ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA MECÁNICA

"ESTUDIO NUMÉRICO DE FLUJO PARTÍCULADO DENTRO DE UNA TURBINA TIPO FRANCIS USANDO OPENFOAM"

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO MECÁNICO

NARVÁEZ QUELAL MATEO ANDRÉS

mateo.narvaez@epn.edu.ec

DIRECTOR:

Ing. HIDALGO DÍAZ VÍCTOR HUGO, D.Sc.

victor.hidalgo@epn.edu.ec

CODIRECTOR:

Ing. VALENCIA TORRES ESTEBAN ALEJANDRO, Ph.D.
esteban.valencia@epn.edu.ec

Quito, mayo 2021

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por **MATEO ANDRÉS NARVÁEZ QUELAL**, bajo mi supervisión.

Firmado electrónicamente por:
VICTOR HUGO HIDALGO DIAZ

Ing. Víctor Hugo Hidalgo Díaz D.Sc.

DIRECTOR DE PROYECTO

Ing. Esteban Alejandro Valencia Torres Ph.D.

CODIRECTOR DE PROYECTO

DECLARACIÓN

Yo, **MATEO ANDRÉS NARVÁEZ QUELAL**, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondiente a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

Firmado electrónicamente por:
MATEO ANDRES NARVAEZ QUELAL

Mateo Andrés Narváez Quelal

DEDICATORIA

A mis padres, por enseñarme a enfrentar al mundo de la manera correcta, por mostrarme que solo el trabajo duro nos volverá mejores, por estar conmigo en cada paso respaldándome y llenándome de amor. Esta tesis la desarrollé yo, pero les pertenece a ustedes, sin su guía y apoyo difícilmente hubiera podido conseguir este objetivo. Los amo padres queridos.

A mis hermanos, ustedes fueron la principal motivación para salir adelante en este proyecto, saber que son mis hermanitos quienes vienen siguiendo mis pasos es una gran responsabilidad que la llevo orgulloso porque estoy orgulloso de ustedes y de lo que pueden alcanzar. Para mi este es solo una meta lograda más, sé que ustedes en su momento lo van a hacer mejor aún y voy a estar ahí junto a ustedes cuando eso suceda. Ustedes son los mejores.

A mis abuelitos Manuelito y Rosita, gran parte de este sueño logrado nace de ustedes, hoy no están físicamente para disfrutarlo con nosotros, pero están presentes en mis acciones, en mi manera de ser y en mi forma de pensar, su enseñanza me acompaña en cada momento de mi vida y siempre los recordaré con orgullo y gran amor.

AGRADECIMIENTO

A mis padres, que me han enseñado con el ejemplo y me han convertido en una persona de bien y de aporte para la sociedad, Este trabajo no habría podido ser llevado a cabo sin su ayuda, guía y apoyo.

A mis hermanos, por darme la motivación necesaria para ser mejor cada día, y siempre buscar ser el mejor.

A la Escuela Politécnica Nacional, por ser la encargada de forjarme correctamente en el ámbito profesional.

A la Facultad de Ingeniería Mecánica, por ser la mejor facultad de mundo y acogerme durante toda mi etapa universitaria .

A mis amigos, mi grupo “LOS MAMADITOS” con quien tuve el placer de compartir esta carrera, quiero que sepan no habría sido posible concluir esta etapa sin todos los buenos y malos momentos que pasamos juntos. Por todas las experiencias que nos volvieron un grupo lleno de apoyo y compañerismo, los llevo siempre en mi corazón, gracias por su amistad.

A mi gran amigo Xavier Amán, no existen palabras para agradecer tu tiempo tu paciencia y tu amistad incondicional, no recuerdo un solo momento en la universidad en el cual no haya contado con tu lealtad y confianza, con toda sinceridad fuiste un soporte fundamental para que todo esto sea posible.

Al Ing. Stalyn Simbaña, por haberme mostrado el mundo del CFD, por todo el tiempo, colaboración y guía prestada durante el desarrollo de esta tesis y por una amistad sincera.

A mi director de tesis, Ing. Víctor Hidalgo D.Sc. por tener siempre una buena predisposición a guiarme durante el proceso de desarrollo de este trabajo de titulación.

Índice de contenido

CERTIFICACIÓN i

DECLARACIÓN ii

DEDICATORIA iii

AGRADECIMIENTO iv

Índice de contenido v

ÍNDICE DE FIGURAS viii

ÍNDICE DE TABLAS x

RESUMEN xi

ABSTRACT xii

INTRODUCCIÓN 1

Pregunta de investigación 3

Objetivo general 3

Objetivos específicos 3

1 MARCO TEÓRICO 4

1.1 Antecedentes 4

1.2 Turbina Francis. 5

1.2.1 Funcionamiento 6

1.2.2 Partes de la turbina Francis. 6

1.2.3 Ubicación de la Central San Francisco. 7

1.3 Material granular y sus características 7

1.3.1 Tamaño del grano. 8

1.3.2 Forma del grano 8

1.3.3 Presencia de material granular en flujo hídricos 9

1.4 Dinámica de Fluidos Computacional (CFD) 11

1.4.1 Ecuaciones que gobiernan el CFD 11

1.4.2 Metodología del CFD 12

1.4.3 El CFD y su aplicación en turbinas hidráulicas 13

1.4.4 OpenFOAM 13

1.5 Mallado estructurado 14
1.5.1 Criterios de Calidad de Malla 15
1.6 DEM 16
1.6.1 Suposiciones del DEM. 16
1.6.2 Ecuaciones que gobiernan el Método del Elemento Discreto. 17
1.6.3 Contacto entre partículas 18
1.6.4 Modelos de Contacto. 18
1.6.5 LIGGGTHS ® 22
1.7 CFD-DEM 23
1.7.1 Procedimiento de acoplamiento CFD-DEM 24
2 METODOLOGIA 27
2.1 Geometría 28
2.2 Optimización de Mallado estructurado 30
2.3 Importación y fusión de la malla 33
2.4 MRF 35
2.5 Calidad de malla. 36
2.6 Simulación estable método SIMPLE 37
2.6.1 Condiciones de operación de la central hidroeléctrica. 38
2.6.2 Condiciones de Borde y de Frontera. 38
2.6.3 Solver (simpleFoam) 41
2.7 Simulación transitoria método PISO 42
2.7.1 Solver (pisoFoam). 43
2.7.2 Condiciones de Borde 44
2.7.3 Número de Courant. 45
2.8 Simulación DEM 46
2.8.1 Material granular en la zona de estudio. 46
2.8.2 Características del material particular 47
2.8.3 Preproceso de DEM 49
2.8.4 Modelo de desgaste 51

2.9 Acople CFD-DEM: 52
2.9.1 Estructura de archivos. 52
2.9.2 Condiciones iniciales y de borde 55
2.9.3 Ajustes de Acople 55
2.9.4 Paralelización y recurso computacional 56
3 RESULTADOS Y DISCUSIÓN 58
3.1 Independencia de malla. 58
3.2 Calidad de malla 60
3.3 Residuales 61
3.4 Simulación con método SIMPLE 64
3.5 Simulación método PISO 68
3.6 Acoplamiento CFD-DEM 72
4 CONCLUSIONES 82
4.1 Conclusiones 82
4.2 Trabajos futuros. 83
Referencias Bibliográficas 85
ANEXOS 90
ANEXO I. SCRIPT PARA AUTOPOCISIONAMIENTO Y COSTURA DE DOMINIOS COMPUTACIONALES 90
ANEXO II EJECUTABLE CFDEM 92
ANEXO III ligggthsCommands 94
ANEXO IV couplingProperties 95
ANEXO V in.liggghts_init 99
ANEXO VI in.liggghts_run 101
ANEXO VII in.liggghts_restart 103

ÍNDICE DE FIGURAS

Figura 1.1 Geometría y partes de la turbina Francis de la central hidroeléctrica San Francisco. 6
Figura 1.2 Ubicación de la central San Francisco en la cuenca del rio Pastaza. 7
Figura 1.3 Comparación de mallas estructurada y no estructurada. 14
Figura 1.4 Deformación de una partícula y su equivalencia en superposición. 19
Figura 1.5 Modelo de contacto de fuerzas resorte amortiguador. 19
Figura 1.6 Esfera en contacto con un plano. 21
Figura 1.7 Contacto en un solo punto entre partículas esféricas de material y tamaño diferente. 21
Figura 1.8 Esquema gráfico del cálculo conjunto del método de acoplamiento CFD-DEM. 26
Figura 2.1 Metodología del presente caso de estudio. 28
Figura 2.2 Modelo geométrico del rodete a) Mora y b) Guascal. 29
Figura 2.3 Comparativa entre geometría del tubo de descarga. 29
Figura 2.4 a) Refinamiento logrado por Velasco. b) Malla modificada. 30
Figura 2.5 Comparativa del Spiral case. 31
Figura 2.6 Comparativa del Guide vane. 31
Figura 2.7 Comparativa del Runner. 32
Figura 2.8 Comparativa del Draft Tube. 32
Figura 2.9 Estructura de archivos necesarios para la importación de la malla a OpenFOAM. 33
Figura 2.10 Detalle de la diferencia entre malla multirregión (a) y malla de región única (b). 35
Figura 2.11 Zona MRF en el dominio computacional. 36
Figura 2.12 Diagrama de flujo algoritmo SIMPLE. 37
Figura 2.13 Diagrama de flujo algoritmo PISO. 44
Figura 2.14 Dominio computacional DEM. 50
Figura 2.15 Árbol de archivos para el caso de simulación de la turbina tipo Francis con flujo particular. 53
Figura 3.1 Valores de torque en distintas mallas. 58
Figura 3.2 Residuales de presión de los casos de independencia de malla. 59
Figura 3.3 Residuales de la simulación numérica de algoritmo SIMPLE. 61
Figura 3.4 Residuales de la simulación numérica de algoritmo PISO. 62
Figura 3.5 Residuales de la simulación numérica de algoritmo CFD-DEM. 63
Figura 3.6 Perfiles de velocidad en salida del runner: a) Malla optimizada b) Malla Fina [7]. 64

Figura 3.7 Perfiles de velocidad en el plano XY: a) Malla optimizada b) Malla fina [7]. 65
Figura 3.8 Perfiles de presión vista superior del rodete y alabes fijos: a) Malla optimizada b) Malla fina [7]. 66
Figura 3.9 Perfiles de presión para el rodete y uno de sus alabes: a) Malla optimizada b) Malla fina [7]. 66
Figura 3.10 Perfiles de presión en el plano YZ: a) Malla optimizada b) Malla fina [7]. 67
Figura 3.11 Courant máximo en cada intervalo de tiempo simulado. 68
Figura 3.12 Courant promedio en cada intervalo de tiempo simulado. 69
Figura 3.13 Tiempo de ejecución vs tiempo simulado. 70
Figura 3.14 Distribución de Courant para el intervalo 100e-6. 71
Figura 3.15 Distribución de Courant en el conjunto Rodete-Alabes directores para el intervalo 100e-6. 71
Figura 3.16 Distribución de Courant en corte del conjunto Rodete-Alabes directores para el intervalo 100e-6. 72
Figura 3.17 Distribución de partículas iniciales en el domino computacional. 73
Figura 3.18 Desarrollo de partículas a través del dominio computacional, Vista superior. 75
Figura 3.19 Desarrollo de partículas a través del dominio computacional, Vista lateral. 76
Figura 3.20 Desarrollo de partículas a través del dominio computacional, Vista isométrica. 77
Figura 3.21 Zonas de desgaste por material particulado en la caracola. 78
Figura 3.22 Zonas de desgaste por material particulado en los alabes directores. 78
Figura 3.23 Zonas de desgaste por material particulado en el rodete. 79
Figura 3.24 Zonas de desgaste por material particulado en los alabes del rodete. 80
Figura 3.25 Comparación de daños en el material por desgaste en alabes directrices obtenidos mediante simulación numérica. 80
Figura 3.26 Comparación de daños en el material por desgaste en el rodete obtenidos mediante simulación numérica. 81
Figura 3.27 Comparación de daños en el material por desgaste en el rodete obtenidos mediante simulación numérica. 81

ÍNDICE DE TABLAS

Tabla 1.1 Clasificación granulométrica según la *American Geophysical Union.* 10
Tabla 1.2 Metodología CFD..... 13
Tabla 1.3 Descripción general de las ecuaciones y herramientas empleadas en el método CFD-DEM..... 24
Tabla 2.1 Comparación entre cantidad de elementos de la malla de Velasco y la desarrollada. para este caso..... 33
Tabla 2.2 Datos estadísticos de la malla. 36
Tabla 2.3 Condiciones de operación de la turbina Francis..... 38
Tabla 2.4 Valores experimentales de potencia y rendimiento de la central San Francisco a diferentes caudales. 39
Tabla 2.5 Condiciones de borde (Velocidad y Presión)..... 41
Tabla 2.6 condiciones de borde (k, epsilon y nut). 41
Tabla 2.7 Ajustes de Solvers para las variables simuladas en estado estable. 42
Tabla 2.8 Tamaño y densidad de partículas presentes en la cuenca del rio Pastaza.. 47
Tabla 2.9 Relación de Poisson para diferentes sedimentos..... 48
Tabla 2.10 Propiedades del material particular. 49
Tabla 2.11 Coordenadas del dominio computacional de DEM..... 50
Tabla 2.12 Estadísticas de las superficies tipo Wall..... 51
Tabla 2.13 condiciones de borde para fase solida en CFD (Ksl, Us y voidFraction).... 55
Tabla 3.1 Valores de estudio para independencia de malla..... 59
Tabla 3.2 Valor Ω para distintos casos de simulación..... 60

RESUMEN

En este trabajo de titulación se desarrolló una metodología que permite estudiar numéricamente el funcionamiento de una turbina tipo Francis con flujo particulado en software libre y de código abierto. Para lo cual se parte de la optimización del dominio computacional generado a partir de la geometría de una de las turbinas de la central San Francisco en la provincia de Tungurahua. Se validó los resultados obtenidos con esta malla realizando simulaciones en estado estable del funcionamiento en condiciones de operación normales utilizando el algoritmo SIMPLE. Una vez que se corroboró la calidad de estos resultados se reprodujo el fenómeno en estado transitorio con ayuda del algoritmo PISO y se calibró la simulación poder utilizarla en el método acoplado CFD-DEM que calcula simultáneamente el método de mecánica de fluidos computacional (CFD) con el método de elemento discreto (DEM) permitiendo analizar los efectos del fluido sobre el material particular y viceversa. Se incluyó también el análisis del desgaste debido al impacto de las partículas en la superficie del rodete y de los alabes directores. Este estudio se realizó Enel software *CFDEM coupling,* que se encarga de comunicar a los resultados obtenidos por OpenFOAM (CFD) con los calculados por LIGGGTHS (DEM) durante de la simulación, tanto *CFDEM coupling*, LIGGGTHS y OpenFOAM son programas de código abierto.

Palabras clave: Máquinas hidráulicas, CFD, DEM, OpenFOAM, LIGGGHTS, Cálculo en paralelo.

ABSTRACT

The present work develops a methodology that allows the numerical study of the Francis turbine with particulate flow in free and open-source software. The first step to accomplish that was the optimization of the computational domain generated from the geometry of one of the turbines of the San Francisco power plant in the province of Tungurahua, obtaining results that were validated by performing steady-state simulations of the turbine operating under normal charge conditions using the SIMPLE algorithm. Once the quality of these results was confirmed, the phenomenon was reproduced in a transient state using the PISO algorithm. Then, the simulation was calibrated to be able to use it in the coupled CFD-DEM method that simultaneously calculates the computational fluid mechanics (CFD) method with the discrete element method (DEM) allowing to analyze the effects of the fluid on the granular material and vice versa. The analysis of the wear due to the impact of the particles on the surface of the impeller and the guide vanes was also included. This study was carried out in the "CFDEM coupling" software, which is responsible for communicating the results obtained by OpenFOAM (CFD) with those calculated by LIGGGTHS (DEM) during the simulation, CFDEM coupling, LIGGGTHS and OpenFOAM are open-source software.

Keywords: Hydraulic machines, CFD, DEM, OpenFOAM, LIGGGHTS, Parallel calculation.

ESTUDIO NUMÉRICO DE FLUJO PARTÍCULADO DENTRO DE UNA TURBINA TIPO FRANCIS USANDO OPENFOAM.

INTRODUCCIÓN

El uso de turbinas hidráulicas como alternativa para generación de energía eléctrica en Ecuador, presenta una gran cantidad de ventajas debido a que sus afluentes poseen caudales elevados y cabezas de altura naturales al estar ubicados en un sistema de cordilleras con numerosos desniveles. Además, la energía hidráulica presenta una forma viable y óptima de producir electricidad por su alto nivel de disponibilidad tecnológica y bajo impacto ambiental, sin embargo también representan grandes desafíos de diseño [1].

Por tal motivo, el gobierno del país con el objetivo de cambiar la matriz energética y lograr un mayor aprovechamiento, ha invertido en obras emblemáticas que contribuyen al aumento de la producción de energía que no dependa de la quema de combustibles fósiles, habiendo logrado hasta la fecha una participación del 60.75% en la potencia efectiva de generación de energía eléctrica a partir de fuentes renovables, en este porcentaje la energía hidráulica contribuye casi la totalidad de la generación con un aporte de 5.237,47 mega watts (MW) según el informe emitido por la ARCONEL en abril del 2021 [2]. En Ecuador existen 16 centrales de mediana y gran capacidad, así como 39 centrales pequeñas. La tecnología empleada para la generación de energía hidroeléctrica en las centrales La Sopladora, San Francisco, Toachi Pilatón y Quijos corresponde a la utilización de turbinas tipo Francis [3], lo que denota un amplio campo de aplicación de este tipo de turbomaquinaria en la generación eléctrica del Ecuador.

Uno de los problemas más habituales en esta tecnología de generación eléctrica, es el desgaste por abrasión debido al material granular presente en el caudal de los afluentes de agua. Estos sedimentos destruyen las piezas del equipo hidroeléctrico acortando la vida útil de componentes de alto costo de mantenimiento y reemplazo. Provocando un déficit productivo poniendo en riesgo la confiabilidad y estabilidad operativa [4].

En el Ecuador, los principales afluentes de generación hidroeléctrica tienen la presencia significativa de residuos sólidos urbanos, material vegetal, detritos volcánicos y sedimentos en suspensión que son arrastrados por el río y llegan hasta la toma en centrales hidroeléctricas [5]. Cuando las condiciones de turbulencia y velocidad de flujo no alcanzan a arrastrar los sedimentos de mayor tamaño, estos se quedan en el lecho de embalse. Sin embargo, existen partículas que según su composición permanecen en

suspensión o flotando y son succionadas hacia el mecanismo provocando daños en el material progresivamente. Tomando en cuenta este aspecto, paralizar y desarmar una turbina en funcionamiento periódicamente para estudiar el desgaste por material particular, representaría un costo elevado, una pérdida de recursos además del desabastecimiento de energía eléctrica para los sectores que dependen de esta central de generación eléctrica. Por tal motivo es importante proponer una alternativa y estudiar este fenómeno desde una perspectiva diferente que no produzca tantas afectaciones.

Con el avance continuo de la tecnología, la dinámica computacional de fluidos (CFD), los métodos numéricos y la capacidad de procesamiento de los ordenadores; es posible simular numéricamente este tipo de fenómenos de alta complejidad, con el objetivo de predecir zonas de desgaste debido al flujo con material particulado en una turbina hidráulica [6]. A este trabajo de tesis le preceden los resultados obtenidos por Mora [5] y Velasco [7],en donde se lleva a cabo un estudio numérico de la eficiencia y desarrollo de una malla estructurada perteneciente a una turbina de generación hidroeléctrica con condiciones y características de funcionamiento similares a la turbina tipo Francis N.2 de la central hidroeléctrica San Francisco, situada en la provincia de Tungurahua.

Cabe recalcar que los trabajos anteriores, fueron desarrollados en un entorno de software comercial con una licencia privada, que limita su desarrollo y estudio posterior. Por tal motivo, Velasco [7] centra el desarrollo de su estudio en un ambiente de software de código abierto, logrando optimizar recursos computacionales, mejorar los resultados y ampliando el campo de estudio.

El paquete de software OpenFOAM se presenta como una de las principales ventajas al poseer un código fuente que puede ser usado por otros desarrolladores para ampliar sus características y de este modo replicar fenómenos cada vez más complejos. Por este motivo, OpenFOAM ha logrado ganar espacio en la industria y en la academia, con estudios relacionados en la mecánica de fluidos, transferencia de calor, etc. [8].

Considerando la libertad esencial del software para “Estudiar el código fuente de un programa, modificarlo y mejorarlo” [9], nace *CFDEM Project* que usa el código de Open FOAM junto a otro software de simulación de partículas con método de elementos discretos de código abierto conocido como LIGGGHTS, para lograr un paquete conjunto *open source* dedicado mecánica de fluidos y ciencia de partículas. Este programa será utilizado para este estudio con la finalidad de construir métodos de vanguardia y ventajas comerciales que permita el estudio de fluidos con material particulado para la optimización del diseño de turbinas hidráulicas y su uso en los afluentes del Ecuador.

Pregunta de investigación

Se puede estudiar numéricamente el flujo particulado dentro de una turbina tipo Francis usando OpenFOAM

Objetivo general

Estudiar numéricamente el flujo particulado dentro de una turbina tipo Francis usando OpenFOAM.

Objetivos específicos

- Modelar en 3D y simular la geometría de la turbina desarrollada en estudios previos.
- Realizar un mallado 3D para estudio del estado transitorio del flujo particulado en una malla fija.
- Simular flujo particulado en estado transitorio mediante modelos matemáticos incluidos en OpenFOAM.
- Evaluar el modelo matemático utilizado para inclusión de partículas a la simulación.
- Validar los resultados de manera cualitativa con datos obtenidos del estado del arte.

1 MARCO TEÓRICO

1.1 Antecedentes

La energía eléctrica comprende un aspecto demasiado importante en el desarrollo de la humanidad y en varios aspectos que derivan de ella. Desde hace un siglo, el Ecuador ha buscado alternativas para el abastecimiento de energía eléctrica a partir de recursos renovables y no renovables. Sin embargo, no todas las fuentes de generación energética han sido igual de eficientes a nivel nacional, que además de provocar caídas significativas en el desarrollo social e industrial del país, han traído consigo consecuencias ambientales irremediables.

El sector energético del Ecuador se ha visto obligado a buscar alternativas mucho más limpias. El cambio de la matriz energética a partir de recursos hídricos en el país se impulsó desde el 2007 con la construcción masiva de centrales hidroeléctricas a gran escala que buscaron aprovechar este recurso como impulsor de la generación de energía eléctrica, con el objetivo de dejar atrás el uso de combustibles fósiles. La inversión pública y privada conformó un factor clave para el desarrollo de proyectos hidroeléctricos que poco a poco incorporaron tecnologías y fuentes de producción más limpias al sistema de producción nacional. Actualmente, las centrales hidroeléctricas constituyen el 92% [2] de la generación de energía eléctrica de país, por lo que la aplicación de tecnologías y estudios que mejoren su proceso, son cada vez más importantes. Las centrales hidroeléctricas en el Ecuador cada vez buscan mejorar sus sistemas de diseño, operación y mantenimiento que además de reducir costos a nivel de proyecto, generan una mayor capacidad hidroeléctrica convirtiéndolos en proyectos mucho más eficientes.

La Agencia de Regulación y Control de Electricidad (ARCONEL) de Ecuador realizó un informe estadístico en abril del 2020, en el que muestra que la energía generada por centrales hidroeléctricas en el país representa el 58.45% (5.076,4 MW) de total Potencial nominal del país [2]. En Latinoamérica se estima que la demanda eléctrica crecerá entre 2.6% y 3.7% anualmente hasta el 2040 [10]. Esto encamina a la tecnología de generación eléctrica hacia el desarrollo de métodos más eficientes y baratos para cubrir la demanda futura. La Agencia Internacional de Energías Renovables (IRENA), en el 2018 menciona que, las hidroeléctricas son una fuente de suministro eléctrico más baratas en comparación con las energías fósiles y más barata también que la energía eólica y solar [11]. Las turbinas tipo Francis, Pelton y Kaplan, son 3 tipos de turbinas que destacan por su alto uso comercial, sin embargo, una turbina Francis con diseño

optimizado es capaz de capturar sobre el 90% [12] de la energía del agua por lo que es preciso promover el desarrollo de esta tecnología en el país.

