto con las condiciones del biorreactor RME en estudio.

5.1. Mallado y pre-procesamiento

La configuración del procedimiento de mallado se explica en la sección 4.1 de la sección de metodología, a raíz de esta metodología se logró la exploración de software, la implementación de estrategias de mallado e integración de herramientas y complementos.

Dentro del proceso de mallado existen múltiples software y para que el modelo reconozca la geometría se tiene que generar un archivo .CAD, que contenga las representaciones físicas ajustadas a la geometría deseada en un entorno virtual o asistido por computadora. Las principales tecnologías que se utilizan en el diseño asistido por computadora (CAD) para representar exactamente geometrías complejas son las B-splines y las Non-Uniform Rational B-Spline (NURBS), que fueron desarrolladas para mejorar el proceso de diseño, y allí radica uno de los principales retos intrínsecos del proceso de mallado.

Uno de los principales software como salome utilizado por (Mekonnen, 2016) la modelación de un sensor manifold con una malla tetraédrica, esta herramienta viene ligada a procesos de modelación CFD en software como openFOAM dada su relativa facilidad computacional, es de fácil instalación, su consumo computacional es mínimo ya que fue diseñado bajo

algoritmos simples de modelación, y su interfaz gráfica es mixta ya que permite la elección de geometrías como cilindros o rectángulos pero su adecuación a los requerimientos de usuario debe ser realizada con programación y configuración de puntos y coordenadas manuales lo cual torna complejo el proceso de construcción específicamente en geometrías 3D con alto detalle en sus componentes, como nuestro caso ya que el biorreactor está compuesto por 5 compartimentos de diferentes tamaños y volúmenes donde cada uno cumple una función específica de mezcla e influye en el comportamiento del fluido dentro del reactor.

La aplicación de una malla estructurada en cualquier aspecto de la generación de geometrías tiene ciertas ventajas y desventajas, en este caso se optó por una malla hexaédrica a diferencia de autores como Krychowska et al., 2020 que utilizó una geometría tetraédrica de cerca de 1.9 millones de celdas construidas en el software AutoCAD para aplicar en ANSYS, nosotros encontramos mayor confiabilidad en este tipo de geometrías ya que nuestro modelo es en tres dimensiones y bajo una construcción hexaédrica se permite que las celdas se interconecten en seis puntos o nodos con sus celdas vecinas y así generar la malla. Fundamentalmente en las mallas de dos dimensiones la celda central está conectada por cuatro celdas vecinas mientras que en tres dimensiones, la celda central está conectada por seis celdas vecinas.

Tu et al., 2013 plantea la facilidad de utilizar una malla estructurada como la aquí propuesta ya que esta permite una fácil gestión de la información y la conectividad entre la estructura es regular y consistente, pero cuando las geometrías se tornan complejas y se requiere una amplia refinación los prototipos hexaédricos pueden causar inconvenientes de no ortogonalidad o asimetría en la malla que afectan directamente a las ecuaciones gobernantes y causando iteraciones irreales o variaciones no físicas en la solución del modelo.

Al igual que lo planteado por Blazek, 2015 durante la construcción de nuestra geometría tuvimos retos estructurales donde nuestras coordenadas se veían afectadas por el requerimiento de conexión en cuatro puntos de las celdas, en software como Salome este tipo de ajustes requieren términos adicionales que acarrearán mayor gasto computacional y dificultades de programación por lo que a malla puede resultar afectando en términos de precisión y eficiencia al algoritmo numérico que se aplica.

Dentro del proceso de exploración de herramientas para mallado se contempló el uso de Blender (Gumster, 2020) ya que es una herramienta conocida por nuestro grupo de investigación y como se mencionó anteriormente es Open Source de código libre lo cual da accesibilidad a la réplica de este flujo de trabajo y finalmente el motivo de mayor impacto en la inclusión de este software en este trabajo es su elevada especificidad en diseño gráfico al ser una herramienta creada para animación y renderización que permite modelar geometrías de manera visual en tiempo real en 3D con facilidad y elevada interacción de su interfaz gráfica de usuario.

Es importante señalar que dentro de nuestro proceso de investigación bibliográfica encontramos únicamente 7 artículos que previamente usaron Blender en su proceso de mallado, en la Tabla 6 se detallan estos trabajos:

Tabla 6. Resultados de implementación por otros autores de Blender a CFD.

Titulo	Cita
1. Numerical study of natural convection in a horizontal cylinder filled with water-based alumina nanofluid.	(Meng & Li, 2015)
2. Computational modelling of resin infusion process applied to small vessel hulls.	(Teixeira & Souza, 2019)
3. Generation and validation of virtual nonwoven, foam and knitted filter (separator/coalescer) geometries for CFD simulations.	(Abishek et al., 2017)
4. Validation of a novel open-source work-flow for the simulation of packed-bed reactors.	(Boccardo et al., 2015)
5. Fine and ultrafine particle deposition in packed-bed catalytic reactors.	(Boccardo, Sethi, et al., 2019)
6. Computational analysis of transport in three-dimensional heterogeneous materials.	(Boccardo, Crevacore, et al., 2019)
7. Multiple Aneurysms AnaTomy CHallenge 2018 (MATCH): Phase I: Segmentation.	(Berg et al., 2018)

Como es de notar son pocos y recientes los trabajos que han visto en Blender una herramienta eficaz para los procesos de modelado lo cual respalda nuestro postulado de novedad in incursión en una rama prometedor y de futuro prominente, y afirma que esta investigación sienta las bases de futuros desarrollos por el grupo de trabajo y sus proyectos.

Encontramos que el Politécnico de Torino (Italia) es una institución que investiga ampliamente en el desarrollo de CFD aplicado a ingeniería de bioprocesos y su principal autor es [Boccardo G](#), en el artículo número cuatro plantea la importancia de integrar un flujo de trabajo que permita desarrollar estructuras de alta confiabilidad en la simulación de biorreactores de lecho empaquetado, siendo oficialmente el primer autor en acoplar Blender a OpenFOAM, posteriormente en el año 2019 publicó un estudio (Boccardo, Crevacore, et al., 2019) a cerca del estudio y escalado de las ecuaciones de transporte en medios porosos para reactores químicos y biológicos, basando el mallado en estrategias estructuradas con base cartesiana, también publicó un estudio sobre la deposición de partículas finas y ultrafinas en reactores catalíticos de lecho compacto donde realizan simulaciones CFD para obtener los perfiles de comportamiento del transporte de partículas, este artículo fue de gran utilidad en este trabajo ya que aporta en la comprensión del proceso e importancia de la especificidad del mallado, los anteriores trabajos concuerdan

con las ideas aquí planteadas a cerca de la creación integración de software open source especializados en las áreas requeridas dentro de la modelación CFD.

Durante la construcción de la geometría virtual encontramos dificultad en la generación de malla cilíndrica ya que el proceso requiere generar proyecciones desde los vértices, aristas y caras garantizando interconexión en los nodos de la superficie de una geometría rectangular y en los casos referenciales aquí estudiados plantean que este proceso debe ser manual modificando los scripts y su sintaxis, puesto que este es nuestro primer acercamiento a CFD y nuestro objetivo principal es construir el flujo de trabajo, como grupo de investigación optamos por construir la malla rectangular para aplicar el llenado del reactor. Como se menciona en la metodología, en la configuración del mallado se le dio a cada cara la característica de 10 celdas donde encontramos un balance perfecto entre rendimiento computacional y refinación del mallado, como puede verse en la Imagen 17 las celdas asignadas aumentan el detalle de una manera directamente proporcional al tamaño de la cara, es decir para caras grandes como los tanques del reactor las 10 celdas se van a distribuir cubriendo áreas de mayor tamaño pero en los compartimentos pequeños las mismas 10 celdas se distribuyen ocupando de manera proporcional áreas más pequeñas lo cual garantiza mayor resolución matemática y visual a la hora de dejar correr el fluido por la geometría, al igual que (Rivas et al., 2020b) la estrategia de mallado con respecto a una geometría estándar es unificar el número de celdas pero según plantea el mismo autor y de acuerdo a las experiencias aquí descritas si la geometría está compuesta por figuras complejas y alta variabilidad de curvas y paredes se recomienda usar un mallado híbrido con integración de componentes tetraédricos.

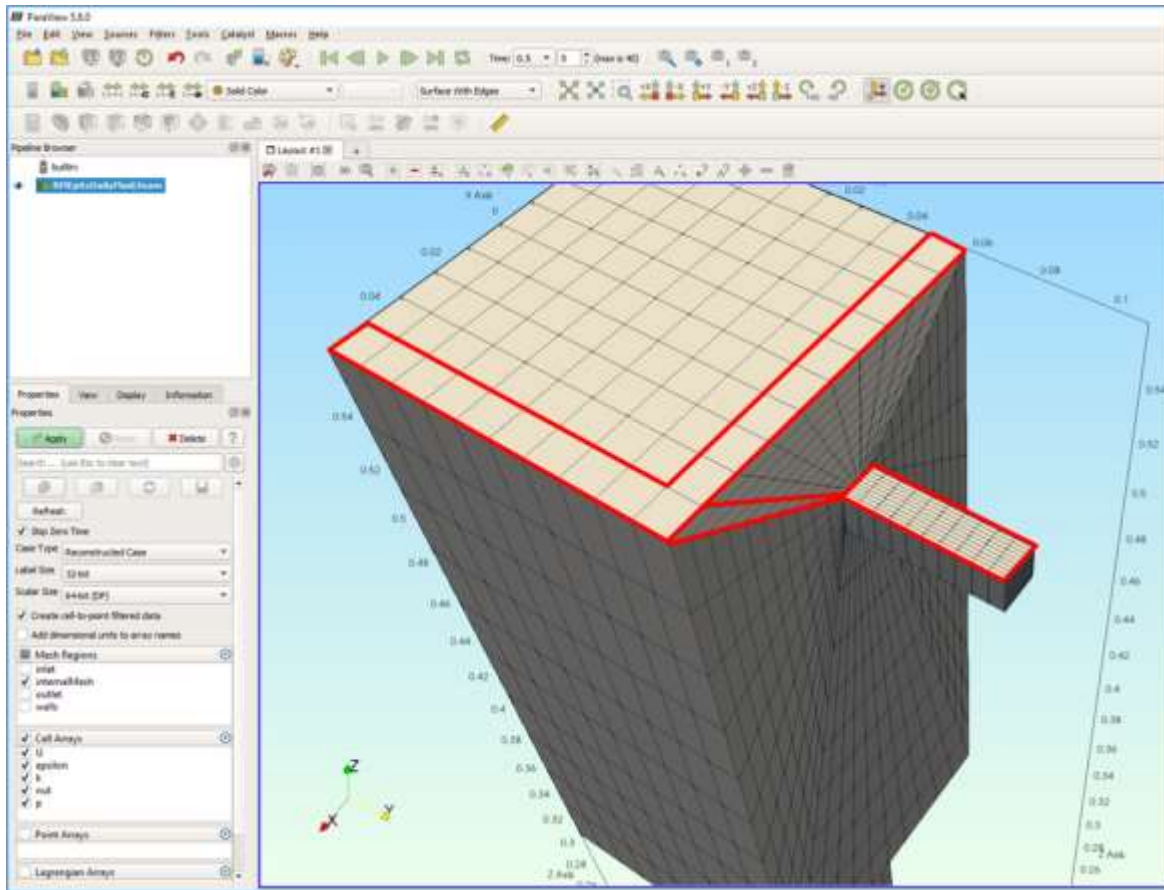


Imagen 17. Distribución de las celdas en la geometría del biorreactor.

Finalmente en el proceso de mallado se incorporó la herramienta de SwiftBlock (Nogenmyr, K.J., Folkersma, 2016) que es un complemento para blender que permite la integración de bloques de 3D con parametrización en diferentes variables de diseño entre ellas la selección de características de cada bloque como la elección de tipo de cara entre pared, vacía o parche, al igual que (Välikangas & Karvinen, 2018) encontramos rapidez y automatización en esta herramienta en cuanto a configuración de diseño en tiempo real a diferencia de Salome donde estas características deben conferirse de manera manual es los archivos de geometría.

5.2. Parámetros de la modelación y software de modelación

Dada la importancia del mallado dentro de la modelación se le dedicó un apartado especial dentro del desarrollo de este trabajo y continuando con el proceso en este apartado se discuten los resultados obtenidos de la corrida de la modelación del reactor utilizado como caso de estudio para la validación del flujo de trabajo.