En la etapa de diseño de una turbina es difícil considerar los sedimentos existentes en los causes de los ríos, debido a la complejidad de predecir como este material particulado va a afectar a los componentes de la turbina con el paso del tiempo. En 2010 estudios realizados sobre de sedimentos sólidos y flotantes para Hidroagoyán, Hidropastaza, San Francisco y Baños de agua santa en Ecuador revelaron que el rio Pastaza, uno de los principales afluentes de generación eléctrica, mantiene una cantidad de residuos sólidos urbanos, material vegetal, detritos volcánicos y sedimentos en suspensión que son arrastrados por el Río Pastaza y llegan hasta la Toma de las Centrales de generación hidroeléctrica. Estos residuos ocasionan paralizaciones en el normal funcionamiento y un deterioro acelerado de los equipos de generación, así como daños en los sistemas auxiliares, tales como el sistema de agua de enfriamiento debido a que el agua, proviene del mismo rio [5]. Esto obliga a implementar modelos matemáticos de flujo particulado mediante técnicas de dinámica de fluidos computacionales (CFD) para analizar los efectos Hidro-abrasivos en este tipo de turbinas y posteriormente desarrollar técnicas para su prevención.

Uno de los principales problemas existentes en la actualidad para la investigación que hace uso de CFD es el uso de software comercial, pues esto limita los estudios debido a la perdida de la libertad de modificar las modelos matemáticas existentes, también se limita la aplicación o implementación de nuevos modelos que no se encuentren en sus librerías, lo que puede restringir el libre entendimiento de las consideraciones matemáticas y físicas tomadas para reproducir el fenómeno. Además, que las licencias de estos programas tienen costos muy elevados, que no siempre pueden ser costeadas por los presupuestos provistos para investigación. Por tal motivo se plantea en este trabajo hacer uso de las ventajas del software libre encaminando el tema de estudio al uso de OpenFOAM.

1.2 Turbina Francis.

Conocidas también como turbinas de admisión total ya que el rodete mantiene contacto con el fluido en todo momento. Esta turbina tiene la particularidad que el agua cambia su dirección de flujo a medida que pasa a través de ella. El fluido ingresa por la caracola de manera radial, fluye hacia su eje, e inmediatamente después de impactar al rodete de la turbina sale en dirección axial al eje, por tal motivo esta turbina se la cataloga también como una turbina de fujo mixto.

1.2.1 Funcionamiento

El flujo de agua entra a la turbina por la carcasa en espiral, fluye hacia los alabes fijos que se encuentran en asegurados en una sola dirección en la estructura de la caracola, paso seguido se dirigen por los alabes directrices hacia el rodete. La forma de espiral con áreas transversales de radio decreciente a lo largo de la trayectoria produce una distribución uniforme de velocidad en los alabes fijos [13]. Los alabes directrices conducen el agua con una dirección tangencial hacia los alabes del rodete, el impacto del flujo contra el borde de ataque de los alabes del rodete es lo que genera el movimiento de rotación. Tras abandonar los alabes del rodete habiendo transferido la energía hacia el eje, el flujo pasa hacia el tubo de descarga que es un difusor que ayuda a recuperar la energía perdida en el rodete con un incremento de presión para incrementar la cabeza hidráulica [14].

1.2.2 Partes de la turbina Francis.

En la Figura 1.1 se detallan las partes de una turbina tipo Francis de la central hidroeléctrica San Francisco ubicada entre la cuenca media y baja del Río Pastaza, municipio de Baños de Agua Santa, Provincia de Tungurahua, región central del Ecuador.

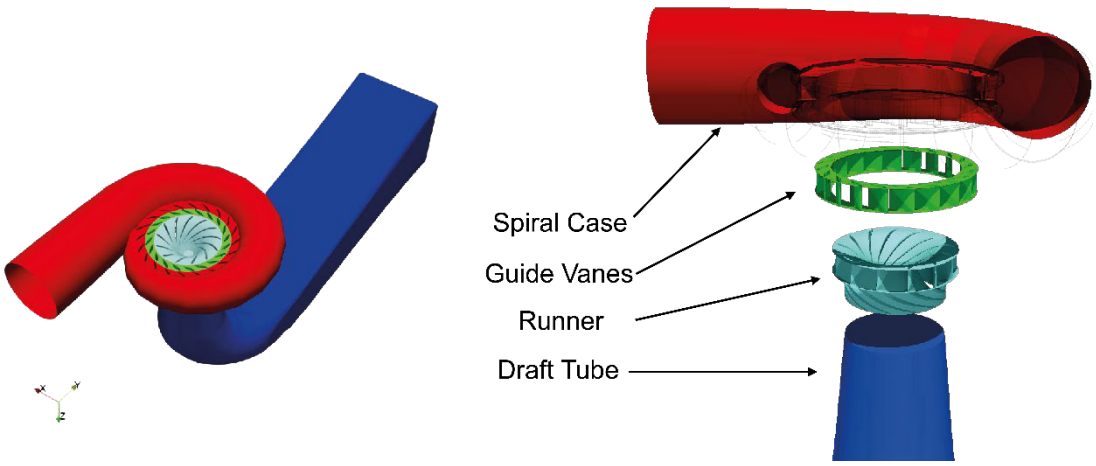

Figura 1.1 Geometría y partes de la turbina Francis de la central hidroeléctrica San Francisco. (Fuente: Propia).

El Rodete está diseñado para extraer la máxima cantidad de energía del agua que fluye a través de él. El diseño se basa en datos como la altura del salto y al caudal especifico de la zona. Cuando la turbina está bien diseñada alcanza a capturar entre el 90 y 95% de la energía del agua y tiene su mejor eficiencia para saltos que van desde los 50 hasta los 700 metros sin embargo ofrece su mejor rendimiento entre los 100 y 300 metros [12].

Dadas estas condiciones de trabajo, la tecnología de la turbina Francis contribuye casi un 60% de la capacidad efectiva de la energía hidráulica producida a nivel mundial [14]. Esta tecnología es utilizada en dos de los proyectos hidráulicos más grandes del mundo, Italpú (Brasil) y Las tres gargantas (China), proyectos con una potencia efectiva de 14.000 megavatios (MW) y 22.500 megavatios (MV) respectivamente [15], [16].

1.2.3 Ubicación de la Central San Francisco.

La turbina de este estudio es una turbina Francis de eje vertical situada en la central hidroeléctrica San Francisco, entre la cuenca del rio Pastaza en el sector de baños de agua santa en la provincia del Tungurahua. Esta es una de las dos turbinas instaladas en el complejo hidroeléctrico, ambas con una capacidad de generación de 115 [MW] [17].

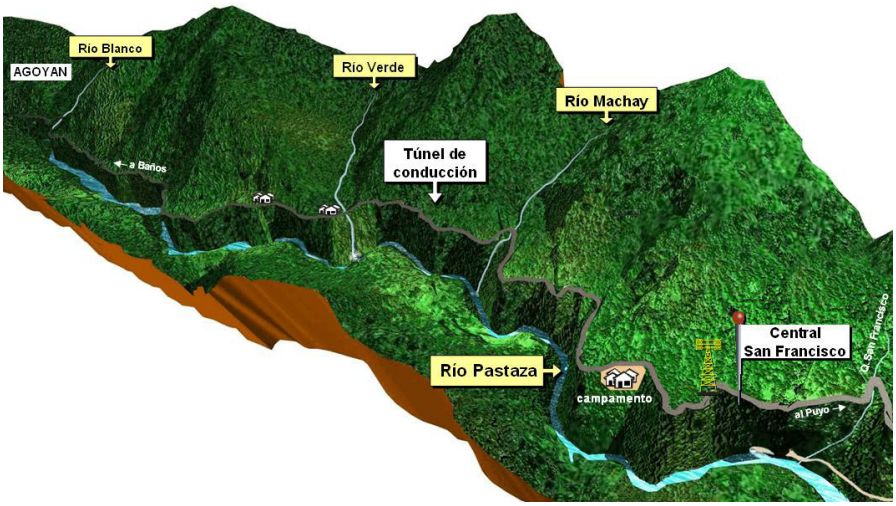

Figura 1.2 Ubicación de la central San Francisco en la cuenca del rio Pastaza. (Fuente:[17])

1.3 Material granular y sus características

El material granular es el conjunto de muchas partículas sólidas pequeñas, como granos o esferas que actúan como una fase sólida discreta de la materia [18]. Durán, en el 2000, define al flujo particular como una colección de partículas de tamaño superior a 1um [19]. El material granular está presente en la mayoría de las ramas de la ingeniería mecánica, civil, química farmacéutica, etc. La forma y tamaño de los granos que la constituyen pueden ser muy variadas.

Para estudiar de manera correcta al material granular es necesario conocer el comportamiento de las partículas individuales que lo componen dado que están puede tener diversas formas, tamaños y composiciones químicas agrupados de manera

aleatoria [20]. Con esta información se puede proponer ciertas aproximaciones que serán fundamentales para estudiar al fenómeno desde un enfoque más optimo.

1.3.1 Tamaño del grano.

En la naturaleza es difícil encontrar formas y tamaños regulares, al igual que en muchos campos de la industria, así que para facilitar el estudio se define a la partícula como una esfera perfecta y dada la variedad de tamaños que puede tener una partícula se trabajará con rangos de tamaño que describirán esta característica en un conjunto de granos.

A medida que la forma de la partícula se aleja de una esfera es más importante definir correctamente sus características. Para caracterizar correctamente una partícula de forma irregular, se usa una ecuación que permite acercarla a un estado regular y definirla como tal. La ecuación (1.1) muestra la longitud del "diámetro de volumen" (d_v) que tendría dicha partícula:

$$d_v = (\frac{6V_p^{\frac{1}{3}}}{\pi}) \tag{1.1}$$

Donde V_p es el volumen de la partícula.

Se hace esta aproximación para definir de manera más fácil a un conjunto de partículas que se encuentran en el mismo rango de tamaño, pues no siempre se va a encontrar granos que tengan forma similar en la naturaleza.

1.3.2 Forma del grano

Se ha enfatizado que la forma de la partícula afecta en la medición de los resultados, en este caso determinar la "no esfericidad" de un elemento en términos matemáticos es una tarea compleja. Sin embargo, para establecer el factor de forma se puede estimar el grado en que la partícula se aleja de ser esférica, para lo cual se relaciona el área de la superficie de la partícula con el área de la superficie de una esfera del mismo volumen [21]. Esta relación se ve detalla en la ecuación (1.2).

$$\Theta_s = \frac{\text{Área de la superficie de la esfera}}{\text{Área de la superficie de la partícula}} \; ; \text{del mismo volumen} \tag{1.2}$$

No obstante, encontrar con precisión el valor del área superficial de la partícula es un trabajo complejo por lo que se trabaja con mediciones indirectas, aproximaciones y promedios de una muestra de partículas. Es lo más cercano que tenemos para definir

la forma de las partículas dada su amplia variedad entre partículas de una misma muestra y de una muestra diferente.

1.3.3 Presencia de material granular en flujo hídricos

El estudio del movimiento de un fluido permite analizar los cambios de velocidad y dirección que sufren las partículas a lo largo de su recorrido por un medio fluido. La variación de las propiedades de las partículas respecto al tiempo se caracteriza como un flujo en estado transitorio. En la naturaleza es muy poco probable encontrar condiciones propicias para mantener un análisis de flujo laminar por lo que se considerara que el flujo mantiene un carácter turbulento, en el que las partículas poseen una característica de movimiento errático generalmente dibujando pequeños remolinos en su trayectoria, lo que ocasiona constantes colisiones entre ellas provocando cambios en la cantidad de movimiento y perdidas de energía significativas. El grado de turbulencia de un flujo se determina a partir del número de Reynolds (Re) detallado en la ecuación (1.3) representa la relación entre las fuerzas inerciales y las fuerzas viscosas que actúan en un fluido [22].

$$Re = \frac{\rho V L}{\mu} \tag{1.3}$$

Donde ρ representa la masa específica, V la velocidad, L es una dimensión característica y μ la viscosidad dinámica.

Este tipo de flujo también se considera como bifásico, ya que se compone de un fluido que interacciona con dos fases, en este caso sólido y líquido. La interacción del flujo hídrico y el material granular responde a los procesos hidrodinámicos y morfológicos de un río. El transporte de sedimentos en un río es la base de estudio para el desarrollo del presente proyecto enfocándose en la composición granular del material particular en el río Pastaza.

Las partículas arrastradas por el flujo de un río se conocen como sedimentos, que corresponden a todas las partículas de suelo y roca transportadas por una corriente de agua [23]. El flujo natural de un río transporta dos tipos de material: el de fondo que corresponde al material de formación del cauce y el de lavado, conformado por partículas finas que el agua transporta en suspensión. Generalmente, un conjunto de partículas no presenta tamaños uniformes, sino que responden a una distribución de tamaños [24].

La Tabla 1.1 Muestra la escala para la Clasificación establecida por la *American Geophysical Union*, esta información es necesaria para concluir un análisis granulométrico.

Tabla 1.1 Clasificación granulométrica según la *American Geophysical Union.*

Grupo	Clase	Tamaño [mm]
Piedras / Guijarros	*Muy grande*	2048 – 4096
	Grande	1024 – 2048
	Mediana	512 – 1024
	Pequeña	256 – 512
Cantos / Cascajo	*Grande*	128 – 256
	Pequeña	64 – 128
Grava	*Muy gruesa*	32 – 64
	Gruesa	16 – 32
	Mediana	8 – 16
	Fina	4 – 8
	Muy fina	2 – 4
Arena	*Muy gruesa*	1 – 2
	Gruesa	0,5 – 1
	Mediana	0,25 – 0,5
	Fina	0,125 – 0,25
	Muy fina	0,062 – 0,125
Limo	*Gruesa*	0,031 – 0,062
	Mediana	0,016 – 0,031
	Fina	0,008 – 0,016
	Muy fina	0,004 – 0,008
Arcilla	*Gruesa*	0,002 – 0,004
	Mediana	0,001 – 0,002
	Fina	0,0005 – 0,001
	Muy fina	0,00024 – 0,0005

(Fuente:[25])

Un análisis granulométrico permite conocer el tamaño de las partículas que conforman un sedimento a través de una representación porcentual de sus diámetros. Este procedimiento puede realizarse de forma manual o mecánica mediante la separación de partículas constitutivas del agregado según tamaños, de tal manera que se puedan conocer las cantidades en peso de cada tamaño que aporta al peso total[25].

Una vez distribuidos según su tamaño, los sedimentos pasan a tener nombre a través de la clasificación granulométrica, existen varias metodologías de clasificación, como la escala de Wentworth o la Clasificación establecida por la American Geophysical Union.

1.4 Dinámica de Fluidos Computacional (CFD)

La Dinámica de Fluidos Computacional (en inglés *Computational Fluid Dynamics*) hace referencia a una técnica numérica que busca comprender la geometría de los patrones de flujo, transferencia de calor y otros fenómenos físicos [26], a través de esquemas de discretización que definen resultados en cada punto de la función en el tiempo y espacio.

Esta herramienta combina dos disciplinas muy importantes: cálculo computacional y mecánica de fluidos; que resuelven ecuaciones de flujo fluido en la región de interés a partir de condiciones específicas en los contornos del dominio que se traducen en simulaciones numéricas que imitan el comportamiento de los sistemas de flujo [27].

El CFD se rige bajo las ecuaciones de la dinámica de fluidos también conocidas como Ecuaciones de Navier-Stokes [28]. Estas ecuaciones describen el movimiento tridimensional de las partículas en un fluido viscoso donde los gradientes de temperatura no representen valores extremos asegurando una buena aproximación a la realidad de un flujo viscoso en dos o tres dimensiones.

1.4.1 Ecuaciones que gobiernan el CFD

En primer lugar, se tiene la ecuación de continuidad la cual establece que la velocidad a la que la masa entra en un sistema es igual a la velocidad a la que la masa sale del sistema más la acumulación de masa dentro del sistema. La forma diferencial de la ecuación de continuidad se presenta en la siguiente ecuación (1.4).

$$\frac{\partial p}{\partial t} + \nabla \cdot \left(\rho \cdot \vec{V}\right) = 0 \tag{1.4}$$

En donde: ρ es la densidad del fluido.

t es el tiempo.

u es el campo vectorial de la velocidad del flujo.

La derivada del tiempo presente en la ecuación puede entenderse como la acumulación o pérdida de masa en el sistema, mientras que el término de divergencia representa la diferencia entre el flujo de entrada y el flujo de salida. Las ecuaciones de Navier-Stokes forman una ecuación de continuidad vectorial que describe la conservación del momento

lineal. Si el fluido es incompresible ósea que no hay cambio en el volumen a través del tiempo, la ecuación de continuidad de masa se simplifica a una ecuación de continuidad de volumen demostrado en la ecuación (1.5):

$$\nabla * \vec{V} = 0 \tag{1.5}$$

lo que significa que la divergencia del campo de velocidad es cero en todas partes. Físicamente, esto equivale a decir que la tasa de dilatación del volumen local es cero, por lo tanto, un flujo de agua a través de una tubería convergente se ajustará únicamente aumentando su velocidad ya que el agua es en gran parte incompresible [29].

Las ecuaciones de Navier Stokes son la representación matemática de la conservación de masa y momento para fluidos newtonianos y nacen de la aplicación de la segunda ley de newton al movimiento de un fluido, la forma matemática de la ecuación de Navier-Stokes es:

$$\frac{d\vec{V}}{dt} = -\nabla p + \rho\vec{g} + \mu\nabla^2 d\vec{V} \tag{1.6}$$

Esta ecuación refleja de relación entre el cambio de velocidad con respecto del tiempo, el gradiente de presión, fuerzas externas y la difusión del momento.

En definitiva, estas ecuaciones describen el movimiento del fluido proporcionando resultados bastante confiables, sin embargo, las soluciones numéricas involucran errores introducidos y están limitadas por la potencia de cálculo requerida [30]. A pesar de esta limitante el uso del método experimental en la resolución de problemas de flujo de fluidos no supera la eficiencia de los métodos computacionales.

1.4.2 Metodología del CFD

La metodología del CFD consta de tres etapas fundamentales: preproceso, resolución y postproceso. La etapa de preprocesamiento es la más demandante. Aquí se realizan suposiciones sobre el tipo de flujo en estudio, como por ejemplo sus propiedades de viscosidad, turbulencia y densidad; también se adaptan las condiciones del flujo a un modelo informático discretizado donde se realizan aproximaciones a las variables de estudio a través de nodos o números finitos para la creación de mallas y la aplicación de condiciones iniciales y de contorno.

Tabla 1.2 Metodología CFD.

Preproceso	- Definir los objetivos - Generar el modelo CAD - Crear la malla computacional - Configurar la simulación
Proceso	- Calcular la solución - Evaluar la convergencia de la solución
Postproceso	- Analizar los resultados - Presentación de gráficos, tablas, videos, etc. - Comparación de datos experimentales y/o analíticos

(Fuente: Propia)

1.4.3 El CFD y su aplicación en turbinas hidráulicas

Los métodos computacionales en el estudio de fluidos tienen un costo y un periodo de elaboración mucho menor que el experimental, ya que este último requiere de la fabricación de modelos e instalación física sujeta a pruebas de funcionamiento. El uso de CFD ha permitido una mejora significativa en la eficiencia de las turbinas hidráulicas gracias a las modificaciones necesarias impuestas en la etapa de diseño que garantizan la eficiencia optima antes de su fabricación [33].

La aplicación del CFD en la ingeniería cumple muchas funciones, entre ellas determinar la viabilidad de un producto desde su fase conceptual de diseño para garantizar un correcto funcionamiento y asegurar un tiempo de vida útil óptimo. Esta herramienta ha sido de gran utilidad en el diseño de turbinas Francis, ya que mediante una simulación es posible no solo predecir la eficiencia sino también el comportamiento del fluido dentro de la turbina determinando resultados más confiables [34].

El CFD predice el comportamiento del flujo optimizando su rendimiento y analizando varios parámetros importantes como presión y velocidad, determinando zonas críticas donde pueden ocurrir fallas. Más aun conociendo que la turbina hidráulica es el componente más crítico en las plantas hidroeléctricas, ya que de este depende el rendimiento general [35].

1.4.4 OpenFOAM

El presente estudio se focaliza en el uso de OpenFOAM como herramienta para la simulación numérica en una turbina tipo Francis en estado transitorio con malla fija.

Open Field Operation and Manipulation Utility Openfoam por sus siglas en inglés, hace referencia a un código de software libre gratuito programado en C++ que permite el fácil

acceso y modificación de códigos para todos los usuarios. Esta característica destaca el uso de OpenFOAM sobre los softwares comerciales, ya que estos últimos no presentan de manera clara la aplicación de características del fenómeno durante el desarrollo de su modelación impidiendo a los usuarios conocer qué condiciones iniciales se emplearon para llevar a cabo la simulación. Además, un software comercial impide el libre acceso a su código de fuente por lo que no se pueden realizar modificaciones o adaptaciones en sus parámetros de cálculo.

OpenFOAM se basa en esquemas de volúmenes finitos que se enfoca de manera especial en la simulación numérica de la Dinámica de Fluidos Computacional (CFD) permitiendo el desarrollo de aplicaciones categorizadas como *solvers* y *utilities* [8].

1.5 Mallado estructurado

Se conoce como “malla” a la subdivisión de un dominio computacional continuo en un número finito de elementos de control, los cuales pueden presentarse en una, dos o tres dimensiones [37]. Este método numérico se relacionó con los métodos computacionales desde el nacimiento de los primeros computadores electrónicos, posteriormente la evolución de las capacidades gráficas en los ordenadores impulsó el uso de mallas en el campo científico con aplicaciones de relevancia dentro de la imagen médica, la simulación numérica, la visualización científica y la animación 3D [38].

La generación de mallas es un requisito para la aplicación de varias técnicas numéricas como el método de elementos finitos (FEM) y el método de volúmenes finitos (FVM), ambas son técnicas numéricas de simulación 3D lo que conduce al uso de elementos poliédricos. El proceso de generación de mallas se puede clasificar en dos categorías según la topología de los elementos que llenan el dominio, estas dos categorías básicas se conocen como mallas estructuradas y no estructuradas. Los diferentes tipos de mallas tienen sus ventajas y desventajas en términos de precisión de la solución y complejidad del proceso de generación de mallas [39].

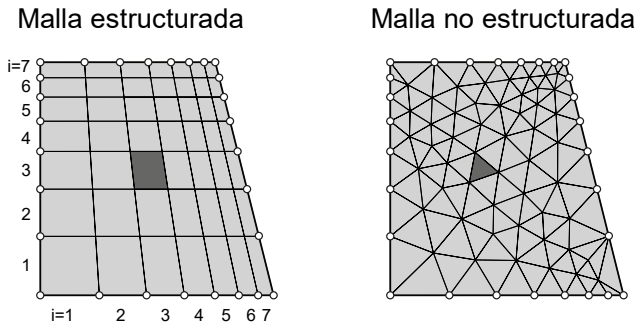

Figura 1.3 Comparación de mallas estructurada y no estructurada.
(Fuente Propia)

En la Figura 1.3 se puede observar dos mallas con igual discretización en su borde lo que quiere decir que sus bordes tienen la misma cantidad de nodos ubicadas a la misma distancia y posición, la malla estructurada cuenta con 49 elementos de cuatro lados ordenados en toda la superficie, cuentan con una calidad alta y se puede acceder a ellos fácilmente con valores de indexado o coordenadas, por otro lado la malla no estructurada cuenta con 79 elementos triangulares dispuestos de manera aleatoria con una calidad inferior.

El uso de mallas estructuradas ayuda en cálculos matemáticos complejos pues permiten discretizar un medio continuo de manera ordenada, facilitando y optimizando cálculos complicados [40]. Estos elementos tienen la característica de tener una buena calidad y un comportamiento numérico adecuado lo que permite su implementación en proyectos que buscan una precisión elevada en la predicción de sus resultados [7]. Dada esta estructura, permite capturar con mayor detalle y de manera eficiente ciertos fenómenos físicos.

1.5.1 Criterios de Calidad de Malla

En una solución numérica es necesario que los elementos de la malla posean una alta calidad a fin de asegurar una correcta discretización y buenas condiciones en el sistema lineal. La calidad de malla posee un impacto directo en el análisis computacional, definiendo la solución obtenida y el tiempo necesario para llegar a esta. Muchos autores han determinado que una buena discretización del dominio es esencial para conseguir una mejor aproximación al fenómeno físico de flujo [8].

La calidad de la solución de los métodos computacionales depende fuertemente de la resolución de la malla, es decir del número de elementos que representan el área de interés. Sin embargo, a continuación, se detallan los criterios más destacables para asegurar una buena calidad en el mallado, según [41].

- El cambio de tamaño de las celdas debe ser sutil. El crecimiento brusco puede afectar directamente a la calidad de la malla. La distancia entre nodos consecutivos debe incrementar de forma lenta, para asegurar una mayor precisión en los cálculos.
- El alargamiento de las celdas no debe ser excesivo. Este alargamiento típico debe desarrollarse en torno a la unidad.
- La deformación de las celdas debe ser mínima.

- La malla puede ser fina en ciertas zonas. Procurar un desarrollo de celdas finas en las capas límites, en zonas cercanas a discontinuidades y fuertes gradientes Por otra parte, puede mantenerse gruesa en zonas del dominio donde se sepa que la solución más detallada no va a verse afectada.
- Las celdas deben mantener la orientación de la dirección del flujo. Los flujos oblicuos a las caras entre celdas suelen ocasionar problemas de difusión elevada, esta es una de las razones por las cuales se prefieren mallas poliédricas en lugar de tetraédricas en un mallado tridimensional.