Las condiciones elegidas para modelar el llenado del biorreactor y comprobar la integración de software son las expuestas en la Tabla 2 donde se tuvieron en cuenta parámetros experimentales previamente establecidos por Pérez Peña & Velandia Blanco, 2020 en conjunto con parámetros predefinidos del manual de funcionamiento del solver establecido

(simpleFOAM) como la viscosidad turbulenta y cinemática así como la tasa de disipación de energía cinética turbulenta.

Estos parámetros conservan una estrecha relación con el solucionador y el modelo de solución del problema, diferentes autores como (Boccardo, Crevacore, et al., 2019; Marcato et al., 2021; Pouffary et al., 2008) destacan la capacidad de obtener resultados precisos como una mezcla de variables de tipo experimental teórico y suposiciones del sistema, la observación y el conocimiento de lo que se quiere modelar es fundamental al plantear el problema, es por eso que los resultados obtenidos en el llenado resultan interesantes al encontrar un comportamiento de llenado en la primer parte acorde a lo esperado pero en el segunda parte, es decir en el llenado del tanque superior un proceso de llenado por distribución.

En cuanto a la elección del software de modelación actualmente existe una amplia gama de software para modelación CFD entre ellos COMSOL, ANSYS, MFX, PALABOS y OPENFOAM entre muchos otros, que son utilizados para predecir el flujo de un fluido, en ellos se integra la modelación matemática, los métodos numéricos y las herramientas de software tanto de solución como de visualización (pre y pos procesamiento). Durante el estudio de los software disponible encontramos que OpenFOAM es la herramientas más completa para modelación CFD de código abierto disponible, al igual que (Blanco-Aguilera, Lara, Barajas, Tejero, & Díez-Montero, 2020b; Boccardo et al., 2015; Mekonnen, 2016; Mohiuddin & Mathkour, 2015) quienes han trabajado con este software en modelaciones de una fase, multifase, biorreactores, reactores químicos y transporte de partículas lo cual visibiliza la interoperabilidad que se puede lograr a través de openFOAM.

Autores como Marroquín-Fandiño et al., 2020 quienes utilizaron COMSOL y obtuvieron destacables resultados en la caracterización del fluido de un mili biorreactor o Krychowska et al., 2020 quien utilizó ANSYS para modelar y simular el comportamiento de un biorreactor dividido en compartimentos tuvieron que contar con la licencia de estos software que para el caso de ANSYS está en un valor de aproximadamente \$ 30 mil USD (aproximadamente \$ 115 millones de pesos colombianos) y COMSOL cuya licencia básica oscila los \$ 4 mil USD y debe pagarse anualmente una tarifa de mantenimiento de cerca del 20% del costo de la licencia que le permite usar nuevas versiones del software adicionalmente si se requiere modelar parámetros específicos que no están en la licencia básica se deben adquirir módulos que cuestan entre \$ 600 y \$ 4000 USD para licencias que estén actualizadas siendo así que COMSOL puede llegar a costar cerca de \$ 10 mil USD (aproximadamente \$ 40 millones de pesos colombianos), estos valores anteriores son únicamente para software de modelado si se quiere implementar el mallado en software .CAD especializado la inversión puede ser de entre \$ 700 y \$ 2 mil USD al año (aproximadamente en \$2.6 y 7.6 millones de pesos colombianos).

Expuesto lo anterior encontramos que el flujo de trabajo aquí propuesto tiene un valor de licenciamiento de \$ 0, por lo que se asegura la replicabilidad y libre experimentación,

Blender es una herramienta especializada en el diseño gráfico 3D y OpenFOAM tiene una gran capacidad en cuanto solucionadores para la mayoría de fluidos y su comportamiento pero al ser contar con un código abierto cualquier persona con conocimientos puede modificarlo para acoplar la modelación a sus condiciones.

5.3. Pos procesamiento y flujo de trabajo

En este apartado se presentan los resultados obtenidos a partir de la creación de la malla en Blender, la corrida en openFOAM y el pos procesamiento en paraView junto con una síntesis del flujo de trabajo creado.

El software de pos-procesamiento elegido fue ParaView ya que es una herramienta de bajo consumo computacional que cuenta con paneles especializados en la generación de cortes sobre la geometría que permiten una mejor visualización del comportamiento desde cualquier ángulo deseado adicionalmente permite examinar y analizar los resultados de simulaciones en paralelo por tanto puede optimizar el rendimiento de las CPUs del equipo y está orientado a una visualización rápida de archivos con una gran cantidad de datos.

La herramienta de pos procesamiento permitió visualizar el desarrollo del fluido por la geometría. La Imagen 18 muestra el comportamiento en la malla interna del reactor con el modelo descrito en la metodología resuelto, Se puede observar, como era de esperar, que el fluido se desplaza inicialmente como flujo piston pero una vez en el tanque superior encontramos una tendencia a la mezcla completa ya que el volumen de control es mayor y el resalto que sufre al cambiar de compartimento influyen en la dispersión del mismo, como lo reporta Otálora-Tapiero, 2018 el biorreactor no tiene un comportamiento fijo por el contrario tiende a mezcla completa (81.74%) y flujo pistón (18.25%) según lo encontrado experimentalmente lo anterior demuestra que la precisión del modelo desarrollado y proporciona un punto de partida confiable para implementar el flujo de trabajo desarrollado.