1.6 DEM

La Teoría del Modelo de Elemento Discreto fue introducida en 1956 por Alder y Wainwright con el objetivo de estudiar la dinámica molecular, fue hasta 1970 que Cundall y Stranck en su estudio "*A discrete numerical model for granular assemblies*", desarrollaron el Método del Elemento Discreto.

Este método simula la naturaleza de una fase sólida compuesta por partículas, las cuales interaccionan entre sí a través de sus puntos de contacto. Esta fase sólida es interpretada como un conjunto de elementos discretos que conforman un sistema complejo de partículas, estos elementos se desplazan independientemente unos de otros y son considerados elementos rígidos.

Este método se rige por las leyes de equilibrio dinámico de Newton, el movimiento de las partículas se origina en la integración de equilibrio de fuerzas. El método DEM es un modelamiento numérico de la realidad, con resultados comparables a un fenómeno físico de alta complejidad numérica con una alta precisión. Para concluir la teoría, se supone que las partículas no interaccionan con efectos externos como factores de presión o temperatura ajenos ya que se mantienen libres de efectos termodinámicos.

1.6.1 Suposiciones del DEM.

Es importante conocer las asunciones de este modelo para facilitar el proceso de cálculo, por lo que tenemos las siguientes consideraciones:

1. Todas las partículas son cuerpos rígidos por lo que la geometría no va a cambiar ante la interacción con las paredes y otras partículas.
2. El contacto entre partículas se da solamente en un punto de contacto.

3. Las partículas poseen un contacto blando que permite exista una superposición de las partículas en sus puntos de contacto este valor de superposición se determina con la ley de desplazamiento de fuerza.
4. El solapamiento entre partículas es pequeño en comparación con el tamaño de las partículas y mucho más pequeño que el valor de la traslación.
5. La interacción ocurre únicamente en el punto de contacto entre partículas.
6. El *timestep* o paso de tiempo (Δt) será lo suficientemente pequeño para que cada partícula logre interaccionar solamente con las partículas en contacto y no afectará a otras.
7. Los valores de aceleración y velocidad son constantes en cada *timestep* o paso de tiempo (Δt) especifico.
8. La traslación y rotación de las partículas son descritas únicamente por la segunda ley de newton.
9. El *timestep* o paso de tiempo (Δt) es tan pequeño que las perturbaciones no se propagan a través de una partícula más allá de sus vecinos inmediatos.
10. En cada instante las fuerzas que actúan sobre una partícula se determinan exclusivamente por su interacción con sus partículas en contacto.

1.6.2 Ecuaciones que gobiernan el Método del Elemento Discreto.

El DEM calcula la trayectoria de las partículas individualmente tomando en cuenta la influencia de las partículas cercanas, otras fuerzas presentes en el medio de simulación o paredes que limiten el movimiento de los granos. Esta trayectoria depende de un movimiento de rotación y traslación que son descritos por la segunda ley de Newton:

$$m_i \frac{dv_i}{dt} = \sum_j F_{ij}^c + F_i^g + F_i^e + F_i^\delta \tag{1.7}$$

$$I_i = \frac{d\omega_i}{dt} = \sum_j M_{ij} \tag{1.8}$$

Donde: v_i es la traslación y ω_i es la velocidad angular de la partícula i.

F_{ij}^c es la fuerza y M_{ij} es el torque que actúan sobre la partícula i debido al contacto j con otros elementos del medio como partículas o paredes.

F_g es la fuerza de la gravedad.

F_i^e es la fuerza total externa que se ejerce sobre la partícula i proveniente de otras fuentes (Como la interacción de fluidos, por ejemplo).

F_i^{δ} es una fuerza local de amortiguamiento.

En DEM, los mecanismos de disipación de energía como la fricción, se modelan explícitamente [44].

1.6.3 Contacto entre partículas

La modelación de la interacción entre partículas puede ser descrito como “Esferas suaves” o “esferas duras” [45].

a. Esferas Suaves

Este modelo de contacto indica que las partículas pueden interactuar a través de las fuerzas de corto y largo alcance, y las trayectorias se determinan con la integración de las ecuaciones de movimiento de Newton. Este modelo permite a las partículas traslaparse y con ello determinar las fuerzas de contacto en base a la distancia del traslape y la velocidad de contacto relativa de las partículas. La cantidad de traslape y la fuerza de contacto resultante son llamados modelos elásticos, en donde destacan los modelos Hertz y el de contacto lineal. Esto presenta una ventaja ya que estudia la interacción de partículas con un enfoque más natural incluyendo en el cálculo más factores que alteran al fenómeno, una desventaja es que ocupa un mayor gasto computacional.

b. Esferas duras

Este sistema determina la trayectoria de las partículas mediante la conservación de momento de colisiones binarias, lo que supone un tiempo instantáneo para la interacción entre dos partículas por tal motivo colisiones múltiples en un mismo instante no pueden ser consideradas. En este sistema las colisiones son procesadas de una en una en el orden en que se vayan produciendo. Este sistema es útil para sistemas con un escenario poco denso de partículas, en donde las colisiones con partículas vecinas son muy escasas. Los modelos de esferas duras son considerablemente más veloces y requieren menos poder computacional.

1.6.4 Modelos de Contacto.

En DEM, las partículas son representadas como cuerpos rígidos, por tal la deformación producto de la colisión con paredes o con otras partículas es aproximada con los modelos de contacto. El modelo de contacto se encarga principalmente de definir la

fuerza y el torque que generan dichas colisiones. Las partículas pueden estar en contacto múltiple con los elementos que la rodean, cada uno de estos contactos se calcula por separado. En un marco ajeno a DEM, el contacto entre dos esferas al iniciar la interacción es únicamente un punto luego la deformación amplia la superficie de contacto. DEM equipara esta deformación con una superposición entre las superficies como se muestra a continuación:

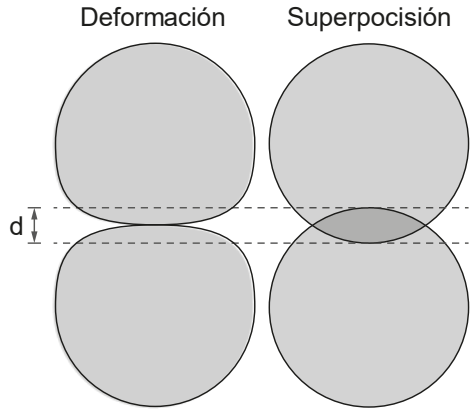

Figura 1.4 Deformación de una partícula y su equivalencia en superposición.
(Fuente propia)

La superposición entre partículas d o δ se define de para el enfoque de partículas suaves con un modelo de contacto. Con esto, la precisión de la simulación DEM depende casi en su totalidad de la mecánica de contacto elegida usualmente estas se fundamentan en la dirección de acción de la fuerza de contacto, sea esta normal o tangencial [46]. Se construye el modelo para el modelo de contacto modelándolo con un resorte, amortiguador y un control deslizante como se muestra en la Figura 1.5.

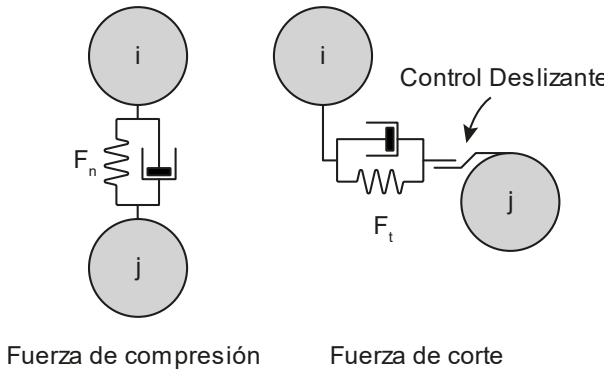

Figura 1.5 Modelo de contacto de fuerzas resorte amortiguador.
(Fuente propia)

Para resolver la mecánica de contacto se utilizará la teoría del contacto hertziana, esta es una teoría clásica de la mecánica del contacto y es una herramienta muy útil para ingenieros e investigadores. Aunque la derivación de la teoría es relativamente difícil, la solución final es un conjunto de ecuaciones analíticas simples que relacionan las propiedades del sistema con la tensión desarrollada. Este modelo de matemático ayuda a resolver la mecánica de contacto de material particular con propiedades similares, además presenta una solución eficiente al problema de la deformación que promueve un ahorro de recursos computacionales y tiempo de simulación.

La teoría del contacto de Hertz se deriva de la solución analítica de las ecuaciones de la teoría de la elasticidad bajo la solución propuesta por Timoshenko y Godier en 1970 con las siguientes aproximaciones [47]:

1. Las superficies son espacios infinitamente grandes.
2. El perfil de presión es parabólico o sea asume que la forma de los cuerpos en contacto puede aproximarse con formas parabólicas, por ejemplo, esferas, elipses o cilindros.
3. Se aplican todos los supuestos de la teoría clásica de la elasticidad (pequeña deformación, material homogéneo).

La deformación para este en este modelo de contacto viene dada por la siguiente relación:

$$u_z(x,y) = \frac{2\pi}{E'} \iint \frac{p(x',y')}{\sqrt{(x-x')^2 + (y-y')^2}} dx'dy' \tag{1.9}$$

Aquí: u_z es la deflexión elástica (superposición en DEM).

$\frac{1}{E'}$ es el módulo de Young reducido, y es equivalente a:

$$\frac{1}{E'} = \frac{(1-v_1)^2}{E_1 + \frac{1-v_2^2}{E2}} \tag{1.10}$$

v_1 y v_2 son los coeficientes de poisson de las partículas.

E_1 y E_2 son los módulos de Young de las partículas.

$p(x,y)$ es la presión de contacto.

Si el perfil de presión es arbitrario, esta ecuación no conduce a la solución analítica. Sin embargo, la solución de Hertz se obtiene bajo el supuesto de una distribución de presión parabólica, que es una muy buena aproximación para cuerpos esféricos, elípticos o cilíndricos en contacto. Por tal motivo, para el estudio de este modelo de contacto de en partículas esféricas se usa la siguiente aproximación:

$$p(x,y) = p_0\left(1 - \frac{r^2}{a^2}\right)^{0.5} \tag{1.11}$$

En donde: r es un punto arbitrario en la superficie.

a es radio de contacto de Herzt.

p_0 es la presión de Hertz.

Si reemplazamos esta expresión en la ecuación de deflexión tenemos la siguiente igualdad para el modelo de contacto de Hertz:

$$u_z = \frac{\pi p_0}{4E'a}(2 \cdot a^2 - r^2); r \leq a \tag{1.12}$$

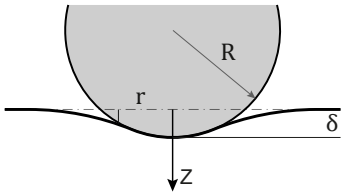

Figura 1.6 Esfera en contacto con un plano.
(Fuente Propia)

El caso que se adapta a la simulación de partículas es el contacto en un solo punto o contacto esférico.

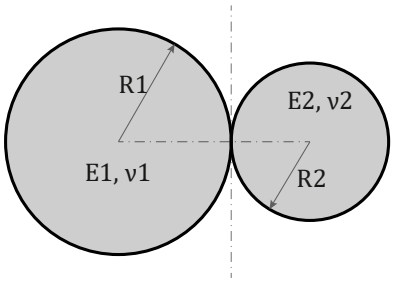

Figura 1.7 Contacto en un solo punto entre partículas esféricas de material y tamaño diferente.
(Fuente: Propia)

Para este caso de estudio el radio de Hertz (a) se calcula con siguiente ecuación:

$$a = \sqrt[3]{\frac{3FR}{E'}} \tag{1.13}$$

Mientras que R esta dado por:

$$\frac{1}{R} = \frac{2}{R_1} + \frac{2}{R_2} \tag{1.14}$$

δ es igual a:

$$\delta = \frac{\pi \cdot a \cdot p_0}{2E'} \tag{1.15}$$

p_0 está definido como:

$$p_0 = \frac{\frac{2}{\pi} E' \cdot \delta}{R^{0.5}} \tag{1.16}$$

Y F es lo mismo que:

$$F = \frac{4}{3} E' \cdot R^{0.5} \cdot \delta^{0.5} \tag{1.17}$$

1.6.5 LIGGGTHS ®

Para desarrollo de esta investigación se utiliza como programa DEM a LIGGGHTS ® en su versión Publica. LIGGGHTS (R) -PUBLIC es un software de simulación de partículas con método de elementos discretos de código abierto.

LIGGGTHS por sus siglas en ingles LAMMPS *Improved for General Granular and Granular Heat transfer Simulations* en español LAMMPS Optimizado para Simulaciones Granulares en General y Granulares con transferencia de calor. LAMMPS es un simulador que basa su cálculo en la dinámica molecular clásica ofreciendo funcionalidades muy simples y básicas para simulaciones tipo DEM. Christoph Kloss desarrollador principal de LIGGGTHS ® desarrollo este código con el objetivo de mejorar esas capacidades y aplicarlas a gran escala en un ámbito industrial. Actualmente se usa este software dentro del campo de los productos químicos, bienes de consumo, farmacéutica, ingeniería agrícola, producción de alimentos, producción de acero, minería, producción de plástico, etc., considerada una herramienta fundamental para mejorar los procesos de producción.

Dado el desarrollo privado del código a través de LIGGGTHS-PREMIUM, es capaz de funcionar de manera eficiente en equipos de escritorio o portátiles con un solo procesador, aunque está diseñado para equipos con múltiples procesadores dado su desarrollo sobre MPI *(Message Passing Interface)* que le permite para un correcto y optimizado proceso de simulación en paralelo.

LIGGGTHS permite simular de manera muy precisa simulaciones con unas cuantas partículas, hasta sistemas con miles de millones de partículas, además está diseñado para ser fácil de modificar o ampliar con nuevas capacidades, como nuevos campos de fuerza, tipos de átomos o condiciones de contorno.

1.7 CFD-DEM

El modelamiento de un flujo con partículas sólidas es un sistema muy complejo de resolver. Las partículas generalmente son mucho más pequeñas que las escalas típicas de fluido, por lo cual su estudio en CFD presenta un inconveniente. El acoplamiento de CFD-DEM es una alternativa propicia para el modelamiento de sistemas fluidos granulares. El DEM se aplica para modelar el sistema de partículas granulares mientras que el CFD se encarga de resolver el flujo fluido a través de las ecuaciones de Navier-Stokes.

La combinación de estos dos permite aplicar el enfoque euleriano al fluido y el lagrangiano a las partículas. El enfoque lagrangiano toma a las partículas como cuerpos esféricos individuales que son rastreados a lo largo del dominio estudiando las fuerzas que actúan sobre ellas, mientras que la euleriana toma al flujo como un medio continuo donde se analizan las variables de campo y las funciones de espacio y tiempo en un volumen de control. Aunque el flujo de las partículas puede comportarse como un fluido continuo a veces, su comportamiento discontinuo no se puede simular con el modelado tradicional basado en el continuo [48].

El enfoque CFD-DEM ha sido ampliamente utilizado en los últimos años para estudiar sistemas de interacción fluido-partículas debido a sus múltiples ventajas computacionales sobre la simulación numérica directa [49], además en este enfoque, no existe una limitación específica sobre la concentración y distribución del tamaño de las partículas, ya que estas pueden ser usadas simultáneamente en el solucionador DEM sin generar aumento de costos en el CFD.

El método acoplado de CFD-DEM es gobernado por la ecuación promediada local de Navier-Stokes, la ecuación de continuidad en fase fluida empleando CFD que engloba

las ecuaciones de la conservación de masa y la cantidad de movimiento, mientras que la segunda ley de Newton calcula el desplazamiento para la fase sólida granular utilizando DEM [49].

Tabla 1.3 Descripción general de las ecuaciones y herramientas empleadas en el método CFD-DEM.

Fase	Ecuaciones	Herramienta
Líquida	*Conservación de masa* *Conservación de la cantidad de movimiento*	CFD
Sólida	*Principio fundamental de la dinámica de Newton*	DEM

(Fuente: Propia)

En el modelo acoplado CFD-DEM existen varios factores prácticos a considerar durante el diseño de modelo. Inicialmente se esperaba emplear este modelo para simular sistemas de partículas de alta fracción volumétrica, que concentra una cantidad excesiva de partículas en el dominio generando costos computacionales exagerados [50].

En las ecuaciones de la fase fluida se han incluido los efectos de la transferencia de impulso. Existen dos formulaciones distintas enfocadas en la modificación de la ley de conservación de la cantidad de movimiento, donde, la caída de presión compartida entre fase líquida y sólida es conocida como Modelo A; y la caída de presión aplicada únicamente a fase fluida como Modelo B [44].

1.7.1 Procedimiento de acoplamiento CFD-DEM

Una vez validado un modelo monofásico tanto para partículas como para fluido, se combina el método de CFD con el DEM a través de dos paquetes de software de código abierto: OpenFOAM y LIGGGHTS respectivamente. Finalmente, estos códigos se acoplan en el enfoque conjunto de CFD-DEM.

Kloss et al [51] describen la rutina de acoplamiento entre el método CFD y DEM tomando a consideración que el tamaño de las partículas es más pequeño que la cuadrícula computacional, es decir, que no alcanzan a llenar una celda completa. El procedimiento se detalla a continuación de manera resumida:

1. Se calculan las posiciones y las velocidades de las partículas a través del DEM y los valores obtenidos se transfieren al solucionador de CFD.

2. Se calcula un primer campo de flujo de fluido.
3. Se determina la celda de malla correspondiente a cada partícula.
4. Se determina la fracción de volumen y la velocidad media de las partículas para cada celda.
5. Se calculan las fuerzas que actúan en cada partícula sobre la base de la fracción de volumen.
6. Se calcula la velocidad del fluido tomando en cuenta la fracción de volumen local y el intercambio de impulso a través del solucionador CFD.
7. Se obtiene los términos de intercambio de cantidad de movimiento entre partículas y fluido.
8. Se calcula las fuerzas del fluido que actúan sobre las partículas y se envían al solucionador de DEM.
9. El procedimiento se repite desde el paso 1 para el siguiente grupo de partículas.

La interacción entre CFD y DEM se realiza a través de la evaluación de campos de intercambio en intervalos de tiempo definidos, donde los códigos trabajan de manera secuencial. Tanto el código CFD como el DEM emplean una Interfaz de Paso de Mensajes (MPI) para realizar sus cálculos en paralelo y realizar el intercambio de datos. En el proceso de acoplamiento, el solucionador DEM y CFD se encuentran en constante comunicación para el intercambio de datos. Generalmente los pasos de tiempo del CFD deben establecerse en una orden de magnitud menor a los pasos de tiempo del DEM, ya que la dinámica de colisión es alta, así como los requisitos de superposición de partículas en el modelado de colisión [51].

La metodología de acople entre CFD y DEM se presenta en la Figura 1.8.

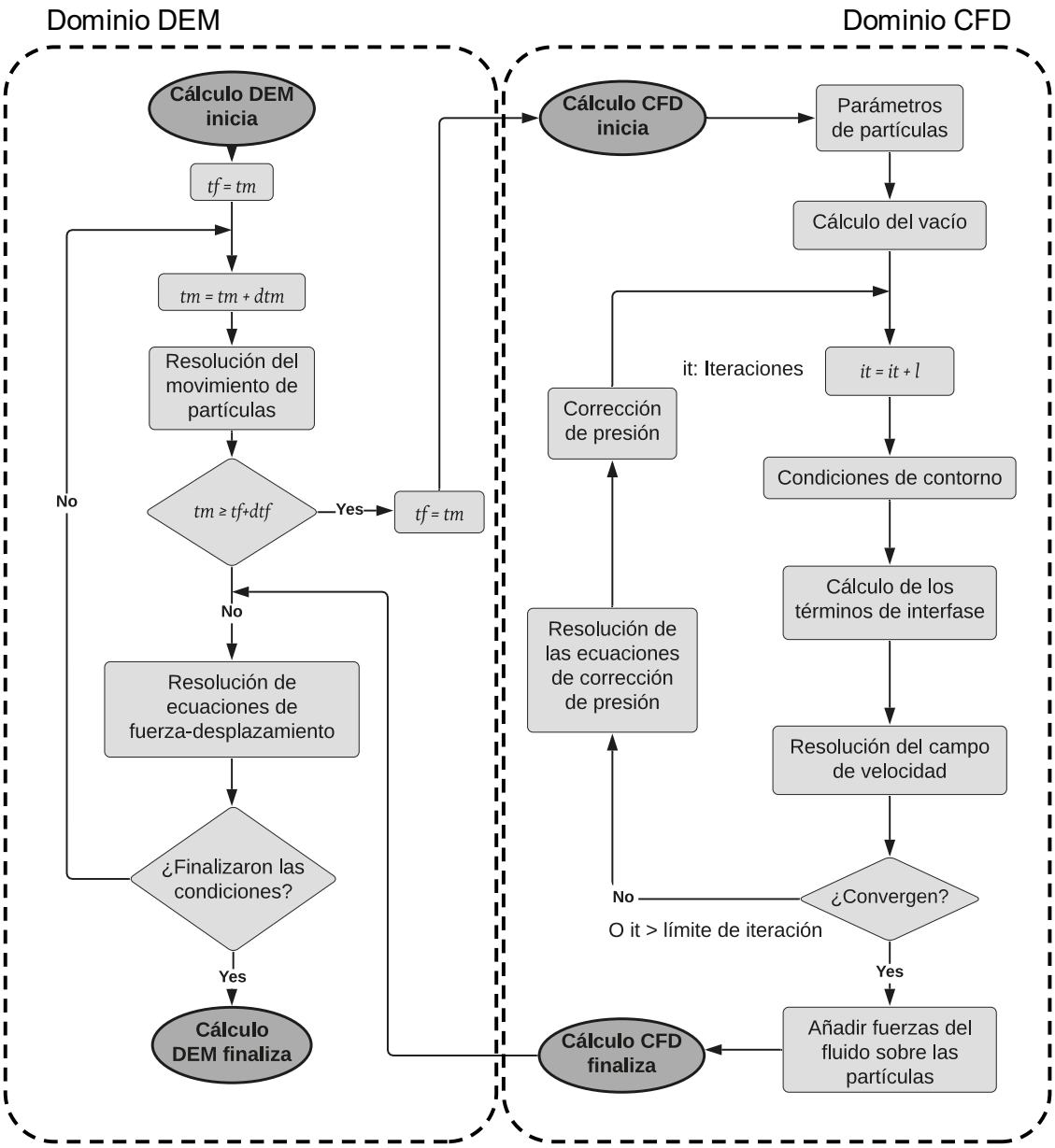

Figura 1.8 Esquema gráfico del cálculo conjunto del método de acoplamiento CFD-DEM.

(Fuente: [52])

2 METODOLOGIA

Este capítulo detalla todos los pasos que se siguieron para Estudiar numéricamente el flujo particulado dentro de una turbina tipo Francis usando OpenFOAM.

En la Figura 2.1 se destacan cuatro etapas importantes; Comenzando por el preproceso, prepara los dominios computacionales para las simulaciones numéricas CFD y CFD-DEM partiendo de las mallas estructuradas desarrolladas anteriormente por Velasco [7]. Estas mallas son tratadas para tener un número menor de elementos previo a su exportación del entorno de Ansys a OpenFOAM, con el objetivo de optimizar los tiempos de simulación y la carga computacional sin perder calidad en la malla. De esta etapa se obtiene también las superficies que se utilizarán en las simulaciones DEM y CFD-DEM como paredes o límites físicos en el cálculo.

Una vez tratados los dominios computacionales se los usará en las simulaciones numéricas junto a los datos de operación de la hidroeléctrica de San Francisco, valores que se los considerará como condiciones de borde y condiciones iniciales para la simulación con un algoritmo de solución (SIMPLE) con el objetivo de obtener resultados de un estado de estabilidad de flujo y la convergencia de resultados para así utilizarlos como nuevas condiciones iniciales en una simulación en estado transitorio con el algoritmo de solución (PISO) para estudiar así todo el domino y controlar la calidad de los datos obtenidos y aprovechar el poder computacional disponible óptimamente.

Tras concluir esta etapa de análisis del flujo en el interior de la turbina, pasamos al estudio de este fluido con la inserción de partículas, las cuales son calculadas en base a los datos granulométricos del cauce del rio caracterizados en base a la clasificación granulométrica de la *American Geophysical Union* y generadas mediante DEM. Estos datos de posición, composición y forma del material granular junto con los resultados de las simulaciones previas son parte de las condiciones iniciales de la simulación de flujo particulado. Toda esta información se utilizará para ingresarla al algoritmo de solución de CFD-DEM y simular numéricamente la naturaleza de las partículas y su relación con el flujo al interior de la turbina Francis.

Finalmente, para verificar la confiabilidad de los resultados obtenidos se realiza una validación cualitativa de los resultados visualizándolos y tratándolos con la ayuda de paraview, presentando las zonas de mayor contacto con las partículas, tanto en la caracola, los alabes y el tubo de descarga.

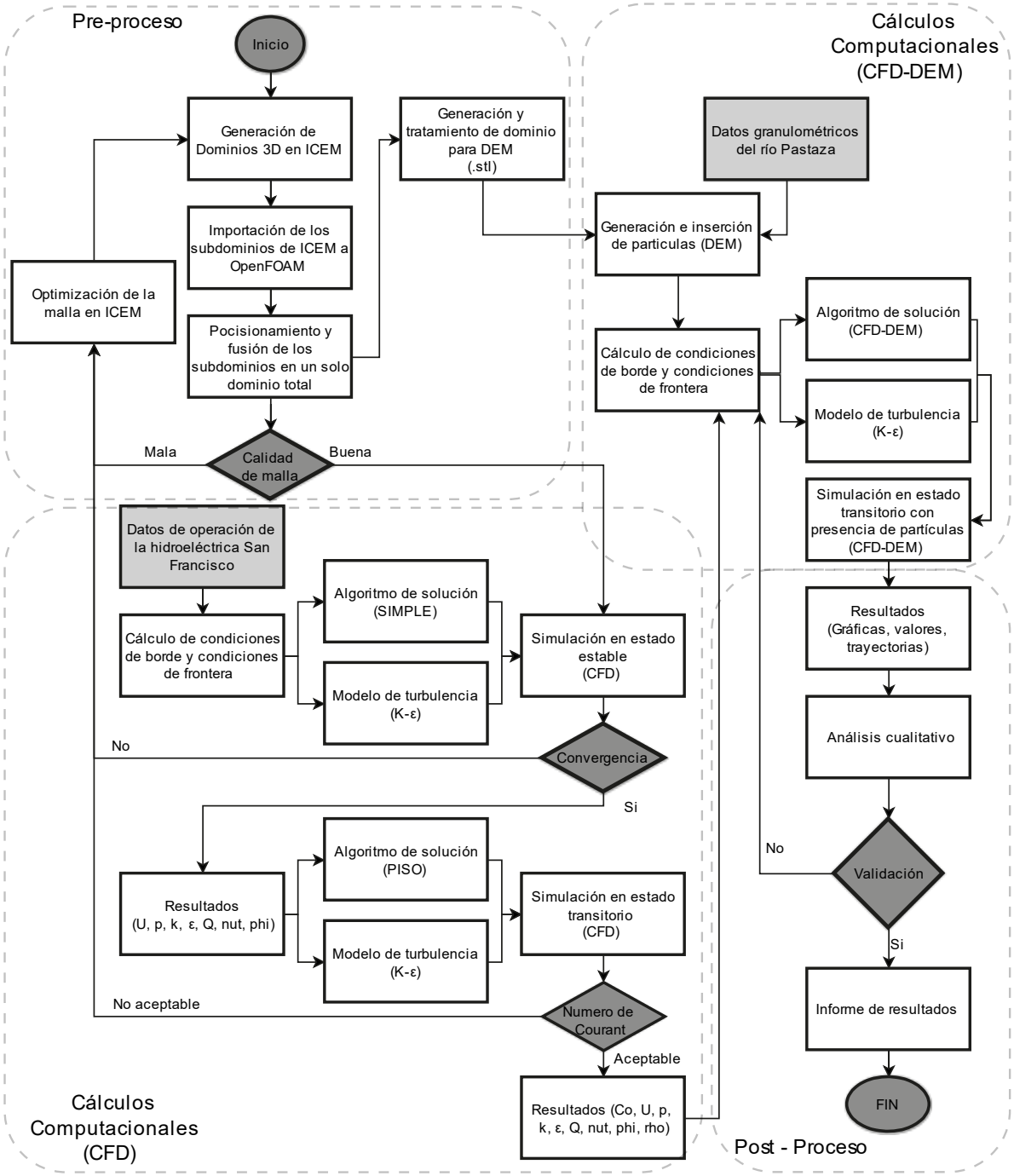

Figura 2.1 Metodología del presente caso de estudio.

2.1 Geometría

Modelar una turbina Francis desde los planos provistos por CELEC es un trabajo sumamente complicado que fue realizado a detalle por Mora en 2018 [5] y optimizado para facilitar su uso en simulación numérica por Guascal y Quishpe en 2019 [53]. Para lograrlo se realizó un proceso de ingeniería inversa con ayuda de un escáner 3D y la

optimización se realizó simplificando puntos en las superficies y retirando puntos excedentes.

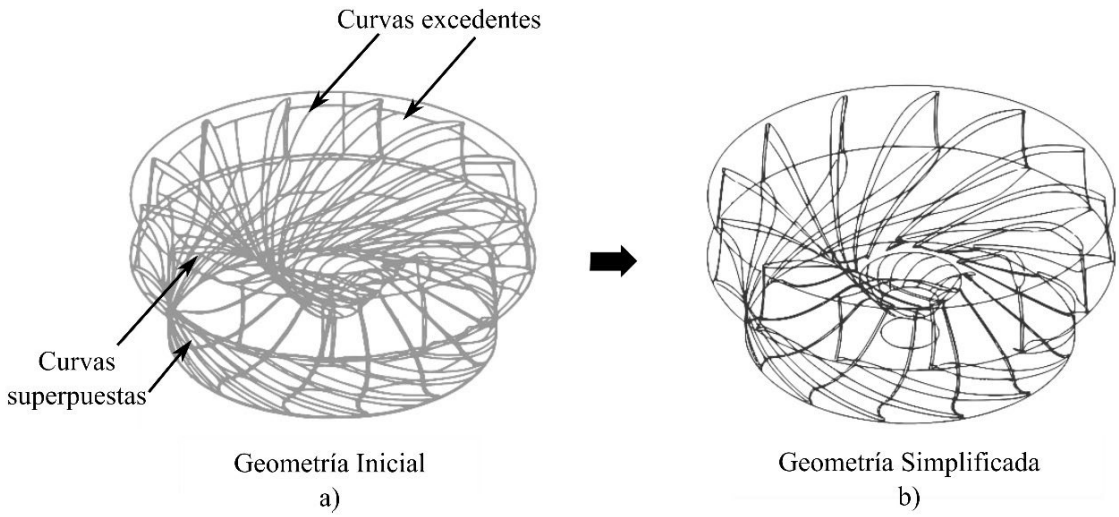

Figura 2.2 Modelo geométrico del rodete a) Mora y b) Guascal.
(Fuente: [53])

En la Figura 2.2 se puede ver la optimización lograda por Guascal, este resultado fue tomado por Velasco en su estudio [7] para modelar el dominio geométrico en ICEM y lograr un mallado estructurado.

Velasco mejoró muchas partes sensibles en la geometría lograda anteriormente para obtener mejores resultados, suavizando superficies y acercándolo más a la geometría real de la turbina como se presenta en la Figura 2.3.

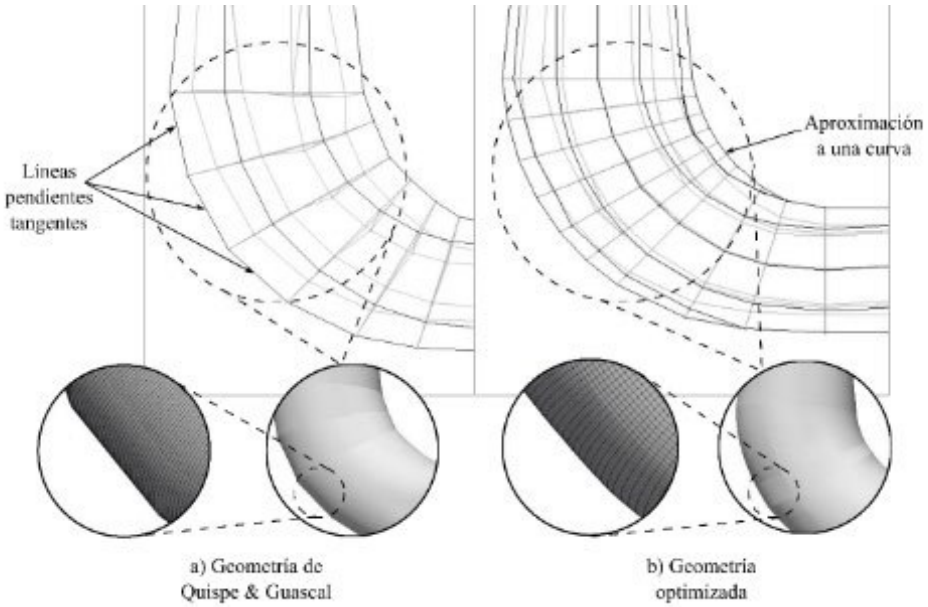

Figura 2.3 Comparativa entre geometría del tubo de descarga.
(Fuente:[7])

2.2 Optimización de Mallado estructurado

Las mallas obtenidas por Velasco suponen una propuesta excelente para el análisis de variables muy específicas en el comportamiento del flujo y su interacción con el medio, lo cual implica un número muy elevado de elementos en el dominio computacional, dada la existencia de refinamientos especiales en las superficies y concentraciones numerosas de elementos en zonas de mayor interés. El análisis propuesto anteriormente, hace uso de un conjunto de cuatro mallas que comparten información a través de sus superficies externas contiguas, obteniendo un dominio total que asciende a los 3.75 millones de elementos para un mallado grueso y 6.3 millones para un mallado fino [7]. Esta cantidad es manejable en el dominio de CFD sin problema, pero para introducir al método de elemento discreto (DEM) en el cálculo, esta malla tan numerosa aumenta el tiempo de simulación y hace un uso desmedido de poder computacional por lo cual es necesario reducir el número de elementos del dominio.

La malla fue modificada cuidando que los detalles característicos de la turbina no se vean afectados, por lo que se direccionó la reducción de elementos en zonas con mayor densidad de elementos. En la Figura 2.3 se ve la reducción de elementos lograda en el *draft tube*, en donde se cuida la aproximación a la curva para un mejor análisis del flujo de descarga [7].

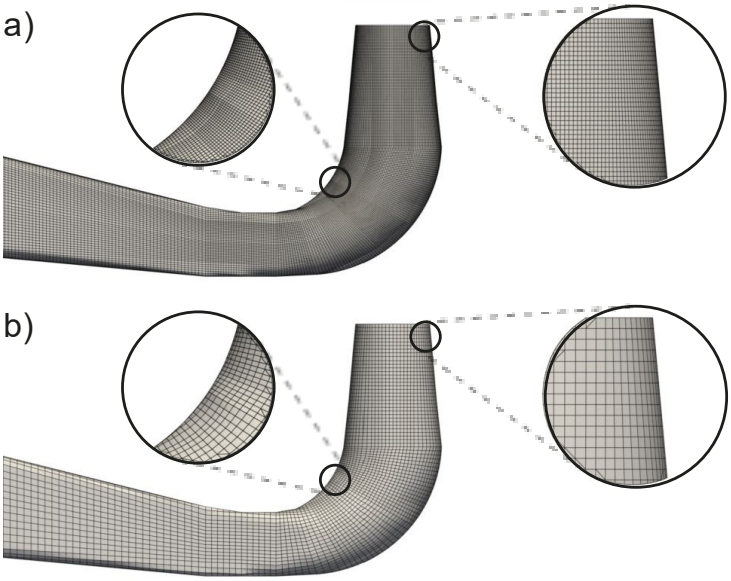

Figura 2.4 a) Refinamiento logrado por Velasco. b) Malla modificada.
(Fuente: Propia)

Este proceso de mallado y optimización de las partes de la turbina se realizó en ICEM, un programa de paga que presenta una amplia gama de herramientas que facilitan la

manipulación y el manejo de la información de los nodos de la malla con ayuda de una interfaz gráfica amigable con el usuario, a continuación, se presenta los dominios obtenidos producto de la optimización en ICEM, en el lado izquierdo se ven los dominios de partida y en el derecho las mallas optimizadas.

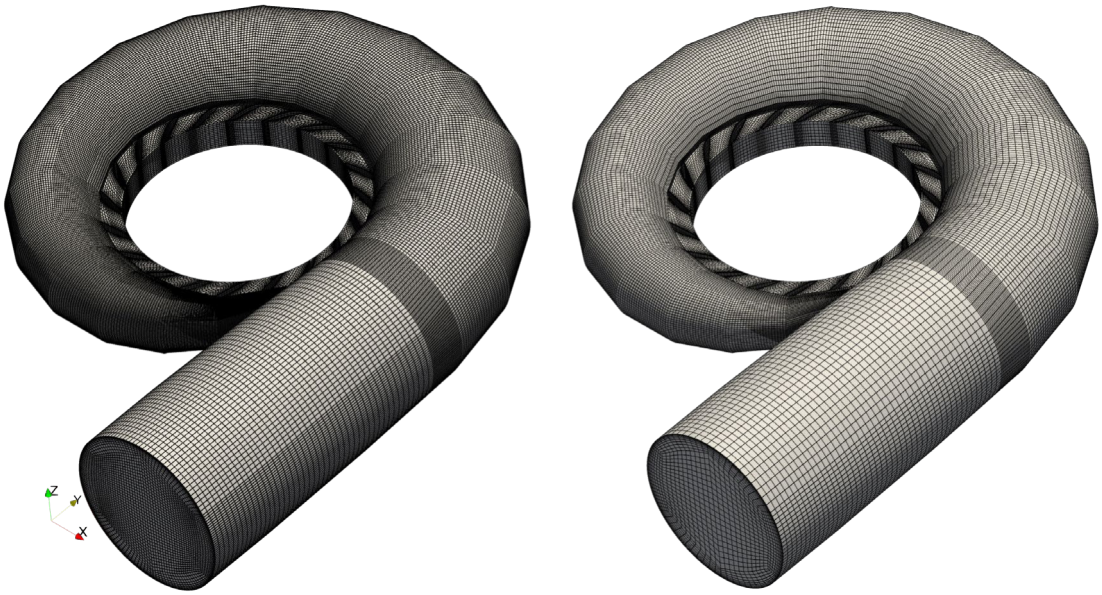

Figura 2.5 Comparativa del Spiral case.
Fuente: (Propia)

Figura 2.6 Comparativa del Guide vane.
Fuente: (Propia)

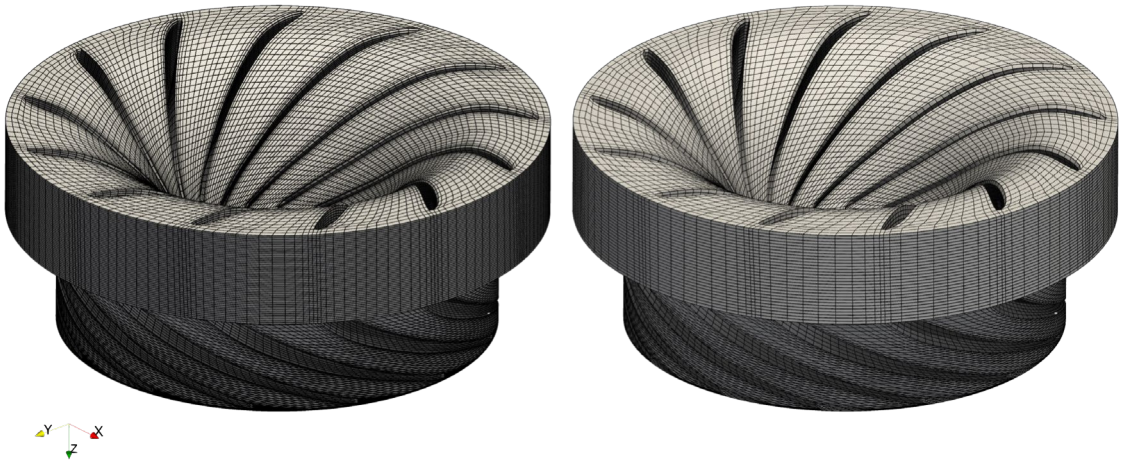

Figura 2.7 Comparativa del Runner.
Fuente: (Propia)

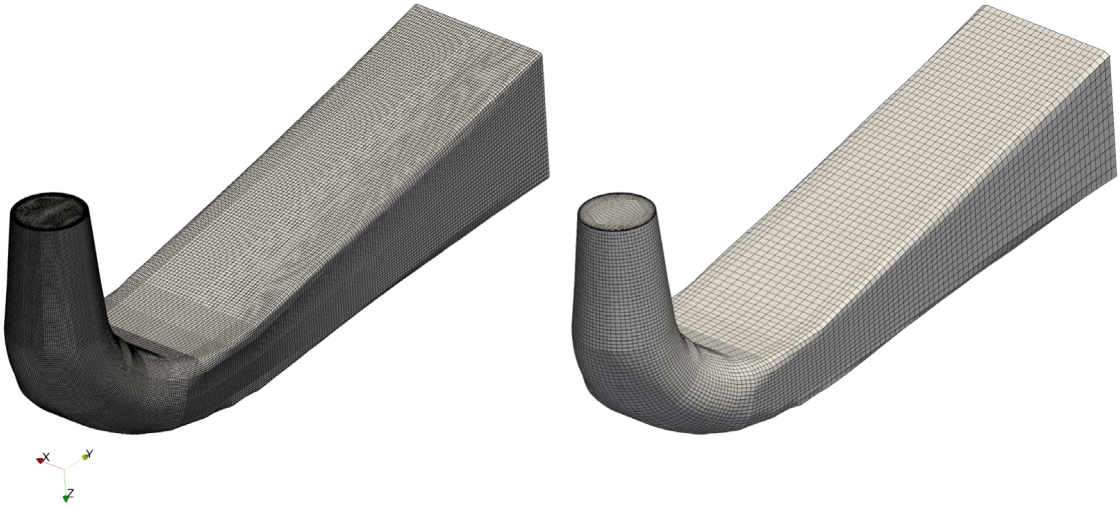

Figura 2.8 Comparativa del Draft Tube.
Fuente: (Propia)

En la Tabla 2.1 Se muestra la cantidad de elementos que había previo a la modificación y reducción de elementos de la malla generada por Velasco. Cabe destacar que la malla que fue utilizada como punto de partida es el resultado de una optimización de las geometrías logradas por Guascal y Quishpe en el 2019. Habiendo conseguido una transición más suave en el mallado y ajustando las superficies a la geometría real de las turbinas.

Tabla 2.1 Comparación entre cantidad de elementos de la malla de Velasco y la desarrollada. para este caso

Malla	Número de elementos	
	Original	Modificada
Spiral Case	1.537.298	588.180
Guide	1.551.080	700.400
Runner	902.045	188.906
Draft Tube	2.299.128	191.808
Total	6.289.551	1.669.294

2.3 Importación y fusión de la malla

Una vez terminado el proceso de reducción de elementos en ICEM, se exportará la información de cada parte de la turbina hacia el programa de simulación mediante una malla (.msh). Para importar esta información es necesario crear un nuevo caso para cada malla y con ayuda del aplicativo *fluent3DMeshToFoam* generar los archivos de la carpeta polyMesh que será interpretada por OpenFOAM. La estructura de archivos necesaria para la importación de los dominios computacionales se muestra en la Figura 2.9.

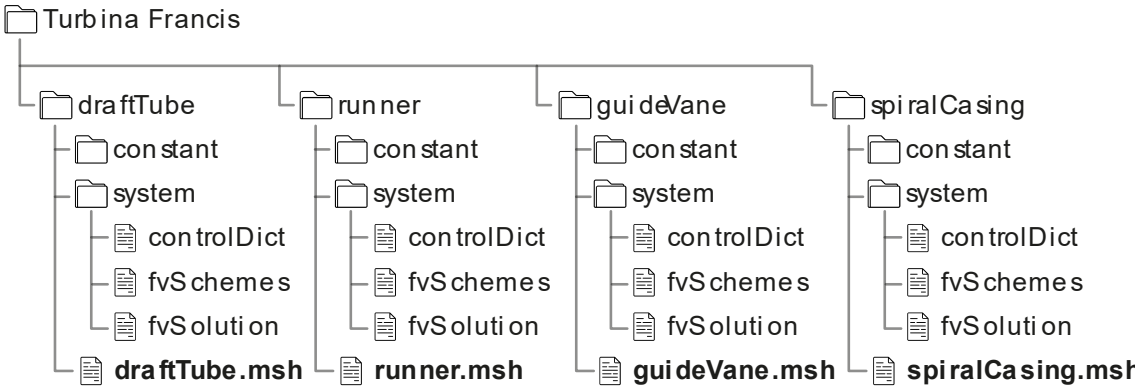

Figura 2.9 Estructura de archivos necesarios para la importación de la malla a OpenFOAM. (Fuente: Propia)

Se vio la necesidad de estructurar los archivos de este modo para poder utilizar *scripts* de comandos que se encargaran de posicionar a las partes de la turbina de manera correcta y automática, permitiendo a las superficies limítrofes estar lo más cerca posible. Como se puede ver en el flujograma de este estudio (Figura 2.1), este caso de estudio depende de la calidad de la malla obtenida en ICEM, por lo que cada vez que se obtienen resultados no deseados toca empezar desde el inicio, es por eso que se ha decidido manipular las archivos de este modo ahorrando tiempo y permitiendo concatenar las diferentes etapas simplemente ejecutando las secuencias de comandos

desde un archivo ejecutable y no ejecutando cada comando una y otra vez. El script utilizado para posicionar correctamente la malla se lo puede revisar en el ANEXO I.

Una vez resuelto el tema de la posición de los subdominios es necesario fusionar las mallas en un solo dominio sin interfaces de transferencia de información, pues esta cualidad dificulta el cálculo de las trayectorias del material particular, por lo cual se fundirá los 4 dominios computacionales en uno solo. Se realizará esta modificación con el fin de no entrar al dominio de una malla móvil pues esta característica incrementa el tiempo de cálculo y dificulta la convergencia de los resultados, además que la característica se encuentra aún en desarrollo por lo cual hay que aportar al proyecto CFD-DEM® [51] para poder acceder a las versiones beta de la herramienta, esta situación limita el estudio al uso de una malla fija.

Para conseguir esto se tiene que fusionar los casos de malla OpenFOAM generados anteriormente en un solo caso, para lo cual vamos a utilizar el aplicativo *mergeMeshes* que nos permite juntar en un mismo caso a dos o más dominios computacionales, manteniendo intacta su estructura, e identificando cada subdominio como una región diferente. Sin embargo, en este punto aún tenemos una malla con 4 regiones por lo que se va a utilizar el aplicativo *stitchMeshes* que permite “coser” dos regiones diferentes de la malla en una superficie que ambas regiones compartan, generando nuevas celdas en la interfaz de los dominios, controlando la generación de las nuevas celdas con los parámetros de calidad de malla de que el programa tiene por *default*. Este proceso es ejecutado 4 veces para fusionar la malla en la zona de interfaz entre: la caracola y los alabes directores, los alabes directores y el runner, y el runner con el draft tube. Las superficies utilizadas para coser a las regiones desaparecen al terminar el proceso logrando así tener una sola región en la totalidad del dominio computacional como se puede ver en la Figura 2.10, la parte a) muestra una malla con 4 regiones y la parte b) la malla con una sola región que será utilizada en este estudio.

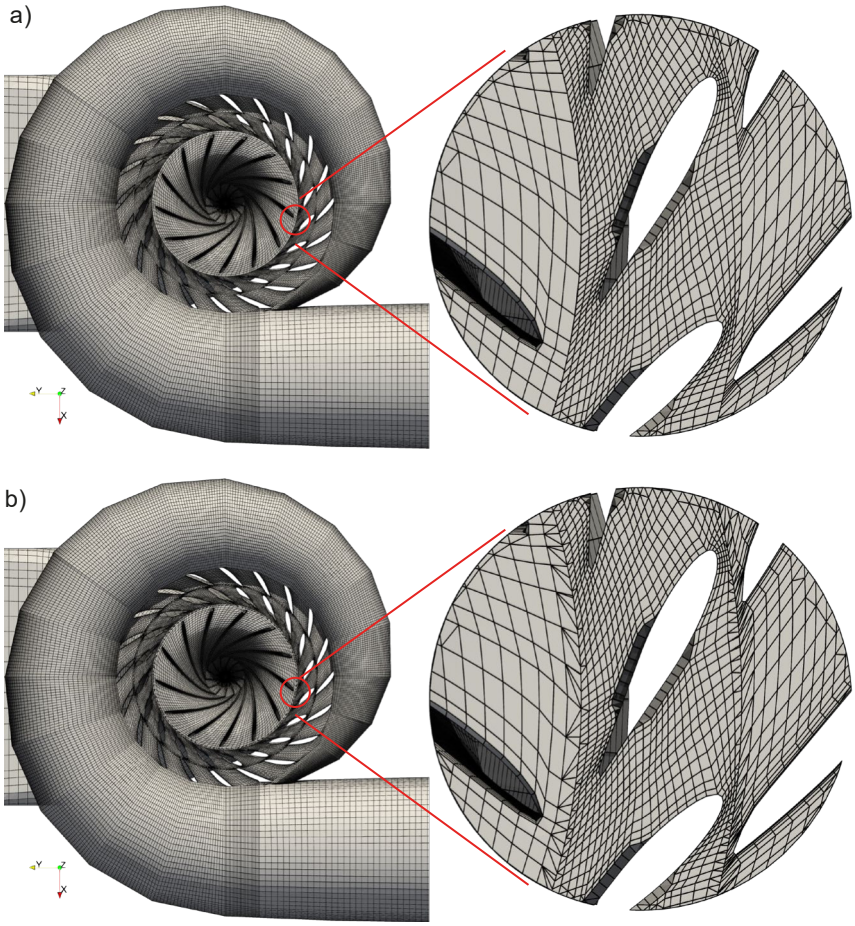

Figura 2.10 Detalle de la diferencia entre malla multirregión (a) y malla de región única (b). (Fuente: Propia)

2.4 MRF

Para llevar a cabo la simulación, es preciso realizar otras aproximaciones para simplificar el caso y reducir el tiempo de cálculo. Una de ellas es considerar la geometría bajo un régimen de cuerpo sólido en rotación en conjunto con condiciones estacionarias que se simulan como un grupo de ecuaciones tipo "*single rotating frame of reference" o en español,* "un único marco de referencia rotacional" situado en una zona de la malla estática. Para este análisis la rotación se obtiene tras interpretar la fuerza centrífuga y la fuerza de Coriolis, y el campo de velocidades ingresa al cálculo como velocidad relativa.