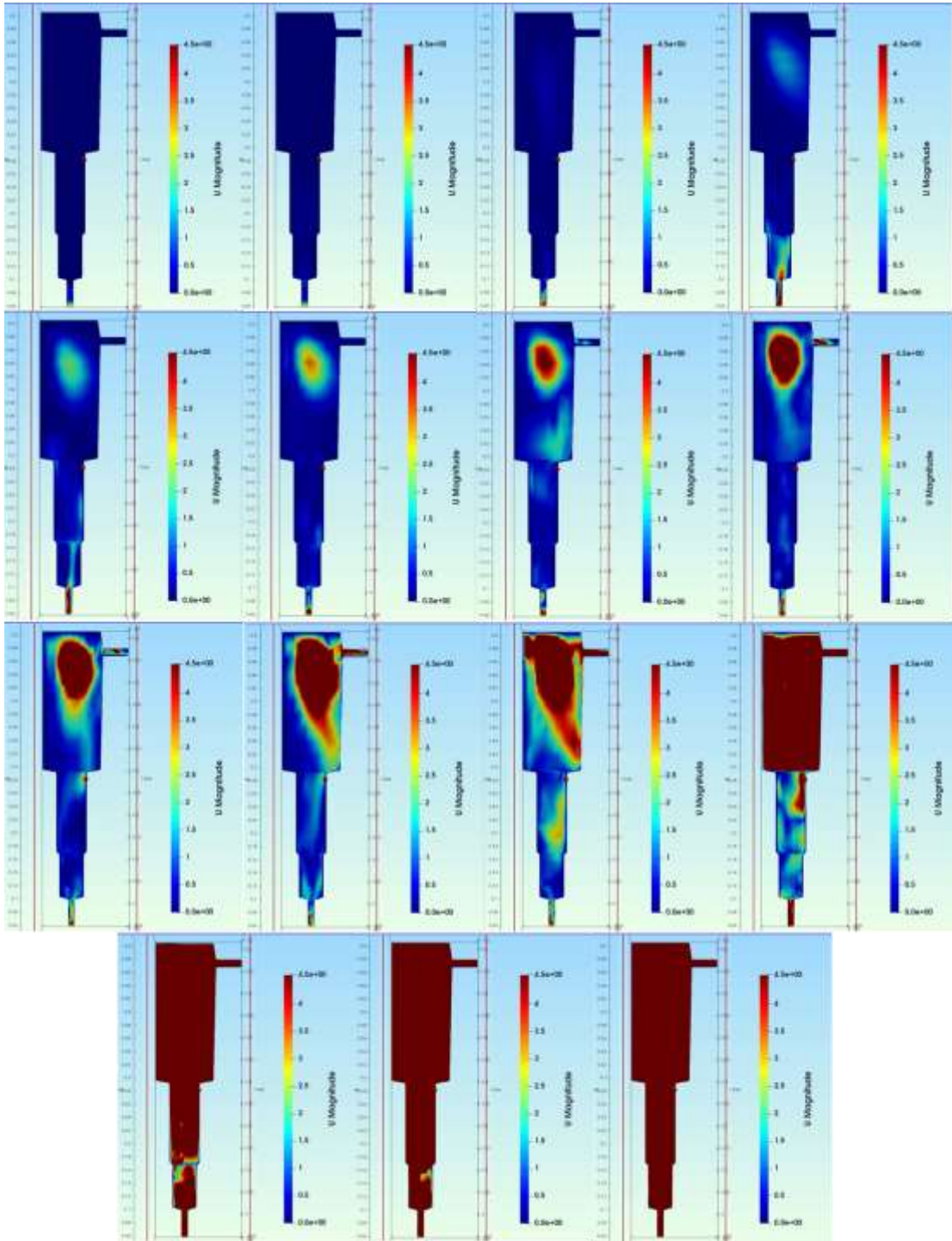


Imagen 18. Resultado de la modelación: Visualización del llenado del tanque

5.4. Aporte social

Este trabajo desea ser un aporte integral en la construcción de conocimiento orientado a los bioprocesos, las aplicaciones del flujo de trabajo aquí propuesto son esencialmente

técnicas pero las consecuencias de su aplicación tendrán repercusiones en el campo social, ambiental y económico. Estudios como el de Rahman et al., 2019 indican los diferentes beneficios que tiene la aplicación de biorreactores a escala rural específicamente para el auto abastecimiento de biogás en comunidades como por ejemplo en la india con cerca de 4 millones de biodigestores instalados, dada nuestra experiencia en esta campo podemos afirmar las siguientes ventajas de estos sistemas:

- Los biorreactores en producción continua alcanzan una independencia una vez se han optimizado los parámetros de operación que permite una generación de biogás limpio y constante.
- En un trabajo con comunidad previo a este estudio se identificó la elevada producción de hollín y cenizas que ocasiona la cocina con combustibles no convencionales como la leña por lo tanto la implementación de estos sistemas crea hábitats donde las comunidades tienen un menor riesgo de sufrir enfermedades respiratorias.
- El tiempo que invierte una familia en la recolección de madera puede ser aprovechado en otras actividades ya que los residuos que alimentan los biorreactores están disponibles con facilidad.
- A partir del efluente de estos sistemas también se obtienen productos de valor agregado como fertilizantes de gran calidad, ahorrando así el dinero de los ingresos de una familia de agricultores al reducir su dependencia de los combustibles fósiles utilizados para fabricar fertilizantes artificiales. El ahorro total de carbono fósil se estima en 4,9 toneladas al año (Rahman et al., 2019).

En el marco de lo anterior los bioprocesos optimizados mediante aplicación del flujo de trabajo aquí planteado podrían aportar a ODS específicos como: **1)** Fin de la pobreza; **3)** Salud y bienestar; **5)** Igualdad de género; **6)** Agua limpia y saneamiento; **7)** Energía sostenible y no contaminante; **9)** Industria innovación e infraestructura; **11)** Ciudades y comunidades sostenibles y **15)** Vida de ecosistemas terrestres.

6. CONCLUSIONES

Este estudio presentó la creación de un flujo de trabajo orientado al aprovechamiento de herramientas de código libre para la modelación del comportamiento hidrodinámico de biorreactores, este estudio resalta la necesidad de contar con un análisis previo detallado que permita establecer las mejores condiciones de pre-procesamiento. Las principales ventajas de acoplar Blender como generador de malla es aprovechar sus características de software de diseño gráfico 3D y el acoplamiento de la herramienta de SwiftBlock que transcribe las características de la malla como vértices, bloques y parches

Durante la simulación se procuró un ahorro en costos computacionales dando configuraciones de malla hexaédrica con celdas de (10 10 10) es decir 10 celdas por cada cara de la geometría así se logró un balance adecuado entre calidad de simulación, robustez

de modelo y velocidad de iteración, el proceso de mallado debe reproducir con gran exactitud las características físicas del reactor por lo que se debe estar atento durante la construcción a los tanques del reactor, los ajustes de las divisiones de los mismos deben quedar con una ligera inclinación (~5 mm) ya que el complemento swiftBlock detecta estos orificios como errores de construcción y por defecto los asume como una superficie sólida lo que impide el flujo del líquido al momento de la corrida.

Las modelaciones hechas en OpenFOAM 20.09 v2 muestran buen comportamiento con respecto a los test reales en el reactor RME así como las soluciones de los tutoriales que fueron realizadas para validar la herramienta, el pos-procesamiento es efectivo en ParaView aunque se debe explorar el potencial de blender en esta etapa ya que es una herramienta gráfica más potente en términos de prestaciones para la graficación continua y automatización de un fluido. A partir del breve análisis bibliométrico se puede destacar la incorporación de herramientas de programación como Python o R Studio que permitirán automatizar las simulaciones creando scripts que sinteticen los datos, ajusten de manera automática y grafiquen para generar mayor comprensión de la información.