OpenFOAM incluye esta funcionalidad a la simulación como MRF (Multiple rotating frame) que facilita la inserción al cálculo numérico múltiples zonas de cálculo tipo single rotating frame of reference. Para este caso de estudio se va a utilizar solamente una zona de referencia rotacional que incluirá toda la parte del rotor como se muestra en la

Figura 2.11. En esta figura se representa en color rojo toda la región que se va a tratar como elementos en rotación.

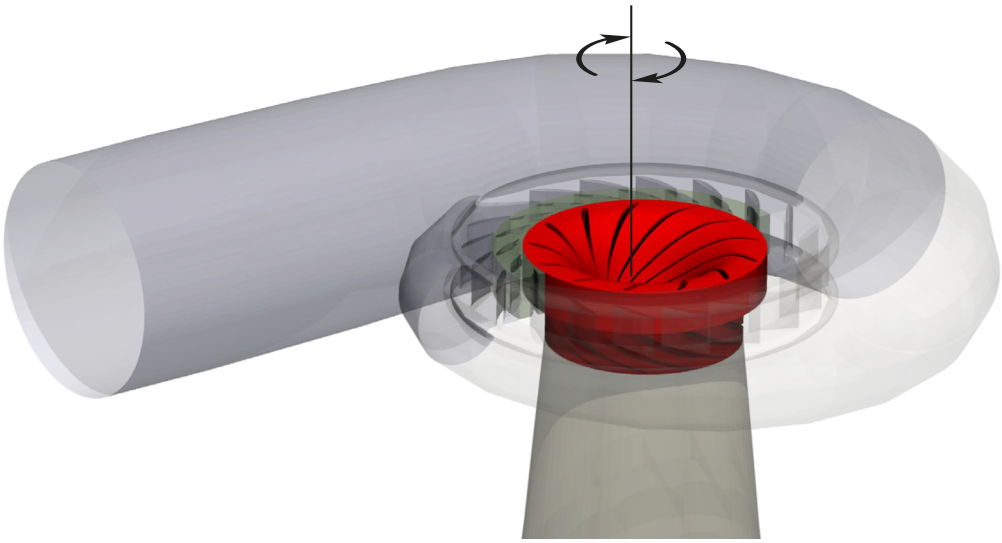

Figura 2.11 Zona MRF en el dominio computacional.

2.5 Calidad de malla.

Para asegurar una convergencia correcta y más eficaz es importante controlar los parámetros de calidad de malla, como se menciona en el capítulo 1. Para verificar estos parámetros es preciso ejecutar el comando *checkMesh* en el entorno de OpenFOAM tras realizar todo el proceso mencionado anteriormente, este comando muestra las características básicas de la malla en base a los parámetros existentes dentro de OpenFOAM y el tipo de malla que contenga el caso. Para este estudio se empieza a trabajar con una malla de una sola región con los siguientes valores estadísticos:

Tabla 2.2 Datos estadísticos de la malla.

Tipo de elementos	Hexaédricos
Número de elementos	1039294
Número de nodos	1155127
Numero de Parches	7
Máx. relación de aspecto	90.388
Min. Área de superficie	5.45e-7 [m2]
Min. Volumen	5.44e-7 [m3]
Max. Oblicuidad	13.79

(Fuente: Propia)

2.6 Simulación estable método SIMPLE

Para flujos incompresibles analizados con métodos numéricos el enfoque de corrección de presión es el más utilizado y estudiado por la comunidad de CFD [54]. En 1972 Patankar y Spalding desarrollan el primer algoritmo de corrección de presión nombrado SIMPLE que son las siglas de *Semi Implicit Method for Pressure Linked Equations* que se basa en las siguientes aproximaciones [54]:

- El campo de presión inicial y los campos de velocidad inicial se asumen de forma independiente, por lo que se descuida la interconexión inherente entre la presión y las velocidades, lo que da lugar a alguna inconsistencia entre ellos.
- Los efectos de las correcciones de presión de las redes vecinas se eliminan arbitrariamente para simplificar el procedimiento de solución, por lo que el algoritmo es semi - implícito.

Este algoritmo es principalmente utilizado para soluciones en estado estable pues no se atiende al cambio de las variables a través del tiempo sino al cambio en base a la presión y a la velocidad. Este algoritmo se lo puede entender revisando el diagrama de flujo de la Figura 2.12 a continuación:

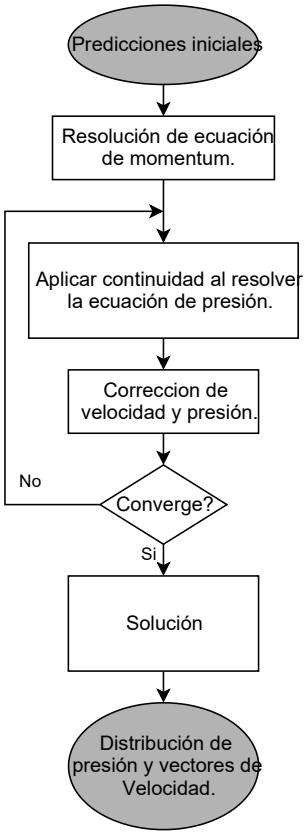

Figura 2.12 Diagrama de flujo algoritmo SIMPLE.
(Fuente: [7])

En estado estable no existe ningún movimiento relativo de mallas ni de regiones ya que se asume que no existen cambios en las propiedades ni las posiciones de los elementos en el tiempo. Para ajustar esta suposición al dominio de nuestro estudio se aplicará el enfoque de rotor congelado *(Frozen rotor)* que será llevado a cabo correctamente con la inserción de una zona MRF y así separar un marco de referencia rotacional en el volumen el rodete de la turbina mientras que al resto de la malla se lo toma como un marco referencial estacionario como se ve en la Figura 2.11.

2.6.1 Condiciones de operación de la central hidroeléctrica.

Las simulaciones llevadas a cabo para este estudio sustentan su cálculo sobre las condiciones de operación de la turbina Francis existente en la Central hidroeléctrica San Francisco. Esta información es de libre acceso al público desde el portal web de CELEC EP [17] y a continuación se resumen las características más importantes que utiliza Velasco para llevar a cabo su investigación [7] y que se utilizará para replicar sus resultados junto a la adición de material particular.

Tabla 2.3 Condiciones de operación de la turbina Francis.

Fabricante	VATECH
Potencia nominal	115 [MW]
Caudal Nominal de trabajo	53 [m3/s]
Altura aguas bajo	6,36 [m]
Rendimiento	95,5% [m]
diámetro entrada caja espiral	3 [m]
temperatura del agua	16 [°C]
velocidad de rotación	34,27 [rad/s]
Sentido de giro	Horario desde vista superior
Densidad	998,9 [kg/m3]
Viscosidad cinemática	1,106e-6 [m2/s]

(Fuente:[17])

2.6.2 Condiciones de Borde y de Frontera.

A continuación, se detalla de manera puntual el origen de los valores que serán utilizados como condiciones de iniciales y condiciones de borde, tanto en este caso de estudio como en el de sus predecesores: Mora [5], Guascal & Quispe [53] y Velasco [7].

Partiendo de datos experimentales presentadas por CELEC EP en el informe de rendimiento de la central San Francisco en el 2016 y se presenta en la Tabla 2.4

Tabla 2.4 Valores experimentales de potencia y rendimiento de la central San Francisco a diferentes caudales.

Q [m3/s]	P1 [Pa]	P2 [Pa]	Potencia E [MW]	Potencia H [MW]	Rendimiento Turbina [%]
			Medido	(P1-P2) *Q	Pot E/Pot H
17,0	2100000	71000	29,3	34,5	0,85
25,9	2090000	72000	40,3	52,3	0,77
33,4	2070000	75000	55,5	66,6	0,83
33,2	2060000	70000	56,2	66,1	0,85
40,3	2050000	75000	70,6	79,6	0,89
45,1	2050000	80000	79,2	88,8	0,89
50,8	2000000	80000	90,9	97,5	0,93
56,0	2000000	80000	100,1	107,5	0,93
59,7	1950000	80000	105,8	111,6	0,93
62,4	1950000	80000	109,5	116,7	0,94

(Fuente:[17])

Aquí se presenta el valor experimental de presión en pascales a la entrada de la turbina (P1) y a la salida del tubo de descarga (P2) a diferentes caudales. Es claro que el caudal que produce el mayor rendimiento es el de 62.4 $\frac{m^3}{s}$ por lo que se lo utilizará para calcular el resto de las condiciones iniciales.

- **Presión**

En primero lugar tenemos la presión que va a estar presente en el parche OUT_DRAFT con unidades de presión normalizada, que no es más que la división de la presión a la salida con la densidad del fluido en la temperatura a la que esté trabajando así obtenemos una magnitud de unidades [m2/s2] o también llamada presión normalizada.

$$p_{Normalizada} = \frac{P}{\rho_{agua}} = \frac{80000\ [Pa]}{998.9\ \left[\frac{kg}{m^3}\right]} = \mathbf{80.0881}\ \left[\frac{\boldsymbol{m^2}}{\boldsymbol{s^2}}\right] \tag{2.1}$$

- **Velocidad**

La velocidad de entrada en el parche IN_SPIRAL fue calculada en función del Caudal de entrada (Q) y el área transversal (A) obtenidas de las tablas Tabla 2.4 y Tabla 2.3.

$$v = \frac{Q}{A} = \frac{62.4\left[\frac{m^3}{s}\right]}{\pi * \left(\frac{3[m]}{2}\right)^2} = \mathbf{8.8278}\ \left[\frac{\boldsymbol{m}}{\boldsymbol{s}}\right] \tag{2.2}$$

- **Turbulencia**

Para continuar el análisis es necesario determinar la naturaleza turbulenta de este fenómeno para lo cual usaremos los valores ya calculados en las ecuaciones anteriores y los valores experimentales de la Tabla 2.3 para determinar el parámetro adimensional de Reynolds, este número adimensional relaciona las fuerzas inerciales y las fuerzas viscosas [29] y su valor para este caso viene dado por:

$$Re = \frac{L_c * U}{\vartheta} = \frac{3\ [m] * 8.204 \left[\frac{m}{s}\right]}{1.106e^{-6} \left[\frac{m^2}{s}\right]} = \mathbf{2.225e^7} \tag{2.3}$$

L_c representa la longitud característica en metros, que es el diámetro de la entrada de la turbina, U es la velocidad de flujo libre o sea la velocidad en metros sobre segundo a la entrada de la caracola y ϑ es la viscosidad cinemática en metros cuadrados sobre segundo.

El desarrollo de la ecuación (2.3) concluye en un número de Reynolds de gran magnitud, según Fox et al. Cuando el número de Reynolds supera el valor de 2300 es un flujo turbulento, por lo que el fenómeno que analizaremos es altamente turbulento y para estudiarlo de manera apropiada debemos incluir en el estudio los efectos de turbulencia.

Velasco concluye en su estudio que el modelo de turbulencia que más se aproxima a la predicción de potencia y eficiencia es $k - \varepsilon$ pues tiene un error de 5.02% [7] por lo que utilizaremos únicamente este modelo para añadir las variaciones debido a la turbulencia en el caso de estudio.

Los valores iniciales de este modelo de turbulencia son: la energía cinética turbulenta (k) y la tasa de disipación turbulenta (ε). Que serán calculadas a partir de las expresiones de Wilcox. D [55].

$$k = \frac{3}{2}(U \cdot I_T)^2 = \frac{3}{2}(8.827 \cdot 0.1)^2 = \mathbf{1.1689} \left[\frac{\boldsymbol{m^2}}{\boldsymbol{s^2}}\right] \tag{2.4}$$

$$\varepsilon = C_\mu^{\frac{3}{4}} \cdot \frac{k^{\frac{3}{2}}}{l_{turb}} \cdot C_\mu^{\frac{3}{4}} \cdot \frac{k^{\frac{3}{2}}}{0.07 \cdot D_h} = 0.09^{\frac{3}{4}} \cdot \frac{1.0095^{\frac{3}{2}}}{0.07 \cdot 3} = \mathbf{0.9889} \left[\frac{\boldsymbol{m^2}}{\boldsymbol{s^3}}\right] \tag{2.5}$$

En la ecuación (2.4) aparece el término de I_T que hace referencia al porcentaje de intensidad turbulenta, puesto que el fenómeno es altamente turbulento se puede usar valores mayores al 10% [56], por tal ocuparemos ese valor. Por otro lado, tenemos la ecuación (2.5) que muestra el termino C_μ que es igual a 0.09 ya que es un valor constante para el modelo de turbulencia que se empleará [55]. La l_{turb} que significa longitud turbulenta. D_h representa al diámetro hidráulico que en este caso es 3 [m] y por último tenemos el término U que es la velocidad a la entrada de la caracola.

A continuación, se resume en una tabla las condiciones iniciales y de borde de cada uno de los parches de la malla. Es preciso destacar que los parches de la entrada y de salida del dominio son del tipo *patch* y los parches que se ubican en los bordes físicos de la malla son del tipo *wall*.

Tabla 2.5 Condiciones de borde (Velocidad y Presión)

Superficie	U	p
BLADE	movingWallVelocity	zeroGradient
WALL_SPIRAL_CASE	fixedValue	zeroGradient
WALL_GUIDE_VANE	fixedValue	zeroGradient
WALL_RUNNER	movingWallVelocity	zeroGradient
WALL_DRAFT_TUBE	fixedValue	zeroGradient
IN_SPIRAL	FixedValue Ec.(2.2)	zeroGradient
OUT_DRAFT	inletOutlet	fixedValue Ec.(2.1)

(Fuente: Propia)

Tabla 2.6 condiciones de borde (k, epsilon y nut).

Superficie	k	epsilon	nut
BLADE	kqRWallFunction	epsilonWallFunction	nutUSpaldingWallFunction
WALL_SPIRAL_CASE	kqRWallFunction	epsilonWallFunction	nutUSpaldingWallFunction
WALL_GUIDE_VANE	kqRWallFunction	epsilonWallFunction	nutUSpaldingWallFunction
WALL_RUNNER	kqRWallFunction	epsilonWallFunction	nutUSpaldingWallFunction
WALL_DRAFT_TUBE	kqRWallFunction	epsilonWallFunction	nutUSpaldingWallFunction
IN_SPIRAL	fixedValue Ec. (2.4)	fixedValue Ec. (2.5)	calculated Ec. (2.3)
OUT_DRAFT	zeroGradient	zeroGradient	calculated

(Fuente: Propia)

2.6.3 Solver (simpleFoam)

Siguiendo el flujograma presentado al inicio de la sección de metodología, el paso siguiente es simular este fenómeno en estado estable para obtener resultados de una simulación tipo Frozen Rotor con una malla fija de una sola región optimizada para una simulación con material particular y compararlos con los resultados que obtuvo Velasco utilizando una malla multirregión [7], para así poder validar estos resultados y utilizarlos como punto de partida para simulaciones en estado transitorio. El objetivo de esto es generar un estado de estabilidad validado y una vez resuelto, aplicarlo como condición

inicial para las simulaciones de estado transitorio que utilizan el mismo algoritmo PISO añadiéndole la fase sólida particular.

El algoritmo SIMPLE es un proceso iterativo para ecuaciones de acoplamiento de conservación de masa y de momento, asumiendo que no hay cambios representativos a través del tiempo. En este punto del estudio del caso se utilizará la herramienta *simpleFoam* que fundamenta en este algoritmo para simular la naturaleza de un flujo. En la Tabla 2.7 se resumen los ajustes del *solver* para la simulación de *frozen rotor* llevada a cabo, estos datos se pueden encontrar en el archivo *fvSolution.* La palabra *solver* hace referencia a los métodos para solucionar ecuaciones que se utiliza en la simulación, estos varían dependiendo del fenómeno y definen la precisión de los resultados y controlan los cálculos del simulador. Los parámetros de tolerancia indican el residual objetivo que se debe lograr antes de continuar con el proceso iterativo para que los resultados converjan correctamente [8].

Tabla 2.7 Ajustes de Solvers para las variables simuladas en estado estable.

Campo	Solver	Tolerance	Rel.Tol	smoother
p	GAMG	1e-6	0,05	GaussSeidel
U	smoothSolver	1e-7	0,10	GaussSeidel
k	smoothSolver	1e-7	0,10	GaussSeidel
epsilon	smoothSolver	1e-7	0,10	GaussSeidel

(Fuente: propia)

En esta sección es necesario controlar los valores residuales pues estos valores indican la cantidad de error o el desequilibrio que tienen las variables en otras palabras miden el nivel de convergencia en que se encuentra la solución. Aakti [57] menciona que un valor de residuales menores o iguales a 10e-3 en una simulación inestable es un indicador de convergencia y de 10e-2 para simulaciones en estado estable. En este caso de estudio, el valor que se analiza para controlar los residuales debe ser menor o igual que la tolerancia mostrada en Tabla 2.7 para que la simulación converja.

2.7 Simulación transitoria método PISO

Una vez lograda la simulación en estado estable y habiendo verificado los resultados obtenidos, utilizaremos los campos resueltos de velocidad, presión, k y épsilon para llevarlos a una simulación en estado transitorio, este proceso se lo llevará a cabo en dos etapas:

- Simulación transitoria método PISO.
- Simulación transitoria CFD-DEM acoplada.

La primera etapa depende de una simulación transitoria que no contendrá fase particular, simplemente se la lleva a cabo para un control previo del número de Courant, residuales y calcular una zona de MRF en el dominio a través del tiempo, puesto que esta característica no se encuentra en el software CFDEM y es importante para caracterizar la naturaleza del flujo, esta etapa se la desarrollará en un entorno puramente de CFD. En esta primera etapa se pule todos los errores que puedan afectar a la calidad de los resultados, es importante hacerlos pues el gasto computacional aquí todavía no es tan demandante como la etapa siguiente.

La segunda etapa empieza tras la inserción de partículas creadas anteriormente en el entorno de LIGGGHTS. Esta inclusión se logra con método de acoplamiento CFD-DEM tras haber llegado a un estado de convergencia en la primera etapa y después de haber corregido errores de no convergencia, calidad de malla, topología, etc., en los procedimientos.

Se ha llevado a cabo el estudio de este modo debido a que el software que realiza el acoplamiento CFD-DEM basa su cálculo en *pisoFoam* un *solver* nativo de *OpenFOAM*. Por lo que partir de un campo de variables resueltas en este *solver* va a garantizar una convergencia correcta a la solución en estado transitorio con fase particular de nuestro caso de estudio sin embargo el análisis CFD-DEM se lo llevará a cabo más adelante.

2.7.1 Solver (pisoFoam).

El algoritmo PISO (Presión implícita con división del operador) fue propuesto por Issa en 1986. Este es un procedimiento de cálculo implícito de presión-velocidad para las ecuaciones de Navier-Stokes desarrollado originalmente para el cálculo no iterativo de flujo comprimible no estacionario, pero se ha adaptado con éxito a problemas de estado estacionario.

OpenFOAM tiene una gran cantidad de solvers que basan su funcionamiento en el algoritmo PISO, este algoritmo es un proceso iterativo para resolver ecuaciones de acoplamiento de momentum y conservación de masa, este algoritmo es utilizado para simulaciones en estado transitorio. En la Figura 2.13 se muestra el diagrama de flujo de este algoritmo, como se puede ver en este proceso iterativo una vez que converge el cálculo se vuelve a establecer las condiciones calculadas como condiciones iniciales y el cálculo continuo hasta que la sumatoria de tiempos alcance el tiempo objetivo.

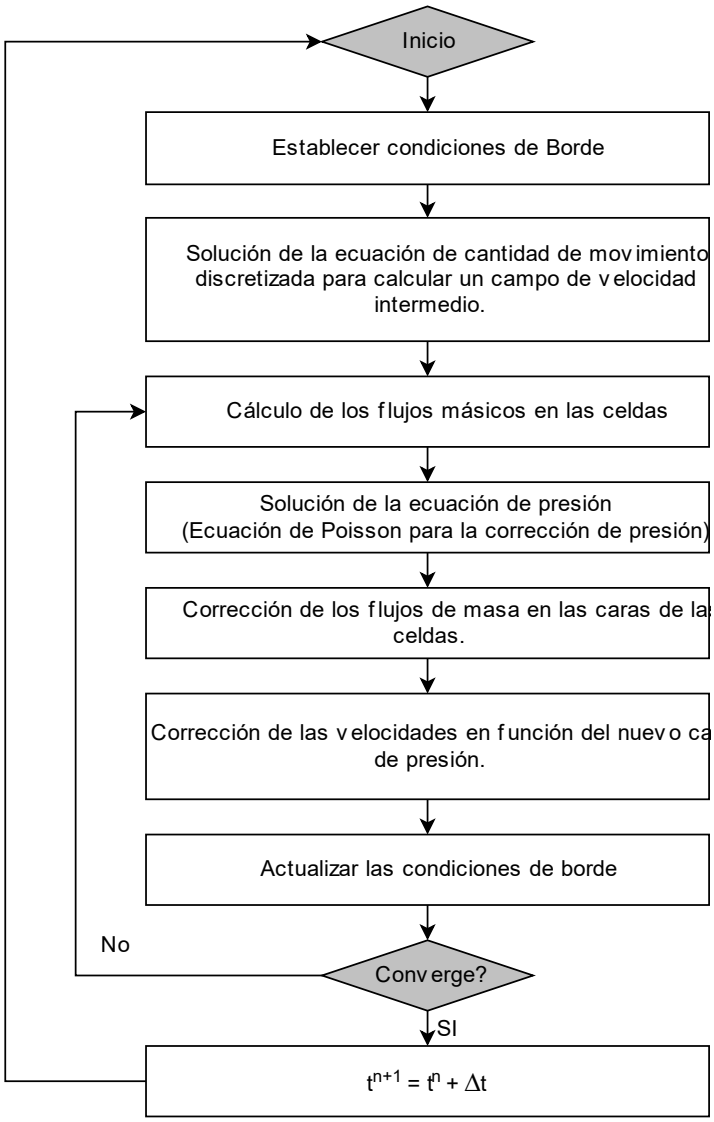

Figura 2.13 Diagrama de flujo algoritmo PISO.
(Fuente: [58])

2.7.2 Condiciones de Borde

Las condiciones de borde serán las mismas utilizadas para la solución en estado estable ya que la velocidad de entrada de la turbina es la misma, las paredes mantienen su condición de *WALL* y la salida del tubo de descarga usa el mismo valor presión, La variante con respecto a las condiciones iniciales del estado estable es el campo interno de la malla o también llamado *internalfield.* Ya que estos campos se encuentran resueltos a diferencia del estado estable que se ingresa un escalar o un vector de referencia para que empiece el cálculo iterativo. Por tal motivo estas condiciones iniciales se cambian por los campos que contiene toda la información resuelta de la malla obtenidos de la simulación acoplada por el método SIMPLE.

En palabras más resumidas, se toma la solución del algoritmo SIMPLE y se la modifica para usarla como condición inicial para el algoritmo PISO.

Este proceso demuestra una clara ventaja en tiempo de convergencia para simulaciones en estado transitorio en comparación a una simulación que inicia su cálculo únicamente con los valores de sus condiciones de frontera como muestra Belamri en su estudio de turbomaquinaria presentado en la ASME Turbo Expo del 2005 [59].

2.7.3 Número de Courant

Como se mencionó anteriormente, esta primera etapa de simulación tiene el objetivo de controlar la estabilidad del cálculo en estado transitorio, y ajustar los parámetros de simulación. Esto se plantea lograr controlando el número de Courant.

El número de Courant es un número adimensional que se puede utilizar en simulaciones de dinámica de fluidos computacional (CFD) para evaluar los requisitos de paso de tiempo de una simulación transitoria para un tamaño de malla y una velocidad de flujo dados y está vinculado a la condición de estabilidad de Courant-Friedrichs-Lewy (CFL) de los esquemas numéricos. Este valor adimensional indica en términos generales cuánto viaja la información (u) a través de una celda de cuadrícula computacional (Δx) en una unidad de tiempo (Δt) [60]. En la ecuación (2.6) se puede ver la relación de este número y el resto de los factores.

$$Co = \frac{u \cdot \Delta t}{\Delta x} \tag{2.6}$$

En donde: Co es el número de Courant.

Δt es el intervalo de tiempo.

Δx es el intervalo de espacio.

u es la velocidad.

El método PISO al ser un algoritmo implícito la condición de Courant debe ser igual a:

$$C_{Max} \leq 1 \tag{2.7}$$

Es importante mantener esta condición hasta alcanzar la estabilidad en estado transitorio para poder insertar las partículas al cálculo y asegurar que los resultados de la segunda etapa no van a divergir después de un tiempo de simulación.

En esta etapa se va a procurar ajustar el intervalo de simulación al más amplio posible, pues utilizar un intervalo muy corto incrementará el tiempo real de cálculo a valores sumamente extensos por tal motivo se va a ajustar siempre al valor más amplio posible

que genere un numero de Courant inferior a 1 utilizando la malla optimizada que generamos al inicio del estudio, en caso de ser necesario se ajustará también la malla y se repetirá todo el proceso hasta este punto.

2.8 Simulación DEM

Una vez obtenidos los resultados esperados de la simulación en estado estable es preciso empezar a estructurar el caso previo de simulación de la fase discreta para continuar la metodología propuesta en la Figura 2.1 . Este proceso se lo realizó en simultaneo a las simulaciones CFD pues para llevarse a cabo solo necesita de la geometría obtenida después del proceso de mallado e insertarla como dominio de límite para la generación de partículas. Dominio que será usado posteriormente dentro del caso acoplado de simulación.

Este caso de simulación DEM permitirá la generación del material particular, se obtendrá la posición en el dominio de simulación, el valor del radio y la velocidad inicial de cada uno de los granos. Este paso es necesario e ineludible ya que es una condición inicial para el caso acoplado.

Para lograr esto es necesario determinar las características del material granular existente en el cauce del río que alimenta turbina tipo Francis de la central hidroeléctrica San Francisco ubicada entre la cuenca media y baja del Río Pastaza.

2.8.1 Material granular en la zona de estudio.

La carga de sedimentos en los ríos del Ecuador es bastante alta, sobre todo en la región amazónica la cual se ve influenciada por aportes sedimentarios de las cuencas andinas. Varios proyectos hidroeléctricos construidos a lo largo del río Pastaza con el fin de aprovechar este recurso hídrico en la generación de energía eléctrica, se han visto inmersos en problemas de acumulación de material sedimentable que destruyen progresivamente los componentes de las máquinas hidráulicas. Este proceso de alteración gradual de las geometrías de las superficies representa un reto para la ingeniería en los procesos de operación [61]. El fenómeno de erosión que afecta directamente a la estructura de captación y turbinado produce un desgaste hidro-abrasivo en los componentes hidráulicos, que representan en gastos de mantenimiento y pérdidas de eficiencia en la generación de energía eléctrica.

La cuenca del río Pastaza es una de las cuencas más importante del país con un elevado potencial hídrico de 82.900 $\frac{m^3}{año}$ por habitante [17], nace de la intersección de dos ríos en las faldas del volcán Tungurahua atravesando grandes distancias hacia

territorio peruano. La presencia de partículas en este río está vinculada con distintos factores ambientales y cinemáticos que promueven el arrastre de sedimentos, generalmente originados por desprendimiento del suelo, ya sea a causa de la erosión o como restos de flujos volcánicos.

Las características de los sedimentos dependen de su origen y del proceso de su transporte siendo el tamaño, la característica de mayor importancia en estas partículas ya que representa su volumen e interacción con el flujo [62]. En ríos con un nivel pluvial bastante elevado es común encontrar gran cantidad de sedimentos, los cuales se clasifican en la Tabla 1.1 Particularmente en el río Pastaza, se encuentran sedimentos de tipo clásticos gruesos como producto de la actividad volcánica del Tungurahua, así como sedimentos fluviales del tipo más fino. La litología de la zona comprende conglomerados gruesos, tobas, arenas y arcillas.

A lo largo de la Cuenca del río Pastaza existe una variedad de ríos que presentan distintas características en función de su altitud. Los ríos de zonas medias (1100-2200 m) presentan pendientes fuertes, alta velocidad del agua, mayor temperatura del agua y cauces aluviales. En este tipo de cauces predomina la sedimentación de arenas y en casos más locales, la concentración de gravas [63]. Esta acotación contiene las características de la zona de estudio, por tal motivo, en el presente trabajo se utilizan los tamaños de partículas que corresponden a la clasificación denotada en la Tabla 2.8, allí se muestra la clasificación según la *American Geophysical Union,* Junto a su diámetro aproximado y su densidad.

Tabla 2.8 Tamaño y densidad de partículas presentes en la cuenca del rio Pastaza.

Tipo de partícula	Diámetro [mm]	Densidad seca [kg/m^3]
Grava mediana	10	1500 -1800
Grava fina	5	1400 – 1500
Grava muy fina	2,5	1300 – 1400

(**Fuente**: propia)

2.8.2 Características del material particular

En vista que el material particular se clasifica en grava y arena se utilizará las propiedades estos sedimentos para el cálculo. Las propiedades necesarias para llevar a cabo el método de elemento discreto dependen del tipo de modelo de contacto que se utilice para describir el fenómeno. Como se explicó en el inciso 1.6.4 el modelo de Hertz se ajusta al fenómeno estudiado, este modelo exige como requerimiento la siguiente información del material particular:

- Módulo de Young

Este módulo refleja la rigidez de los materiales terrestres. El módulo de Young (E*)* es la relación de tensión a deformación.

Ollivela en su publicación de mecánica de suelos indica que el módulo de Young para la grava, la arena y la arcilla varia en el rango de 4.5×10^6 y 5.5×10^6[64]. Por lo que utilizaremos un valor promedio de 5×10^6.

- Coeficiente de Poisson

El coeficiente de Poisson relaciona la deformación en materiales en direcciones perpendiculares entre sí frente una carga aplicada en una dirección. La grava y la arena al ser cerámicos presentan una relación de Poisson relativamente baja que se presenta en la siguiente tabla:

Tabla 2.9 Relación de Poisson para diferentes sedimentos.

Tipo de material	Relación de Poisson ν
Arcilla	0.40 – 0.45
Arena	0.30 – 0.40
Grava	0.15 – 0.25

(Fuente:[64])

Dado que no se puede ingresar un rango de valores en la definición de las características del material se utilizará un valor promedio de 0.3 en la relación de Poisson.

Estas dos relaciones son necesarias para definir el modelo de contacto utilizado en el estudio. Una vez determinadas las propiedades internas del material particular es preciso caracterizar a los granos en el entorno de simulación para lo cual se aproximará su forma a una esfera perfecta con el diámetro que se encuentra detallado en la Tabla 2.8. También es preciso proporcionar la cantidad de partículas que se va a generar y la distribución de partículas según el radio de las esferas.

En resumen, se presenta la Tabla 2.10 con los datos utilizados para la generación de las partículas en DEM.

Tabla 2.10 Propiedades del material particular.

Nombre	Módulo de Young (E)	Coef. De Poisson (v)	Coef. de Restitución	Coef. de fricción	Densidad [kg/m3]	Radio [m]	Volumen [m3]	Cantidad	Masa [kg]
Grava mediana	$5x10^6$	0,3	0,3	0,5	1800	0,01	4,2E-06	100	0,754
Grava mediana	$5x10^6$	0,3	0,3	0,5	1500	0,0085	2,6E-06	100	0,386
Grava fina	$5x10^6$	0,3	0,3	0,5	1500	0,007	1,4E-06	100	0,216
Grava fina	$5x10^6$	0,3	0,3	0,5	1400	0,0055	7,0E-07	100	0,098
Grava muy fina	$5x10^6$	0,3	0,3	0,5	1400	0,004	2,7E-07	100	0,038
Grava muy fina	$5x10^6$	0,3	0,3	0,5	1300	0,0025	6,5E-08	100	0,009
Total								**600**	1,499

(Fuente: Propia)

En vista que las partículas presentan una dimensión minúscula en comparación al dominio computacional y se encuentran muy alejadas entre ellas, se va a desestimar el coeficiente de fricción y restitución pues estos no van a interferir en la naturaleza del fenómeno físico.

2.8.3 Preproceso de DEM

El proceso de cálculo DEM en LIGGGTHS está sujeto a un proceso previo para poder ser ejecutado, este proceso consiste básicamente en generar el dominio computacional de las partículas y posicionar espacialmente las partículas dentro del dominio. En este método de cálculo no es necesario insertar una malla como en CFD, aquí basta con incluir las coordenadas máximas y mínimas de un prisma rectangular e incluir las superficies que vayan a cumplir el papel de paredes para calcular las colisiones y cambios de trayectorias de las partículas cuando interactúen con ellas.

Para lograr este preproceso es necesario haber concluido la etapa de mallado, porque de este proceso vamos a obtener las coordenadas limites que usaremos para definir el prisma rectangular y vamos extraer a los parches tipo *WALL* de la malla como superficies de formato *Stereolithography* (.stl) y tratarlas en Blender para reducir el número de elementos y mejorar la calidad de la superficie e insertar la nueva superficie *WALL ligera* como paredes en el dominio de DEM que posteriormente también serán reutilizadas para CFDEM.

En la Figura 2.14 se muestra con borde negro el prisma rectangular que incluye toda la zona de estudio, esta zona debe incluir el dominio computacional de CFD y las superficies de la geometría de las paredes de la turbina. En la Tabla 2.11 se muestran las coordenadas que definen este dominio, estas coordenadas han sido aumentadas en un decimal para asegurar una inclusión total de todos los dominios de las simulaciones.

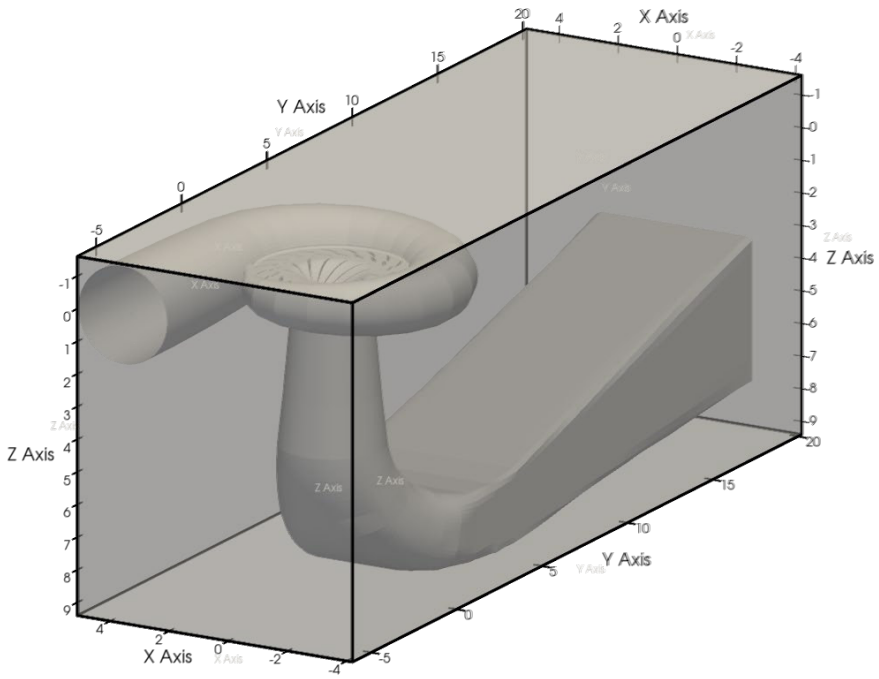

Figura 2.14 Dominio computacional DEM.
(Fuente: Propia)

Tabla 2.11 Coordenadas del dominio computacional de DEM.

	Min	Max
x	-4,2	5,1
y	-6,2	20,1
z	-1,6	9,4

(Fuente: Propia)

En la Figura 2.14 se muestra además la geometría de la superficie de contacto de la turbina esta superficie fue generada a partir de los parches tipo *wall* de la malla obtenida al inicio de la metodología. Esta superficie debió ser modificada debido que contiene una cantidad enorme de triángulos y esto produce problemas en el cálculo de condiciones de contacto en el simulador DEM por tal motivo esta superficie será aligerada con ayuda de Blender, este es un software de animación 3D, edición de video y VFX (efectos especiales) libre y de código abierto [65], que contiene numerosas herramientas para la creación y modificación de superficies. Para logar nuestro objetivo utilizaremos el modificador *Decimate* que permite reducir el recuento de vértices/caras de una malla con cambios de forma mínimos. Este aplicativo se utiliza para reducir el número de elementos para aumentar el rendimiento eliminando los vértices y aristas excedentes. Esta condición es necesaria para este proceso ya que el consumo de memoria RAM aumenta considerablemente según aumenta la cantidad de vértices en la estereolitografía (archivo .stl) además que LIGGGHTS ocupa criterios de calidad superficial más exigentes a los de OpenFOAM y este proceso ajusta las caras a

triángulos más uniformes sin descuidar los detalles principales de la geometría, presentando una ventaja ante la opción de enviarlos directamente como superficie WALL desde el proceso de mallado hasta DEM.

En la Tabla 2.12 se detalla en resumen las características principales de ambas superficies.

Tabla 2.12 Estadísticas de las superficies tipo Wall.

	Superficie WALL	Superficie WALL ligera
Numero de triángulos	294.174	73.543
Numero de lados	441.339	110.363
Numero de vértices	147.061	36.716
Cantidad de Memoria	26MB	117MB
Relación de aspecto máxima	1.6e6	3447
Relación de aspecto promedio	17,663	3,437

(Fuente: Propia)

2.8.4 Modelo de desgaste

El modelo de desgaste que utiliza LIGGGHTS para el contacto granular es conocido como modelo de contacto de Finnie y se basa en dos partes. La primera parte es el cálculo de la dirección (ángulo de impacto) y la velocidad de las partículas que golpean la superficie dúctil. Los materiales frágiles pueden sufrir fracturas tras el impacto y requieren un enfoque diferente del modelo matemático sin embargo se realiza este estudio bajo la suposición que las partículas no se fracturarán bajo ningún tipo de carga y que son esferas perfectas [66].

El modelo asume material dúctil, por lo que el modelo calculará la cantidad de material superficial eliminado en función de las trayectorias de las partículas. Nota: aunque el modelo calcula el volumen de material eliminado, las velocidades de las partículas dependen del tamaño del material, que es una estimación inicial, por lo que el efecto del tamaño de las partículas en la erosión es relativamente incierto y es un enfoque cualitativo del desgaste.

Los materiales como el acero se desgastan por un proceso de deformación plástica en el que el material es removido por la acción de desplazamiento o corte de la partícula erosionada. Las suposiciones detrás del modelo son que existe una relación entre el componente de fuerza normal y el componente de corte de valor constante K. Esto es cierto si la rotación de la partícula durante el corte/impacto es bastante pequeña. Además, se supone que existe una relación entre la profundidad de contacto [66].

Este modelo tiene como resultado un enfoque cualitativo de la cantidad de material removido, puesto que se realizan muchas aproximaciones, y se lo utilizará para realizar

comparativas entre los resultados obtenidos en la simulación de posibles zonas de desgaste por abrasión de partículas, con información encontrada en estudios anteriores referentes a las zonas de desgaste de los elementos de la turbina.

2.9 Acople CFD-DEM:

Una vez obtenido el cálculo inicial de DEM y la simulación controlada con el algoritmo PISO es necesario incluir esta información en la simulación CFD-DEM para analizar el comportamiento de las partículas junto al flujo y viceversa. El proceso de acople a nivel de cálculo se explicó en la Figura 1.8. Entonces, para ejecutar este cálculo, se va a organizar la información generada hasta este punto de manera tal que el software de simulación CFD-DEM lo requiera.

De esta simulación se espera obtener los contornos de velocidad, presión y principalmente determinar las zonas de mayor desgaste por impacto de partículas. Mientras que del flujo particular se obtendrá las posibles trayectorias de las partículas intentando identificar particularidades en base al radio de las diferentes partículas incluidas.

2.9.1 Estructura de archivos.

En vista que CFDEM es un programa de simulación a nivel de código que no cuenta con interfaz gráfica, es importante mantener una estructura de archivos bien ordenada acorde los requerimientos del programa y así lograr que la simulación una configuración previa fácil de entender y fácil de modificar según sea la necesidad, en la Figura 2.15 se muestra la disposición de los directorios y archivos necesarios para llevar a cabo la simulación del caso de estudio.

El software que se va a hacer cargo del acople es CFDEM y este se encarga de utilizar el motor de cálculo de OpenFOAM y añadirle la información calculada y generada a lo largo de la simulación por LIGGGHTS según se lo programe.

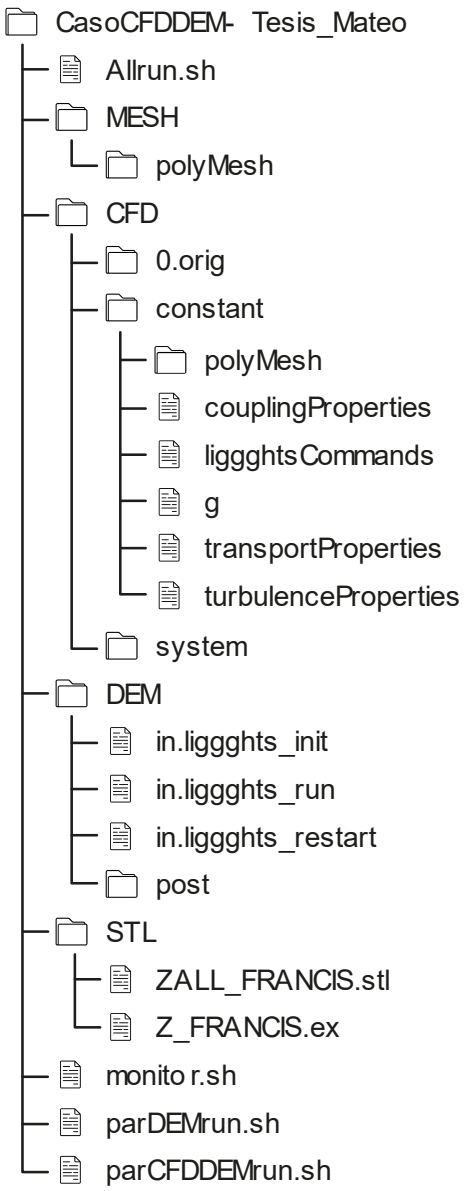

Figura 2.15 Árbol de archivos para el caso de simulación de la turbina tipo Francis con flujo particular.

(Fuente: Propia)

En el árbol de archivos se muestra los archivos necesarios para poder lograr la simulación. Comenzando por el ejecutable del programa *Allrun.sh,* este script contiene los comandos y las configuraciones para llevar a cabo el acople, también se encarga de reestablecer los ficheros para empezar la simulación desde cero, paso seguido analiza si hay un cambio en la malla, de haberlo copia los nuevos archivos en la carpeta *constant*. Este archivo tiene otra característica muy importante que es identificar si el trabajo en curso ha concluido y en caso de no hacerlo programar los archivos para continuar la simulación, esto es fundamental ya que estamos tratando un cálculo que tarda aproximadamente 240 horas en terminar y empezar de cero cada vez que esta la

simulación tenga que pararse presenta un grave problema. La configuración de este script se lo detalla en el ANEXO II.

Podemos ver también el directorio MESH, que contiene la malla generada al comienzo de la metodología, esta malla tiene el formato OpenFOAM (*polymesh)* y será insertada en la carpeta *constant* del directorio CFD cada vez que sea modificada.

A continuación, está la carpeta CFD que contiene un caso estándar de simulación de OpenFOAM, la única distinción que se puede hacer es la presencia del archivo *couplingProperties* y *liggghtsCommands.* Estos archivos contienen la información de acople de la simulación CFD con DEM, *couplingProperties* tiene los valores de ajuste y configuración necesarios para la interacción entre los dos programas de cálculo y *liggghtsCommands* contiene la información de lectura y escritura de los archivos de DEM acorde el progreso de la simulación esta información puede ser encontrada en los ANEXO III y IV.

Después encontramos el directorio DEM, Este directorio contiene la información necesaria para llevar a cabo el cálculo de las condiciones iniciales del material particular en el archivo *in.liggghts_init* y del cálculo iterativo consiguiente para el estado transitorio de las partículas en interacción con el CFD en el archivo *in.liggghts_run* en otras palabras este script contiene la configuración y ajustes para el cálculo de la fase discreta estos archivos están disponibles en los ANEXOS V, VI y VII. El directorio *post* almacena la información generada de la posición, velocidad y radio de las partículas en formato *Visualization ToolKit* (.vtk).

En el caso de simulación está presente también la carpeta STL que contiene las superficies modificadas y optimizadas para el cálculo de las zonas de contacto en el dominio de DEM y CFDEM. En conjunto un archivo de extensión .ex que contiene la ubicación de los elementos que serán evitados en el cálculo dada su baja calidad.

Se puede ver también tres archivos con extensión *Shell script* (.sh). El archivo *parDEMrun* contiene la información de ejecución de la etapa DEM como la ubicación del archivo *in.liggghts_init,* el nombre de los archivos de control tipo “log” y el número de procesadores que se va a utilizar en la paralelización de esta etapa.

El archivo *parCFDDEMrun* contiene la misma información que el archivo anterior pero direccionada a la ejecución de la etapa CFD-DEM, y contiene también le ejecución de utilitarios de postproceso que trabajan en el entorno de CFDEM.

El ultimo archivo de nombre *monitor.sh* contiene un aplicativo generado para este caso de simulación que permite controlar en tiempo real el tiempo de simulación y el número

de Courant máximo y el promedio. Este archivo es de gran ayuda puesto que la información presente en el *stdout* de la ejecución de la simulación CFDDEM es demasiado extensa y no es posible controlar estos valores de otra manera en tiempo real.

2.9.2 Condiciones iniciales y de borde

Las condiciones iniciales se encuentran presentes en la carpeta 0.orig, esta carpeta contiene la información generada el ultimo *timestep* calculado por el algoritmo PISO calculado en la primera etapa de la simulación en estado transitorio.

Sin embargo, estas condiciones iniciales describen únicamente a la fase líquida, por lo que se debe a añadir variables iniciales de los campos de flujo particular en el dominio de CFD, que son:

- *Ksl*, que representa el término implícito de intercambio de cantidad de movimiento entre la fase sólida y la líquida.
- *Us,* representa la velocidad del material particular en el dominio de CFD.
- *voidFraction*, es un valor adimensional representa la existencia o no de fase liquida en una celda, lo que se debe interpretar como si la no existencia de líquido conlleva obligatoriamente la existencia de fase sólida, siendo 1 el valor que indica que la celda tiene únicamente líquido y cualquier valor entre 0 y 1 la existencia de fase sólida.

Tabla 2.13 condiciones de borde para fase solida en CFD (Ksl, Us y voidFraction).

Superficie	Ksl	Us	voidFraction
BLADE	zeroGradient	zeroGradient	zeroGradient
WALL_SPIRAL_CASE	zeroGradient	zeroGradient	zeroGradient
WALL_GUIDE_VANE	zeroGradient	zeroGradient	zeroGradient
WALL_RUNNER	zeroGradient	zeroGradient	zeroGradient
WALL_DRAFT_TUBE	zeroGradient	zeroGradient	zeroGradient
IN_SPIRAL	zeroGradient	zeroGradient	fixedValue (uniform 1)
OUT_DRAFT	zeroGradient	zeroGradient	fixedValue (uniform 1)

(Fuente: Propia)

2.9.3 Ajustes de Acople

Los ajustes de acople detallan la forma en como la información se va a compartir entre el dominio CFD y DEM. Teniendo como principales características:

- Tipo de acople

Como se indica en el capítulo 1, el acople puede ser de tipo A o tipo B y dependen del tipo de formulación que se le dé a la ley de conservación del momento. En donde si la

caída de presión es compartida entre la fase sólida y liquida es un modelo tipo A y si la caída de presión se aplica solamente a la fase liquida es un modelo tipo B.