De manera general, lo expuesto en este documento permite verificar que el flujo de trabajo integrado por las herramientas de código abierto aquí presentado es útil en la investigación y validación del comportamiento de fluidos al interior de un biorreactor metanogénico escalonado, además de exponer que esta tecnología tiene un gran potencial de aplicación dada su facilidad de alimentación con datos de tiempos de residencia, y su bajo costo computacional y económico ya que se emplea software de código libre y colaborativo.

7. SUGERENCIAS PARA FUTURAS INVESTIGACIONES

A lo largo de este trabajo se realiza un enfoque en las posibilidades actuales de la creación de un sistema integrado de herramientas *OpenSource* que permiten detallar el comportamiento de las partículas al interior de los biorreactores específicamente para este caso se seleccionó un Reactor Metanogénico Escalonado (RME), como primera recomendación en futuras investigaciones se aconseja estudiar otras configuraciones de biorreactores, adicionalmente se sugiere ampliar las simulaciones a biorreactores con afluentes reales y por último utilizar herramientas informáticas que automaticen los procesos de simulación para asegurar simulaciones con mayor eficiencia en términos de tiempo y de costo computacional así como diversificación del flujo de trabajo con software libre.

8. BIBLIOGRAFIA

- A. Bakker, & A. Van den Akker. (1994). A computational model for the gas-liquid flow in stirred reactors. *Chemical Engineering Research and Design*, 72, 573–582. <http://www.bakker.org/cfm/publications/TransichemeComp1994.pdf>
- Abishek, S., King, A. J. C., Mead-Hunter, R., Golkarfard, V., Heikamp, W., & Mullins, B. J. (2017). Generation and validation of virtual nonwoven, foam and knitted filter (separator/coalescer) geometries for CFD simulations. *Separation and Purification Technology*, 188, 493–507. <https://doi.org/10.1016/j.seppur.2017.07.052>
- Allonneau, C., Olmos, E., Guyot, S., Ferret, E., Gervais, P., & Cachon, R. (2015). Hydrodynamic characterization of a new small-scale reactor mixed by a magnetic bar. *Biochemical Engineering Journal*, 96, 29–37. <https://doi.org/10.1016/j.bej.2014.12.005>
- Berg, P., Voß, S., Saalfeld, S., Janiga, G., Bergersen, A. W., Valen-Sendstad, K., Bruening, J., Goubergrits, L., Spuler, A., Cancelliere, N. M., Steinman, D. A., Pereira, V. M., Chiu, T. L., Tsang, A. C. O., Chung, B. J., Cebal, J. R., Cito, S., Pallarès, J., Copelli, G., ... Beuing, O. (2018). Multiple Aneurysms AnaTomy CHallenge 2018 (MATCH): Phase I: Segmentation. *Cardiovascular Engineering and Technology*, 9(4), 565–581. <https://doi.org/10.1007/s13239-018-00376-0>
- Blanco-Aguilera, R., Lara, J. L., Barajas, G., Tejero, I., & Díez-Montero, R. (2020). CFD simulation of a novel anaerobic-anoxic reactor for biological nutrient removal: Model construction, validation and hydrodynamic analysis based on OpenFOAM®. *Chemical Engineering Science*, 215, 115390. <https://doi.org/10.1016/j.ces.2019.115390>
- Blanco-Aguilera, R., Lara, J. L., Barajas, G., Tejero, I., & Díez-Montero, R. (2020a). Hydrodynamic optimization of multi-environment reactors for biological nutrient removal: A methodology combining computational fluid dynamics and dimensionless indexes. *Chemical Engineering Science*, 224, 115766. <https://doi.org/10.1016/j.ces.2020.115766>
- Blazek, J. (2015). Boundary Conditions. *Computational Fluid Dynamics: Principles and Applications*, 253–281. <https://doi.org/10.1016/B978-0-08-099995-1.00008-7>
- Boccardo, G., Augier, F., Haroun, Y., Ferré, D., & Marchisio, D. L. (2015). Validation of a novel open-source work-flow for the simulation of packed-bed reactors. *Chemical Engineering Journal*, 279, 809-820–820. <https://doi.org/10.1016/j.cej.2015.05.032>
- Boccardo, G., Crevacore, E., Passalacqua, A., & Icardi, M. (2019). Computational analysis of transport in three-dimensional heterogeneous materials. *ArXiv*, 23(1), 1–15. <https://doi.org/10.1007/s00791-020-00321-6>
- Boccardo, G., Sethi, R., & Marchisio, D. L. (2019). Fine and ultrafine particle deposition in packed-bed catalytic reactors. *Chemical Engineering Science*, 198, 290–304. <https://doi.org/10.1016/j.ces.2018.09.024>

- Buwa, V. V., & Ranade, V. V. (2002). Dynamics of gas-liquid flow in a rectangular bubble column: Experiments and single/multi-group CFD simulations. *Chemical Engineering Science*, 57(22–23), 4715–4736. [https://doi.org/10.1016/S0009-2509\(02\)00274-9](https://doi.org/10.1016/S0009-2509(02)00274-9)
- Carvalho, K. Q. de, Salgado, M. T., Passig, F. H., & Pires, E. C. (2008). Avaliação Hidrodinâmica de Reator UASB Submetido a Variação Cíclica de Vazão. *Revista de Engenharia Sanitária e Ambiental*, 13(2), 226–235. <https://doi.org/10.1590/S1413-41522008000200013>
- Díez-Montero, R., De Florio, L., González-Viar, M., Volcke, E. I. P., & Tejero, I. (2015). Feasibility of hydraulic separation in a novel anaerobic-anoxic upflow reactor for biological nutrient removal. *Bioprocess and Biosystems Engineering*, 38(1), 93–103. <https://doi.org/10.1007/s00449-014-1247-9>
- Ellegaard, O., & Wallin, J. A. (2015). The bibliometric analysis of scholarly production: How great is the impact? *Scientometrics*, 105(3), 1809–1831. <https://doi.org/10.1007/s11192-015-1645-z>
- Flores, A., Wang, X., & Nielsen, D. R. (2019). Recent trends in integrated bioprocesses: aiding and expanding microbial biofuel/biochemical production. *Current Opinion in Biotechnology*, 57, 82–87. <https://doi.org/10.1016/j.copbio.2019.02.007>
- Fuess, L. T., de Araújo Júnior, M. M., Garcia, M. L., & Zaiat, M. (2017). Designing full-scale biodigestion plants for the treatment of vinasse in sugarcane biorefineries: How phase separation and alkalization impact biogas and electricity production costs? *Chemical Engineering Research and Design*, 119, 209–220. <https://doi.org/10.1016/j.cherd.2017.01.023>
- Galerkin, Y. B., Voinov, I. B., & Drozdov, A. A. (2017). Comparison of CFD-calculations of centrifugal compressor stages by NUMECA Fine Turbo and ANSYS CFX programs. *IOP Conference Series: Materials Science and Engineering*, 232(1), 012044. <https://doi.org/10.1088/1757-899X/232/1/012044>
- Gavrilescu, M. (2014). Biorefinery Systems: An Overview. In *Bioenergy Research: Advances and Applications* (Vol. 1). <https://doi.org/10.1016/B978-0-444-59561-4.00014-0>
- Gelves, R., Benavides, A., & Quintero, J. C. (2013). Predicción del comportamiento hidrodinámico en el escalado de un reactor de tanque agitado para procesos aerobios, mediante CFD. *Ingeniare*, 21(3), 347–361. <https://doi.org/10.4067/S0718-33052013000300005>
- Gumster, J. Van. (2020). *Blender for dummies*. <https://books.google.com/books?hl=es&lr=&id=gHDJDwAAQBAJ&oi=fnd&pg=PA1&ots=cwPxiWNjGb&sig=RnQ7usHrY8c8xFWVICUQ7RFu2cE>
- H K Versteeg. W Malalasekera. (2016). An Introduction to Computational Fluid Dynamics. In *Educational Building* (Vol. 2, Issue 2). <https://doi.org/10.24114/eb.v2i2.4393>
- Herther, N. K. (2009). Research evaluation and citation analysis: Key issues and implications.

- The Electronic Library*, 27(3), 361–375. <https://doi.org/10.1108/02640470910966835>
- Jordan, C., & Harasek, M. (2019). *OpenFOAM® Basic Training*.
- Kansal, A. K., Joshi, J. B., Maheshwari, N. K., & Vijayan, P. K. (2015). CFD analysis of moderator flow and temperature fields inside a vertical calandria vessel of nuclear reactor. *Nuclear Engineering and Design*, 287, 95–107. <https://doi.org/10.1016/j.nucengdes.2015.02.011>
- Khabibullin, R., Le Huong, T., Ivanchenko, O., & Petrov, A. (2019). Energy efficiency of food production wastewater anaerobic-aerobic treatment. *E3S Web of Conferences*, 140, 01001. <https://doi.org/10.1051/e3sconf/201914001001>
- Kilts, C. D., Wade, A. G., Andersen, H. F., & Schlaepfer, T. E. (2009). Top Value Added Chemicals from Biomass. *Expert Opinion on Pharmacotherapy*, 10(6), 927–936. <https://doi.org/10.1517/14656560902849258>
- Koerich, D. M., Lopes, G. C., & Rosa, L. M. (2020). Numerical study on the hydrodynamics of a fluidized-bed of bioparticles in tapered bioreactors with square shape cross-section. *Brazilian Journal of Chemical Engineering*, 37(1), 101–115. <https://doi.org/10.1007/s43153-020-00010-4>
- Krychowska, A., Kordas, M., Konopacki, M., Grygorcewicz, B., Musik, D., Wójcik, K., Jędrzejczak-Silicka, M., & Rakoczy, R. (2020). Mathematical Modeling of Hydrodynamics in Bioreactor by Means of CFD-Based Compartment Model. *Processes* 2020, Vol. 8, Page 1301, 8(10), 1301. <https://doi.org/10.3390/pr8101301>
- Kumawat, G. L., Kansal, A. K., Maheshwari, N. K., & Sharma, A. (2019). Computational fluid dynamics simulation of flow-mixing and heat transfer in 4 × 4 rod bundle with a twist-vane spacer grid. *Journal of Nuclear Engineering and Radiation Science*, 5(4). <https://doi.org/10.1115/1.4042120>
- Levenspiel, O. (2012). The tracer method. *Fluid Mechanics and Its Applications*, 96, 1–3. https://doi.org/10.1007/978-1-4419-8074-8_1
- Liu, W., Wang, J., Li, C., Chen, B., & Sun, Y. (2019). Using Bibliometric Analysis to Understand the Recent Progress in Agroecosystem Services Research. *Ecological Economics*, 156(December 2017), 293–305. <https://doi.org/10.1016/j.ecolecon.2018.09.001>
- Marcato, A., Boccardo, G., & Marchisio, D. (2021). A computational workflow to study particle transport and filtration in porous media: Coupling CFD and deep learning. *Chemical Engineering Journal*, 417(March), 128936. <https://doi.org/10.1016/j.cej.2021.128936>
- Marroquín-Fandiño, J. E., Ramírez-Acosta, C. M., Luna-Wandurraga, H. J., Valderrama-Rincón, J. A., Cruz, J. C., Reyes, L. H., & Valderrama-Rincon, J. D. (2020). Novel external-loop-airlift milliliter scale bioreactors for cell growth studies: Low cost design, CFD analysis and experimental characterization. *Journal of Biotechnology*, 324, 71–82. <https://doi.org/10.1016/j.jbiotec.2020.09.022>