- Intervalo de Acople

Este valor indica el tiempo que pasa entre dos intercambios de datos CFD-DEM. Por ejemplo:

Si:

$$DEM_{ts} = 1 \times 10^{-5}\ [s]$$

$$Intervalo\ de\ acople = 10$$

Entonces:

$$Acople\ será\ cada\ 1 \times 10^{-4}\ [s]$$

- Localización del archivo de ejecución de CFD-DEM

Este es uno de los valores más importantes ya que el programa de acople debe remitirse a un archivo que contiene todos los valores de configuración, solo cuando este archivo se ha localizado el programa continua con su ejecución iterativa por tal motivo aquí siempre deberá dirigir al archivo liggghts "../DEM/in.liggghts_run" y esta ubicación va a depender de la estructura de archivos que hayamos decidido usar, en este caso la detallada en la Figura 2.15

2.9.4 Paralelización y recurso computacional

La configuración de este caso de simulación supone una carga computacional muy alta para intentar llevarla a cabo en una disposición de procesadores en serie ya que esto conllevará una saturación en el reloj del procesador además de un uso inapropiado de memoria RAM y promoverá un tiempo de cálculo mucho más extenso [8], por tal motivo se utilizará una disposición de procesadores en paralelo en una partición de método simple en coordinación 2 2 4 lo que significa que la carga se dividirá partiendo el dominio computacional en 2 partes en el eje x, 2 partes en el eje y 4 partes en el eje z dividiendo el dominio para ser calculado por los 16 procesadores en paralelo, este trabajo extra se lo utiliza para sacarle el máximo provecho al recurso computacional que se tiene a disposición.

La computadora a en la cual se está llevando a cabo este estudio es una Asus TUF B460-PRO con procesador Intel i9-10850K con una velocidad de reloj de 3.6Ghz (4.75Ghz Turbo). 32Gb Memoria RAM DDR4 funcionando en *dual channel*, 4TB de

almacenamiento mecánico a 7200RPM trabajando en configuración RAID0, 500GB de almacenamiento en estado sólido dedicados para el sistema operativo y una GPU Nvidia 1660 Super de 6GB.

El procesador en uso es tiene 10 procesadores físicos y 20 procesadores lógicos, sin embargo, se utilizará únicamente 16 procesadores para no saturarlo. El sistema operativo que se empleó para desarrollar este estudio es Window 10 Pro (21H1) con un subsistema de Linux trabajando en WSL 2 con Ubuntu 18.04 LTS.

Los programas utilizados para preproceso son ICEM 18.0 para el desarrollo y modificación de los archivos de mallado estructurado, Blender 2.93 para edición y tratamiento de superficies de contacto. El software de simulación es OpenFOAM 5.x para resolver la etapa CFD, LIGGGHTS 3.8 para resolver el método DEM y CFDEM 5.x para acoplar los métodos CFD y DEM. Los programas utilizados para post proceso son GNUPLOT para graficar los resultados obtenidos de las simulaciones y controlar valores durante el proceso de cálculo, Paraview 5.10.0 para obtener imágenes de los campos de velocidad, presión y poder visualizar el comportamiento de la fase particular, y por último DaVinci Resolve 16 para generar clips de video con la información tratada de paraview y verificar el correcto avance de la simulación en estado transitorio.

3 RESULTADOS Y DISCUSIÓN

3.1 Independencia de malla.

En general, la precisión de los resultados depende en un alto porcentaje del dominio computacional, usualmente con una densidad mayor de elementos se obtiene resultados más precisos, sin embargo, esto afecta proporcionalmente al tiempo de cálculo en una simulación numérica, por tal motivo es importante determinar en qué punto la densidad de elementos de la malla deja de afectar a los resultados de la simulación para optimizar los recursos computacionales al máximo obteniendo buenos resultados.

Dada la complejidad del dominio computacional empleado para este estudio es necesario demostrar que la optimización realizada no va a generar resultados alejados de los valores experimentales y de resultados anteriores. Velasco demostró que el dominio computacional producto de la técnica de mallado empleada en su caso de análisis es independiente de la densidad de elementos [7]. Usando esa información se va a desarrollar 4 mallas con distintas densidades de elementos, partiendo de una malla muy fina empleada por Velasco hasta llegar a una malla optimizada.

Esto se va a determinar al calcular el torque ejercido sobre los alabes móviles de la turbina en el punto en que los valores de cálculo lleguen a la estabilidad utilizando el algoritmo SIMPLE. Estos datos permitirán determinar si las predicciones realizadas mediante simulación numérica son independientes del número de elementos de la malla empleada.

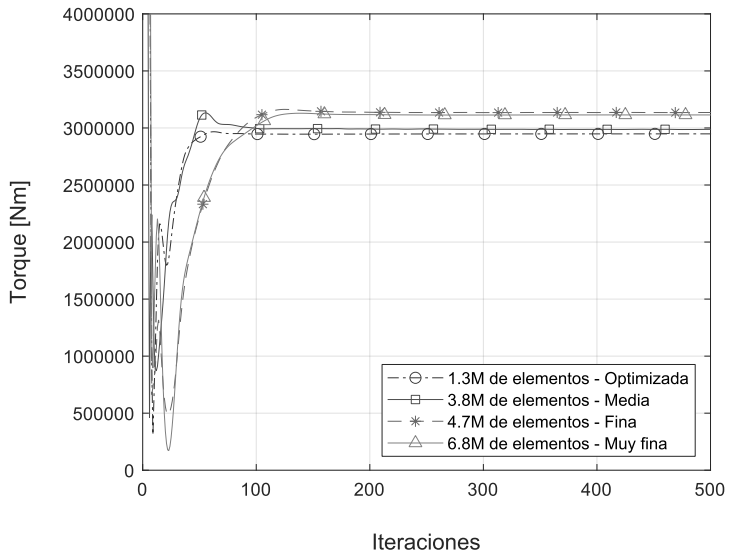

Figura 3.1 Valores de torque en distintas mallas.
(Fuente: Propia)

Como se puede apreciar en la Figura 3.1, todos los casos alcanzan un valor de estabilidad, obteniendo los siguientes resultados:

Tabla 3.1 Valores de estudio para independencia de malla.

Modelo	Número de elementos	Torque [Nm]	Potencia [MW]	Eficiencia	Error [%]	Tiempo de ejecución [h]
Experimental	-	-	109,50	0,94	0,0%	-
Optimizado	1.311.097	2.948.600	103,20	0,88	5,8%	04h31
Media 2	3.774.986	2.986.000	104,51	0,90	4,6%	07h10
Fina	4.648.862	3.135.300	107,45	0,92	1,9%	11h29
Muy fina	6.795.222	3.114.500	106,74	0,91	2,5%	19h01

(Fuente: Propia)

Los resultados destacan que la cantidad de error inferior al 6% lo cual es aceptable en todos los casos, la complejidad geométrica en los cuatro modelos es la misma, sin embargo, el número de elementos que contiene cada dominio computacional difiere al menos con un millón de lo cual representará amplias diferencias en el costo computacional lo cual se puede constatar en el tiempo de ejecución de cada caso.

A continuación, se presenta la Figura 3.2 que muestra el desarrollo de los valores residuales de presión obtenidos para cada malla. Los residuales son las diferencias en el valor de una cantidad (como velocidad, presión, k, etc.) entre dos iteraciones. Esto significa que cuanto más bajo sea el residuo, menos cambiarán los resultados si continúa iterando. Se toma en este caso los residuales del cálculo de la presión ya que estos presentan una naturaleza más caótica y les toma más tiempo en llegar a la estabilidad.

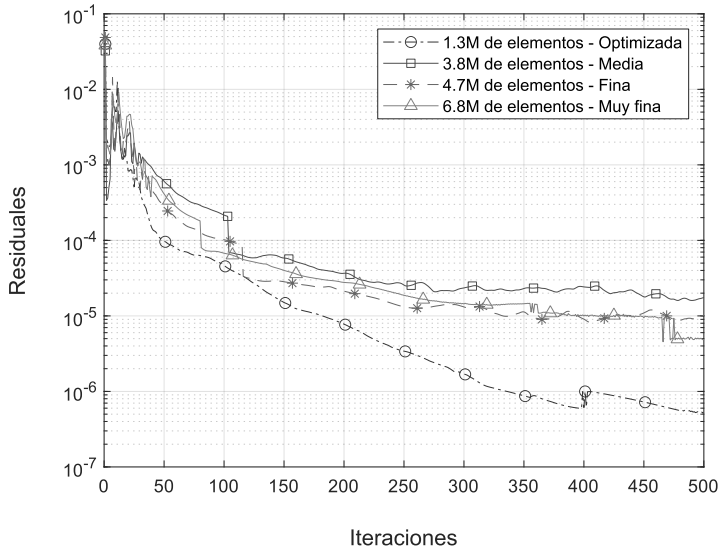

Figura 3.2 Residuales de presión de los casos de independencia de malla.

(Fuente: Propia)

Analizando la naturaleza de las gráficas, podemos notar que el caso de la malla optimizada muestra una tendencia convergente hacia valores mucho más pequeños. Es por este motivo que se tomará este domino computacional para continuar con el estudio de simulación numérica propuesto. Esta ganancia de estabilidad es producto del trabajo de remallado realizado, en donde se acomodó los refinamientos excesivos en las zonas próximas a las superficies limítrofes y en zonas en donde se asume que el flujo ya se encuentra totalmente desarrollado. Logrando reducir un 72% de elementos al caso realizado por Velasco en el 2020.

3.2 Calidad de malla

La calidad de malla es un parámetro de interés y se puede evaluar con el aplicativo *checkMesh* de OpenFOAM, la respuesta de ejecutar este comando muestra los valores muestra los valores destacados en la Tabla 2.2 esta información permitió controlar las modificaciones de la malla efectuadas en el proceso de optimización, ya que se corría el riesgo afectar a los parámetros de calidad obtenidos en investigaciones anteriores.

Así mismo se puede evaluar la calidad de la malla en base a valores adimensionales como el valor Ω que representa la relación entre el número de elementos tridimensionales de una malla y el número de nodos que existe entre ellos, con el fin de estimar implícitamente el recurso computacional necesario para efectuar el cálculo. Este valor es destacable ya que a medida que el valor se aleja de 1 el costo computacional también se incrementa.

Tabla 3.2 Valor Ω para distintos casos de simulación.

Número de Elementos [NE]	Numero de Nodos [NN]	Ω [NE/NN]
1.311.097	1.236.537	1,06
3.774.986	3.981.484	0,95 [7]
2.926.033	2.727.438	1.07 [53]
68.456.549	13.120.572	5,22 [5]

(Fuente: Propia)

En la Tabla 3.2 se presenta el valor de omega para los diferentes casos que han estudiado esta geometría, como se puede ver, el caso estudiado por Mora [5] el valor Ω es cinco, lo que revela la malla utilizada para dicho estudio tiene una calidad baja y un gasto alto uso de recursos en comparación a los resultados obtenidos por Velasco [7] y Quishpe & Guascal [53] que estudiaron más a fondo en el tema de discretización espacial con mallado estructurado y parámetros de optimización, utilizando de manera

más apropiada los recursos computacionales a disposición y asegurando datos más veraces.

Para la malla utilizada en este caso de estudio se presenta un valor de 1.06 mismo que se acerca mucho al valor deseado y anticipa un equilibrio entre el tamaño de la malla y los recursos computacionales necesarios para su correcto estudio.

3.3 Residuales

Se ha detallado en incisos anteriores que los residuales no son soluciones, pero ayudan a determinar la estabilidad de los datos obtenidos y su nivel de convergencia hacia una solución, no necesariamente la correcta, sin embargo, estudiarlos es importante pues ayudan a encontrar el nivel de error que tienen los resultados de la simulación.

A continuación, se muestra tres graficas de residuales pertenecientes a las simulaciones para algoritmo SIMPLE, PISO y CFDEM respectivamente.

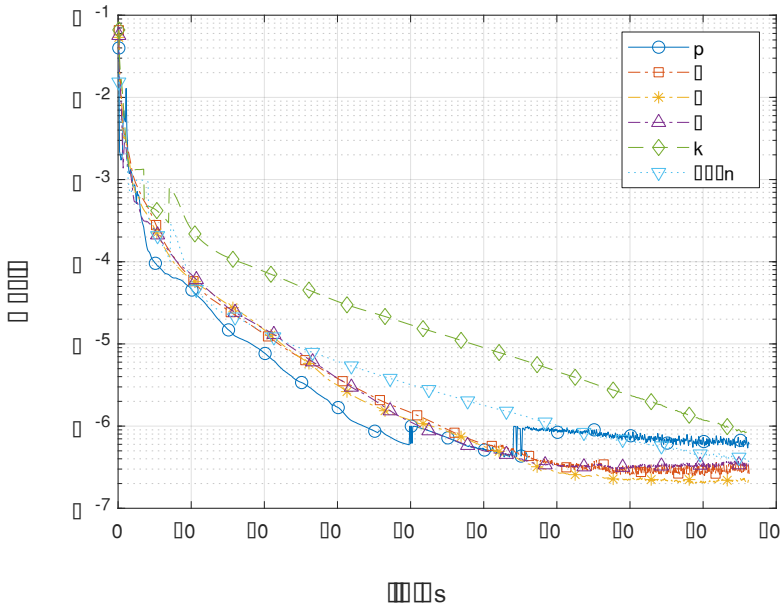

Figura 3.3 Residuales de la simulación numérica de algoritmo SIMPLE.
(Fuente: Propia)

En la Figura 3.3 se presenta el desarrollo de los residuales del cálculo de la primera etapa de simulación. Para esta etapa se utilizó el aplicativo interno de OpenFOAM llamado *simpleFoam* en conjunto con la malla optimizada de 1.3 millones de elementos.

Como se puede constatar, los residuos del proceso iterativo tienen una naturaleza convergente hasta diferencias que se mantienen en el orden de 1×10^{-6} hasta 1×10^{-7} alcanzando este valor en todas las variables en 865 iteraciones. Lo cual indica que el proceso se ha estabilizado y a partir de ese momento en adelante no van a existir

diferencias notables en los resultados, por tal motivo no es necesario continuar con la simulación ya que el proceso ya ha convergido de manera exitosa.

Como se menciona anteriormente, la siguiente etapa del proceso de cálculo consta de dos simulaciones que basan su funcionamiento en el algoritmo PISO, para analizar el fenómeno en estado transitorio y material particular.

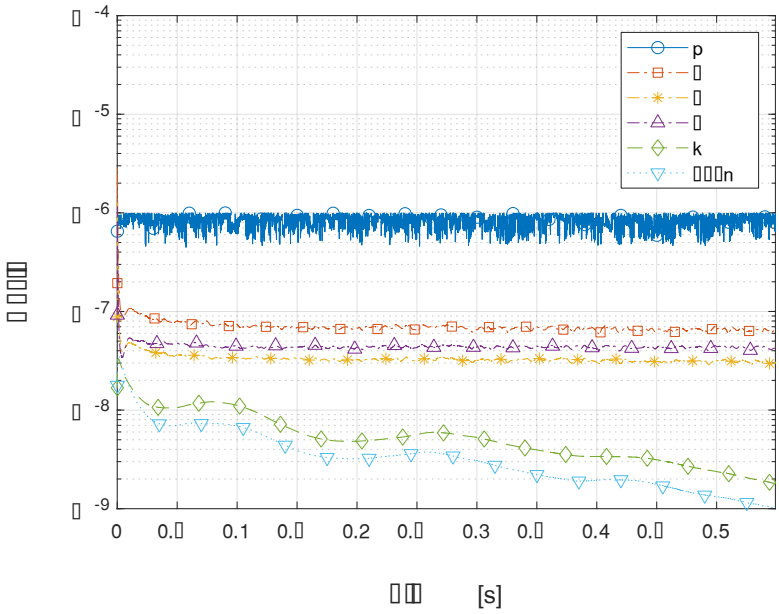

Figura 3.4 Residuales de la simulación numérica de algoritmo PISO.

La Figura 3.4 muestra los residuales de la simulación llevada a cabo con el aplicativo de *pisoFoam*. Aquí se puede ver que el cálculo se da lugar en un instante de tiempo y no de iteraciones, el lapso utilizado en este caso va desde 0 a 0.549 segundos que es el equivalente a 3 vueltas del rotor en las condiciones de borde utilizadas para el cálculo.

Los residuos para este caso son variados, pero muestran una clara uniformidad y estabilidad después de los 0.2 segundos. La presión es primera variable para destacar ya que tiene los valores residuales más amplios y una naturaleza muy caótica sin embargo sus valores son aceptables cercanos e inferiores al 1×10^{-6}.

La velocidad en sus tres componentes cuenta con una naturaleza parecida entre sí, manteniéndose en un rango de valores inferior a 1×10^{-7}, siendo estables a lo largo de toda la simulacion.

Por otro lado, los valores residuales de k y épsilon tienden a disminuir a medida que la simulación avanza manteniéndose en un rango inferior al 1×10^{-8} en el tiempo 0.2 s consiguiendo residuales más pequeños a medida que la simulación avanza.

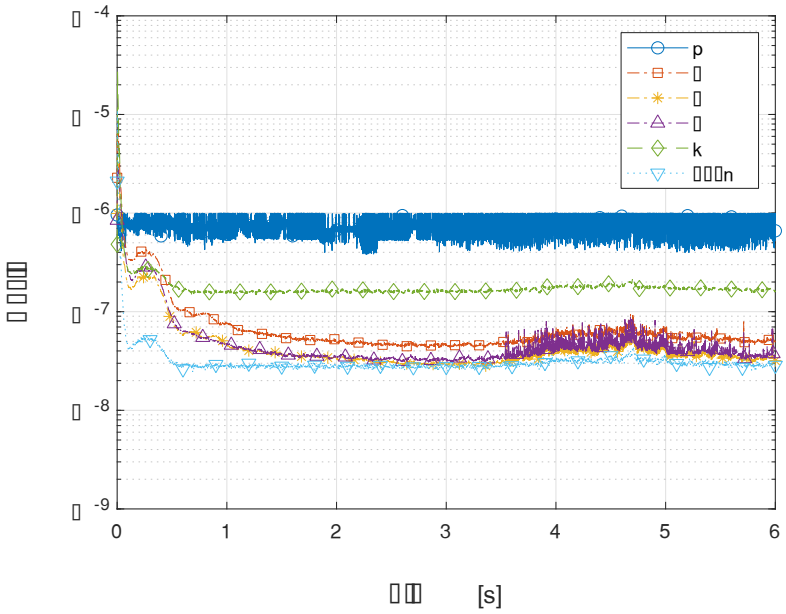

Figura 3.5 Residuales de la simulación numérica de algoritmo CFD-DEM.

Para representar los residuales del caso de simulación CFD-DEM presentamos la Figura 3.5, en este caso particular se extendió el tiempo de cálculo hasta 7 segundos ya que en este lapso se puede estudiar el ingreso de las partículas al sistema por la entrada en la caracola hasta que empiezan a salir de la turbina por la salida del tubo de descarga, habiendo pasado por todas las zonas de interés.

Aquí se evidencia una naturaleza diferente a los anteriores casos, con un valor de residuo mayor que ronda entre 1×10^{-6} hasta 1×10^{-8} este aumento de la diferencia en los residuos se puede deber a la presencia de las partículas en el cálculo. Podemos ver una presión caótica pero estable en un valor aceptable junto a las variables de las componentes de velocidad, épsilon y k que por otro lado mantienen un más pequeño rondando el 1×10^{-7} en todo el tiempo de estudio, pero presentando lapsos caóticos desde el segundo 3.5 en adelante. Esta alteración en los residuales puede deberse a la inclusión del material particular en el dominio, ya que a medida que este material avanza por el sistema debe interactuar con el resto de las variables.

3.4 Simulación con método SIMPLE

Hasta este punto se ha logrado comprobar que las simulaciones resueltas con la malla optimizada tienen valores estables y son resueltas correctamente, ahora es preciso evaluar los resultados y determinar que los valores obtenidos en los campos de velocidad y presión sean correctos y representen de manera adecuada el fenómeno físico, comparando los resultados obtenidos en el nuevo dominio computacional con la información recabada en estudios anteriores.

En la Figura 3.6 se muestran los perfiles de velocidad obtenidos a la salida del rodete, en la intersección del *runner* y el *draft tube* calculados con la malla optimizada y una malla fina [7]. Aquí se puede apreciar que los campos de velocidad están siendo visualizados en el mismo rango de valores, destacando que los dos gráficos presentan tendencias similares en la formación del vórtice. Es también destacable la velocidad máxima de 30 [m/s], esta velocidad es la alcanzada por el fluido al pasar por el runner y es visible en las zonas más próximas al runner. La principal diferencia se nota en la formación de los campos de baja velocidad y se debe a la reducida cantidad de elementos presentes en esta zona, sin embargo, los valores obtenidos son los mismos.

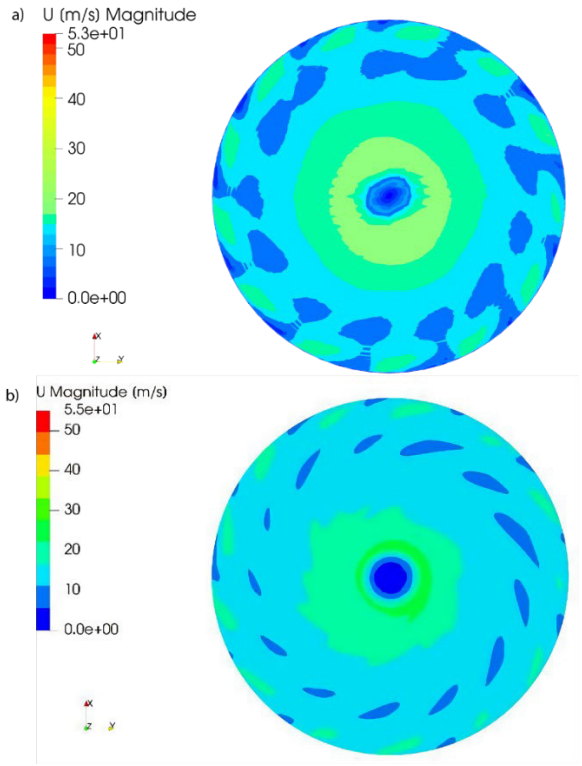

Figura 3.6 Perfiles de velocidad en salida del runner: a) Malla optimizada b) Malla Fina [7]. (Fuente: Propia)

La Figura 3.7 Muestra un corte en el plano XY de toda la turbina, en donde podemos ver que los resultados obtenidos en la malla optimizada muestran las mismas tendencias

que la malla fina y los valores obtenidos son los mismos en todo el dominio. Las principales diferencias se presentan en las zonas más próximas a las superficies ya que al no existir un alto refinamiento no se puede apreciar con detalle la condición de no deslizamiento debido a la fricción. También se puede notar que la zona de disipación del vórtice es más extensa que en la malla fina, sin embargo, esto no afecta en gran medida al estudio de flujo particular.

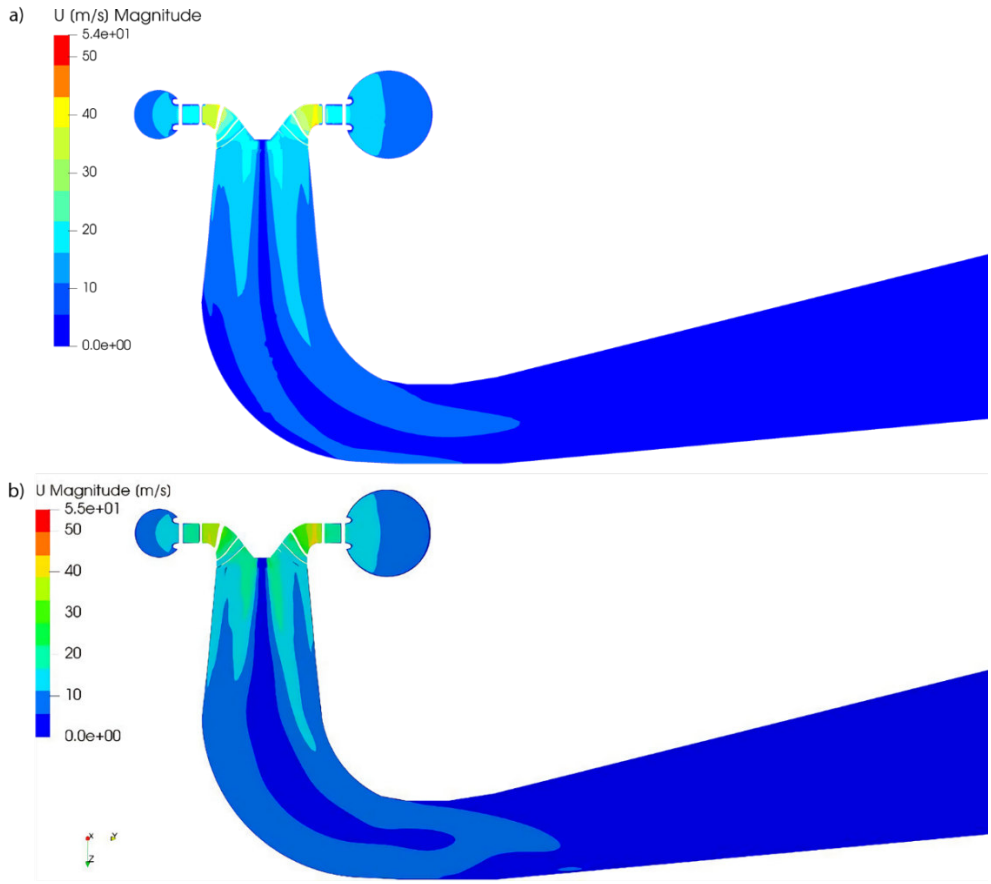

Figura 3.7 Perfiles de velocidad en el plano XY: a) Malla optimizada b) Malla fina [7]. (Fuente: Propia)

En la Figura 3.8 y Figura 3.9 se muestran los perfiles de presión en el conjunto de los alabes directrices con el rodete vistos desde la vista superior, se ve también el rodete visto con todos sus alabes desde un punto de vista no paralelo a un plano de referencia, para que se puedan ver la distribución de presión en todo el sistema de hidro alabes móviles y en último lugar se presenta un solo alabe móvil. Estas vistas ayudan en la comparativa entre la mallas optimizada y fina para determinar el correcto desarrollo de esta variable visto en resultado anteriores.