- Marton, J. M., Felipe, M. G. A., Almeida e Silva, J. B., & Pessoa, A. (2006). Evaluation of the activated charcoals and adsorption conditions used in the treatment of sugarcane bagasse hydrolysate for xylitol production. *Brazilian Journal of Chemical Engineering*, 23(1), 9–21. <https://doi.org/10.1590/S0104-66322006000100002>
- Mekonnen, A. T. (2016). *Modelling a tracer injection and sensor manifold*. July. <http://hdl.handle.net/11250/2420578>
- Meng, X., & Li, Y. (2015). Numerical study of natural convection in a horizontal cylinder filled with water-based alumina nanofluid. *Nanoscale Research Letters*, 10(1), 2–11. <https://doi.org/10.1186/s11671-015-0847-x>
- Mohiuddin, I., & Mathkour, H. (2015). Computational fluid dynamics application tools. *2015 World Congress on Information Technology and Computer Applications, WCITCA 2015*. <https://doi.org/10.1109/WCITCA.2015.7367062>
- Nizami, A. S., Rehan, M., Waqas, M., Naqvi, M., Ouda, O. K. M., Shahzad, K., Miandad, R., Khan, M. Z., Syamsiro, M., Ismail, I. M. I., & Pant, D. (2017). Waste biorefineries: Enabling circular economies in developing countries. *Bioresource Technology*, 241, 1101–1117. <https://doi.org/10.1016/j.biortech.2017.05.097>
- Nogenmyr, K.J., Folkersma, M. (2016). *SwiftBlock*. <https://openfoamwiki.net/index.php/Contrib/SwiftBlock>
- Noll, P., & Henkel, M. (2020). History and Evolution of Modeling in Biotechnology: Modeling & Simulation, Application and Hardware Performance. *Computational and Structural Biotechnology Journal*, 18, 3309–3323. <https://doi.org/10.1016/j.csbj.2020.10.018>
- Otálora Tapiero Fabián Camilo. (2018). *Evaluación de la producción de Biogás en sistemas de dos etapas usando un reactor anaerobio de lecho fijo estructurado (ASTBR) y reactor metanogénico escalonado (RME) alimentado con licor de pentosas*. Universidade Federal de Ouro Preto.
- Paladino, O., & Neviani, M. (2020). Scale-up of photo-bioreactors for microalgae cultivation by π -theorem. *Biochemical Engineering Journal*, 153(October 2019), 107398. <https://doi.org/10.1016/j.bej.2019.107398>
- Passos, F., Cordeiro, P. H. M., Baeta, B. E. L., de Aquino, S. F., & Perez-Elvira, S. I. (2018). Anaerobic co-digestion of coffee husks and microalgal biomass after thermal hydrolysis. *Bioresource Technology*, 253, 49–54. <https://doi.org/10.1016/j.biortech.2017.12.071>
- Pavlostathis, S. G., & Giraldo-Gomez, E. (1991). Kinetics of anaerobic treatment – a critical review. *CRC Critical Reviews in Environmental Control*, 21(January 2013), 411–490. <https://doi.org/10.1080/10643389109388424>
- Peeters, P. T. (2016). CFD Of Multiphase pipe flow: A comparison of solvers. *TU Delft*. <https://repository.tudelft.nl/islandora/object/uuid%3A88a16dcd-f404-4d29-9bad-d3ffd0cc2023>

- Pérez Peña, S. A., & Velandia Blanco, D. (2020). *Evaluación de la producción de biogás en un reactor metanogénico escalonado (RME) usando cenizas de cascarilla de arroz como material adsorbente para diferentes cargas orgánicas (OLR)*. Fundación Universidad de América. <https://repository.uamerica.edu.co/handle/20.500.11839/7796>
- Pouffary, B., Patella, R. F., Reboud, J.-L., & Lambert, P.-A. (2008). Numerical Simulation of 3D Cavitating Flows: Analysis of Cavitation Head Drop in Turbomachinery. *Journal of Fluids Engineering*, 130(6). <https://doi.org/10.1115/1.2917420>
- Pretorius, L., Benade, S. J., & Kruger, S. (2008). Technology forecasting: The case of Computational Fluid Dynamics (CFD). *Proceedings of the 4th IEEE International Conference on Management of Innovation and Technology, ICMIT*, 7–11. <https://doi.org/10.1109/ICMIT.2008.4654328>
- Rahman, K. M., Edwards, D. J., Melville, L., & El-Gohary, H. (2019). Implementation of bioenergy systems towards achieving United Nations' sustainable development goals in rural Bangladesh. *Sustainability (Switzerland)*, 11(14). <https://doi.org/10.3390/su11143814>
- Ranganathan, P., & Sivaraman, S. (2011). Investigations on hydrodynamics and mass transfer in gas–liquid stirred reactor using computational fluid dynamics. *Chemical Engineering Science*, 66(14), 3108–3124. <https://doi.org/10.1016/j.ces.2011.03.007>
- Rivas, J., Sadino-Riquelme, M. C., Garcés, I., Carvajal, A., & Donoso-Bravo, A. (2020a). Spatial and temporal validation of a cfd model using residence time distribution test in a tubular reactor. *Computation*, 8(4), 1–10. <https://doi.org/10.3390/computation8040094>
- Levenspiel, O. (2012). The tracer method. *Fluid Mechanics and Its Applications*, 96, 1–3. https://doi.org/10.1007/978-1-4419-8074-8_1
- Saikia, K., Vallès, M., Fabregat, A., Saez, R., & Boer, D. (2020). A bibliometric analysis of trends in solar cooling technology. *Solar Energy*, 199(November 2019), 100–114. <https://doi.org/10.1016/j.solener.2020.02.013>
- Solé-Bundó, M., Passos, F., Romero-Güiza, M. S., Ferrer, I., & Astals, S. (2019). Co-digestion strategies to enhance microalgae anaerobic digestion: A review. In *Renewable and Sustainable Energy Reviews* (Vol. 112, pp. 471–482). Elsevier Ltd. <https://doi.org/10.1016/j.rser.2019.05.036>
- Teixeira, T. A., & Souza, J. A. (2019). Computational modeling of resin infusion process applied to small vessel hulls. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 41(10), 1–9. <https://doi.org/10.1007/s40430-019-1936-z>
- Tu, J., Yeoh, G.-H., & Liu, C. (2013). Practical Guidelines for CFD Simulation and Analysis. *Computational Fluid Dynamics*, 219–273. <https://doi.org/10.1016/B978-0-08-098243-4.00006-8>

- Välikangas, T., & Karvinen, R. (2018). Conjugated Heat Transfer Simulation of a Fin-and-Tube Heat Exchanger. *Heat Transfer Engineering*, 39(13–14), 1192–1200. <https://doi.org/10.1080/01457632.2017.1363628>
- Xiang, J., Zhu, L., & Luo, Z. (2020). Application of kinetics and computational fluid dynamics in pinene isomerization. *Chinese Journal of Chemical Engineering*, 28(8), 2111–2120. <https://doi.org/10.1016/j.cjche.2020.03.021>

ANEXOS

Scripts de simulación:

Velocidad

```
/*-----* C++ *-----*\
=====
\\ / F i e l d      | OpenFOAM: The Open Source CFD Toolbox
\\ / O p e r a t i o n | Website: https://openfoam.org
\\ / A n d           | Version: dev
\\ \ M a n i p u l a t i o n |
\*-----*/
FoamFile
{
  version 2.0;
  format  ascii;
  class   volVectorField;
  object  U;
}
// ***** //

dimensions  [0 1 -1 0 0 0];

internalField  uniform (0 0 0);

boundaryField
{
  inlet
  {
    type      fixedValue;
    value     uniform (3 0 0);
  }
  outlet
  {
    type      zeroGradient;
  }
  walls
  {
    type      fixedValue;
    value     uniform (0 0 0);
  }
}
// ***** //
```

Densidad:

```
/*-----* C++ *-----*\
=====
\ \ / F i e l d      | OpenFOAM: The Open Source CFD Toolbox
\ \ / O p e r a t i o n | Website: https://openfoam.org
\ \ / A n d          | Version: dev
 \ \ M a n i p u l a t i o n |
/*-----*/
FoamFile
{
  version 2.0;
  format  ascii;
  class  volScalarField;
  object  p;
}
//*****//

dimensions  [0 2 -2 0 0 0];

internalField  uniform 0;

boundaryField
{
  inlet
  {
    type      fixedValue;
    value     uniform 0.3;
  }
  outlet
  {
    type      zeroGradient;
  }
  walls
  {
    type      fixedValue;
    value     uniform 0;
  }
}
//*****//
```


blockMeshDict- Geometría:

```
/*-----*- C++ -*-----*/

// File was generated by SwiftBlock, a Blender 3D addon.

FoamFile
{
  version 2.0;
  format ascii;
  class dictionary;
  object blockMeshDict;
}
// ***** //

deltaT 1;

writeInterval 1;

// ***** //

geometry
{
}
vertices
(
  (0.05700000002980232 0.054605040699243546 0.5596299767494202)
  (0.05700000002980232 0.054605040699243546 0.29962998628616333)
  (0.05700000002980232 -0.054605040699243546 0.5596299767494202)
  (0.05700000002980232 -0.054605040699243546 0.29962998628616333)
  (-0.05700000002980232 0.054605040699243546 0.5596299767494202)
  (-0.05700000002980232 0.054605040699243546 0.29962998628616333)
  (-0.05700000002980232 -0.054605040699243546 0.5596299767494202)
  (-0.05700000002980232 -0.054605040699243546 0.29962998628616333)
  (0.0064999996684491634 0.0593949556350708 0.5153300166130066)
  (-0.0064999996684491634 0.0593949556350708 0.5153300166130066)
  (0.0064999996684491634 0.0593949556350708 0.5283299684524536)
  (-0.0064999996684491634 0.0593949556350708 0.5283299684524536)
  (0.0064999996684491634 0.10729411244392395 0.5153300166130066)
  (-0.0064999996684491634 0.10729411244392395 0.5153300166130066)
  (0.0064999996684491634 0.10729411244392395 0.5283299684524536)
  (-0.0064999996684491634 0.10729411244392395 0.5283299684524536)
  (-0.029999999329447746 0.02873949520289898 0.2946299910545349)
  (-0.029999999329447746 -0.02873949520289898 0.2946299910545349)
  (0.029999999329447746 0.02873949520289898 0.2946299910545349)
  (0.029999999329447746 -0.02873949520289898 0.2946299910545349)
  (-0.029999999329447746 0.02873949520289898 0.1446299999523163)
  (-0.029999999329447746 -0.02873949520289898 0.1446299999523163)
  (0.029999999329447746 0.02873949520289898 0.1446299999523163)
)
```



```

)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
// block id 4
hex (20 21 23 22 16 17 19 18) (10 10 10) edgeGrading (
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
)

```



```

// block id 6
hex (28 29 31 30 24 25 27 26) (10 10 10) edgeGrading (
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
)
// block id 7
hex (32 33 35 34 28 29 31 30) (10 10 10) edgeGrading (
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
)

```

```

)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
// block id 8
hex (36 37 39 38 32 33 35 34) (10 10 10) edgeGrading (
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
)

```



```
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
(
(0 0 1) (1 10 1) (0 0 1)
)
);
```

```
patches
(
  patch inlet
  (
    (36 38 39 37)
  )
  patch outlet
  (
    (13 15 14 12)
  )
  wall walls
  (
    (0 4 6 2)
    (3 2 6 7)
    (7 6 4 5)
    (3 7 17 19)
    (1 0 2 3)
    (4 0 10 11)
    (11 10 14 15)
    (0 1 8 10)
    (5 4 11 9)
    (1 5 9 8)
    (10 8 12 14)
    (9 11 15 13)
    (8 9 13 12)
    (19 17 21 23)
    (7 5 16 17)
    (1 3 19 18)
    (5 1 18 16)
    (22 23 27 26)
    (17 16 20 21)
    (18 19 23 22)
    (16 18 22 20)
    (27 25 29 31)
    (20 22 26 24)
    (23 21 25 27)
    (21 20 24 25)
  )
)
```

(31 29 33 35)
(25 24 28 29)
(26 27 31 30)
(24 26 30 28)
(34 35 39 38)
(29 28 32 33)
(30 31 35 34)
(28 30 34 32)
(32 34 38 36)
(35 33 37 39)
(33 32 36 37)

)
);

controlDict

```
/*-----*- C++ -*-----*\
=====
\ \ / F i e l d      | OpenFOAM: The Open Source CFD Toolbox
\ \ / O p e r a t i o n | Website: https://openfoam.org
\ \ / A n d          | Version: dev
 \ \ M a n i p u l a t i o n |
/*-----*/
FoamFile
{
    version    2.0;
    format     ascii;
    class      dictionary;
    location   "system";
    object     controlDict;
}
//*****//

application    simpleFoam;

startFrom      startTime;

startTime      0;

stopAt         endTime;

endTime        65;

deltaT         0.1;

writeControl   timeStep;

writeInterval  5;

purgeWrite     0;

writeFormat    ascii;

writePrecision 6;

writeCompression off;

timeFormat     general;

timePrecision  6;

runTimeModifiable true;

cacheTemporaryObjects
(
    kEpsilon:G
);
```

```
functions
{
  #includeFunc streamlines
  #includeFunc writeObjects(kEpsilon:G)
}

// ***** //
```