Es fácil notar que en ambos casos existe el mismo rango de presiones y que estas se ubican en los mismos lugares con los mismos valores las pocas diferencias que se

encuentran son los puntos en donde empieza y termina un contorno. Los resultados son los esperados en todo el dominio del rodete, encontrando una zona de alta presión en la zona de los alabes directrices y disminuyendo poco hasta encontrar una presión negativa en la salida del rodete.

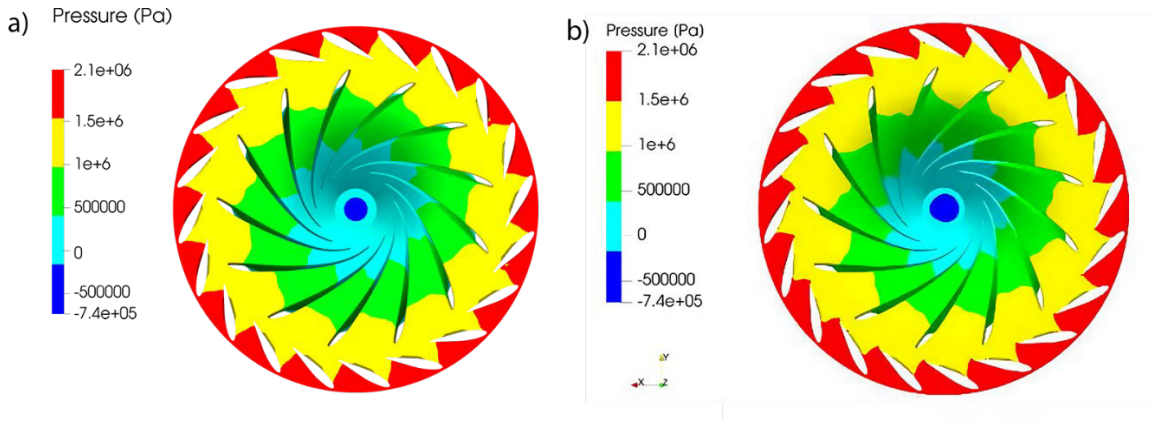

Figura 3.8 Perfiles de presión vista superior del rodete y alabes fijos: a) Malla optimizada b) Malla fina [7].

(Fuente: Propia)

En la Figura 3.9 se muestra solamente al rodete y sus alabes, allí podemos apreciar que los valores están en el mismo rango en toda la superficie del rodete, pero vemos una diferencia notoria en la extensión de los campos de baja, media y alta presión, teniendo en la malla optimizada zonas un poco más cortas que en los resultados anteriores. Esto no es problema ya que este resultado estaba contemplado, dada la naturaleza caótica de cálculo de esta variable, sin embargo, se obtuvieron resultados consistentes con estudios anteriores.

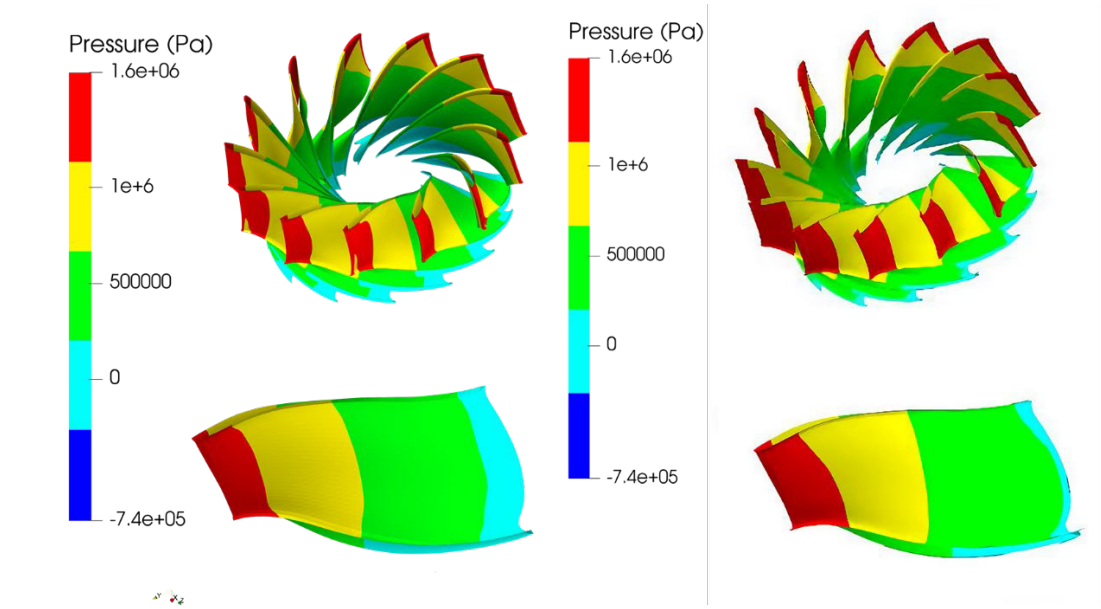

Figura 3.9 Perfiles de presión para el rodete y uno de sus alabes: a) Malla optimizada b) Malla fina [7].

(Fuente: Propia)

La Figura 3.10 muestra un corte en el plano XZ de los resultados obtenidos y los resultados de referencia. En esta imagen podemos ver que de igual manera los resultados son los esperados y son consistentes con los estudios realizados por Mora [5] y Velasco [7].

La caracola mantiene la zona de presión más alta y disminuye a medida que el fluido recorre el sistema de la turbina Francis, de igual modo el tubo de descarga mantiene una presión homogénea en toda su extensión y a la salida del rodete se forma la zona de más baja presión. Existe una diferencia en el tamaño de la zona del vórtice, sin embargo, esto se debe a que el refinamiento en esta zona es mucho más amplio para la malla optimizada por lo que no hay suficientes celdas para mostrar la forma del vórtice en su totalidad, sin embargo, la forma del vórtice obtenido no difiere en gran medida y no causarán problemas de cálculo.

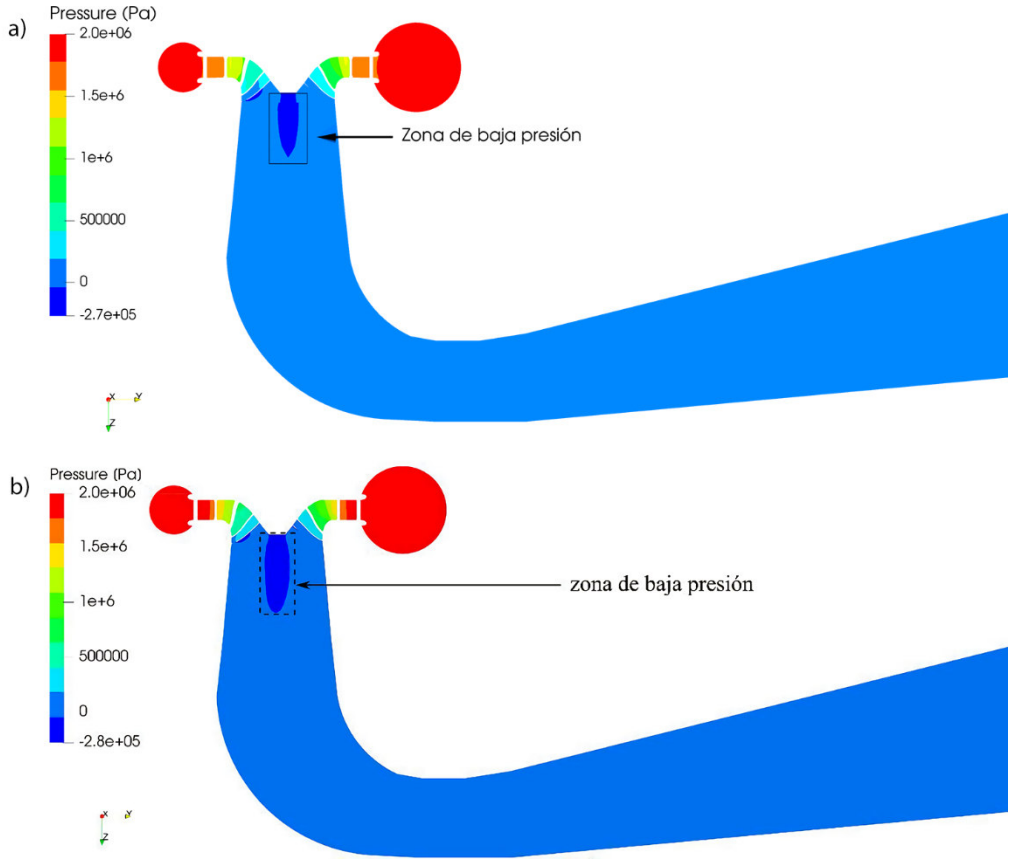

Figura 3.10 Perfiles de presión en el plano YZ: a) Malla optimizada b) Malla fina [7]. (Fuente: Propia)

3.5 Simulación método PISO

Esta simulación es parte de la segunda etapa del proceso para el desarrollo de este estudio, esta etapa calcula la naturaleza del fluido previo a acoplarlo con el método DEM, aquí se calibró el número de Courant para asegurar una buena calidad de resultados. A partir de aquí trabajaremos únicamente con la malla optimizada, ya que el seguir estudiando en paralelo las variaciones de los resultados en cada dominio computacional generado esta fuera del alcance de este estudio y se ha demostrado ya que los valores obtenidos son independientes del dominio computacional.

Este cálculo introduce un nuevo valor que es el *timestep* o intervalo de tiempo, ya que este análisis se efectúa en un periodo de tiempo mesurable, este término no se eligió de manera aleatoria, sino cuidando que el número de Courant no sea superior a 1. En la ecuación (2.6) se puede ver la definición de este parámetro adimensional y como relaciona la variable de velocidad, intervalo de tiempo y dimensiones de celda.

Para determinar este valor en nuestro caso se utilizó las funciones internas de OpenFOAM para que sea calculado en simultaneo con el resto de las variables para todas las celdas de la malla. Este valor adimensional se calcula en cada intervalo de tiempo por lo cual se presentará el desarrollo de valor Courant máximo y el promedio de todo el dominio durante el lapso de estudio que son 0.549 [s] o 3 vueltas del rodete en las Figura 3.11 y Figura 3.12. este valor se lo toma de las conclusiones dadas por Velasco en su estudio, en donde considera este valor como mínimo para poder reproducir correctamente el fenómeno en estado transitorio.

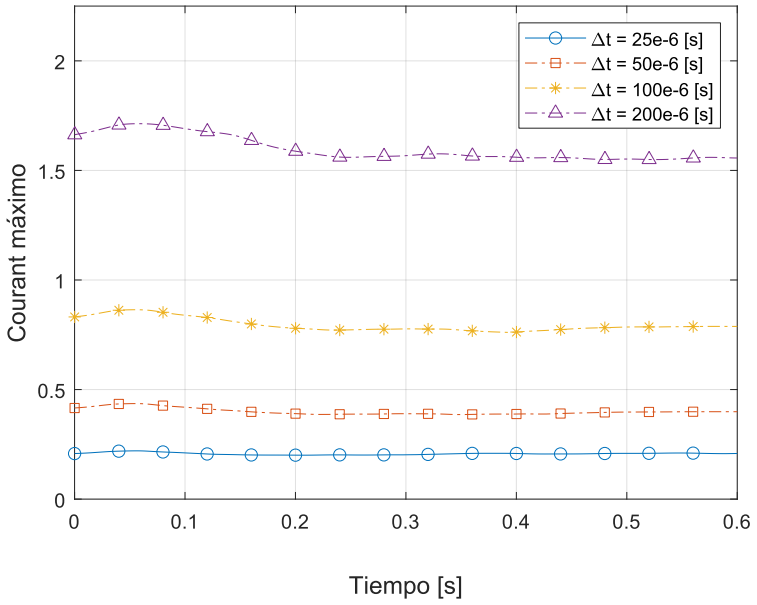

Figura 3.11 Courant máximo en cada intervalo de tiempo simulado.
(Fuente: propia)

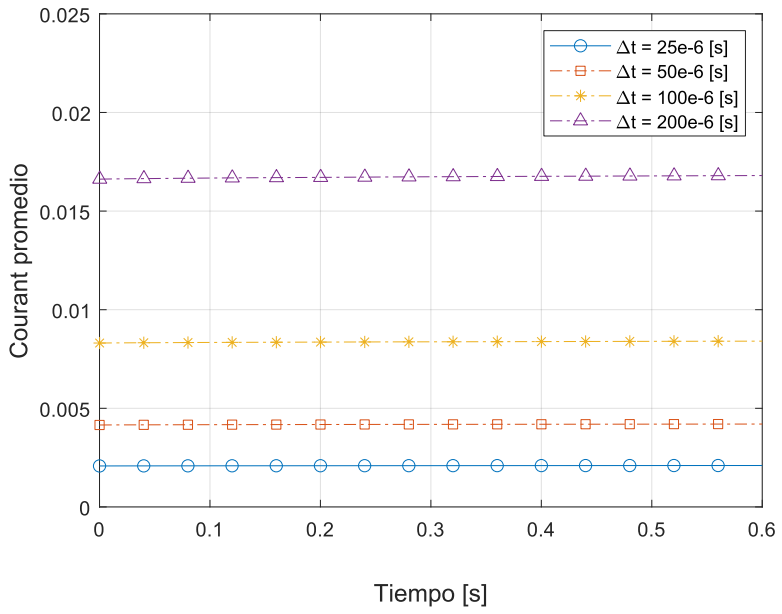

Figura 3.12 Courant promedio en cada intervalo de tiempo simulado.
(Fuente: propia)

Para desarrollo de esta etapa se tomó 4 intervalos de tiempo, que se pueden ver en ambas figuras, el primer grafico muestra los valores máximos de Courant para cada simulación y el segundo el valor promedio en todo el dominio. En ambos gráficos el Courant aumenta proporcionalmente al intervalo de tiempo.

En la primera figura se puede ver que en un intervalo de $25 \times 10^{-6}\ [s]$ este valor permanece estable en 0.2 a lo largo de toda la simulación, y a medida que aumenta el intervalo de tiempo (Δt) aumenta la inestabilidad como se puede apreciar en el intervalo de $200 \times 10^{-6}\ [s]$ que no muestra tendencia hacia ningún valor. Por otro lado, podemos ver que el intervalo más amplio generará los resultados de más baja calidad ya que todos sus valores exceden el valor admisible, el resto de los valores se mantienen con un valor por debajo de la unidad lo que indica que los resultados representarán el fenómeno de manera adecuada.

En la segunda figura podemos ver que todos los valores se mantienen dentro de una estabilidad y en valores muy inferiores al valor admisible, también se puede ver que el Courant promedio también aumenta proporcionalmente al intervalo de tiempo (Δt).

Con los resultados mostrados en las gráficas se puede asumir que cualquier intervalo de tiempo que sea inferior o igual $100 \times 10^{-6}\ [s]$ generaría resultados válidos, por lo cual es importante destacar el “tiempo de ejecución” de simulación en horas es un parámetro de suma importancia y se lo muestra en la figura a continuación.

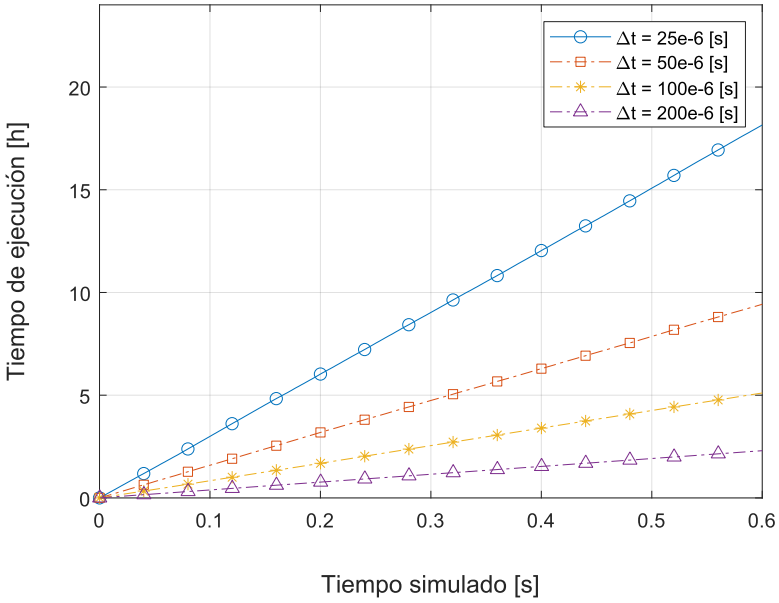

Figura 3.13 Tiempo de ejecución vs tiempo simulado.
(Fuente: propia)

Aquí es importante destacar que el tiempo de ejecución es inversamente proporcional al intervalo de tiempo (Δt), pues mientras mas pequeño se vuelve, el tiempo de ejecución aumenta, para este caso podemos apreciar que se demora 18.5 horas en concluir el lapso deseado con un intervalo de $25 \times 10^{-6}\ [s]$ y por otro lado vemos que demora 2.5 horas en concluir si el intervalo aumenta a $200 \times 10^{-6}\ [s]$.

En resumen, se debe elegir el intervalo de tiempo que produzca el menor número de Courant y ocupe óptimamente el poder computacional a disposición, ya que al momento de incluir la fase solida particular este tiempo de ejecución aumentará drásticamente llevando a cabo en simultaneo el cálculo CFD, DEM y el acoplamiento.

Esto permitió elegir al intervalo (Δt) $100 \times 10^{-6}\ [s]$ ya que mantienen el Courant estable e inferior al valor admisible y su tiempo de ejecución es 3.5 veces inferior al intervalo (Δt) $25 \times 10^{-6}\ [s]$, a continuación, se muestra la distribución de Courant de este caso para su dominio general y en las zonas más sensibles.

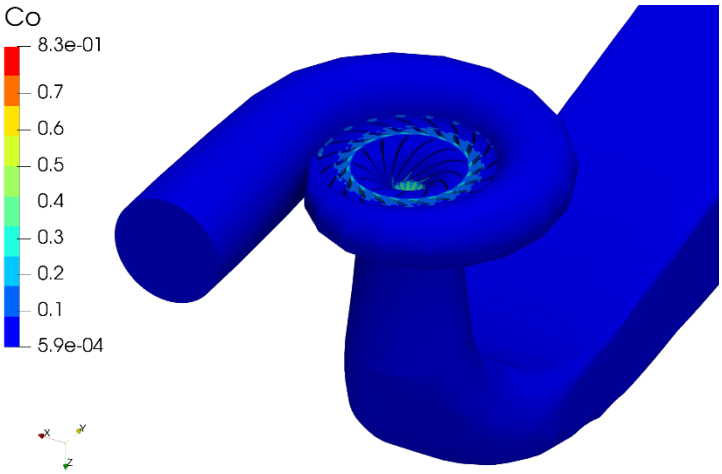

Figura 3.14 Distribución de Courant para el intervalo 100e-6.
(Fuente: propia)

En la Figura 3.14 se presenta la distribución del Courant en todo el dominio para poder identificar fácilmente las zonas de interés. Se puede apreciar que existe un aumento en las zonas donde la velocidad aumenta (alabes directores, rodete). En el resto del dominio se puede ver que el Courant se mantiene en el valor promedio de todo el dominio.

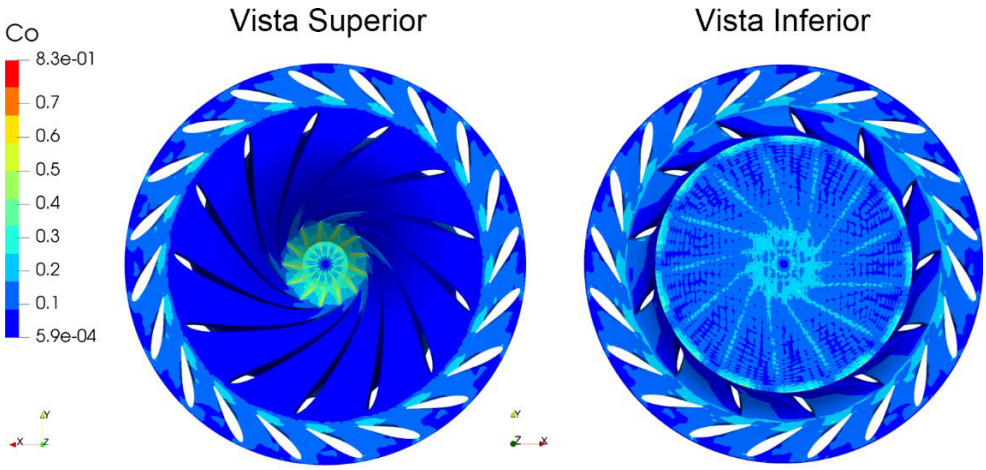

Figura 3.15 Distribución de Courant en el conjunto Rodete-Alabes directores para el intervalo 100e-6.
(Fuente: propia)

En la Figura 3.15 se puede apreciar que el valor de Courant es mayor a medida que se acerca a las zonas de vórtice a la salida del rodete, esto se debe a la topología de mallado, puesto que esta fue una de las zonas en donde se redujeron elementos para optimizar la malla, por tal motivo las celdas tienen mayor tamaño, manteniendo el valor adimensional por debajo del valor admisible en todo este conjunto.

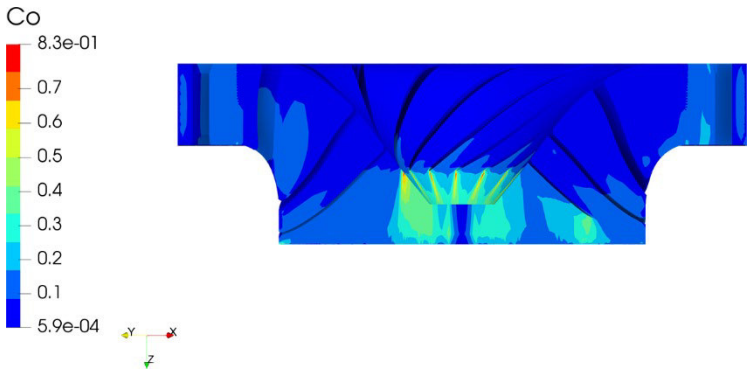

Figura 3.16 Distribución de Courant en corte del conjunto Rodete-Alabes directores para el intervalo 100e-6.

(Fuente: propia)

En la Figura 3.16 se puede apreciar que el Courant es mayor en la zona del vórtice especialmente en la zona en que el fluido transita de la superficie del rodete hacia la zona de baja presión, presentando el valor máximo de 0.83 en esta zona.

Es importante controlar el Courant en estas zonas ya que el flujo de la fase solida por este punto podría hacer que las partículas abandonen el dominio computacional entre cada iteración si el Courant excede la unidad.

3.6 Acoplamiento CFD-DEM

Esta es la etapa final, una vez optimizada la malla y calibrada la simulación, podemos incluir la fase solida al cálculo con el uso del software CFDEM, que se encarga de calcular las partículas iniciales con el método DEM y hacerlas interactuar con los campos de velocidad y presión que son calculados a medida que la simulación avanza en el lapso de quince segundos, tiempo en el cual la mayoría de las partículas insertadas han atravesado todo el mecanismo de la turbina Francis.

Las primeras partículas son insertadas en la zona del tubo de admisión en la caracola de la turbina Francis, esta distribución se muestra en la Figura 3.17, estas partículas tienen las características mostradas en la Tabla 2.10. Se puede apreciar que las partículas cubren un área cilíndrica al interior de la caracola. Se insertaron seiscientas partículas de seis diámetros diferentes desde los 2.5 hasta los 10 milímetros de manera aleatoria utilizando el método DEM. También se indica al programa las superficies que van a limitar el movimiento de las partículas, mismas que servirán para el cálculo aproximado del desgaste debido a los impactos que reciban durante la simulación.

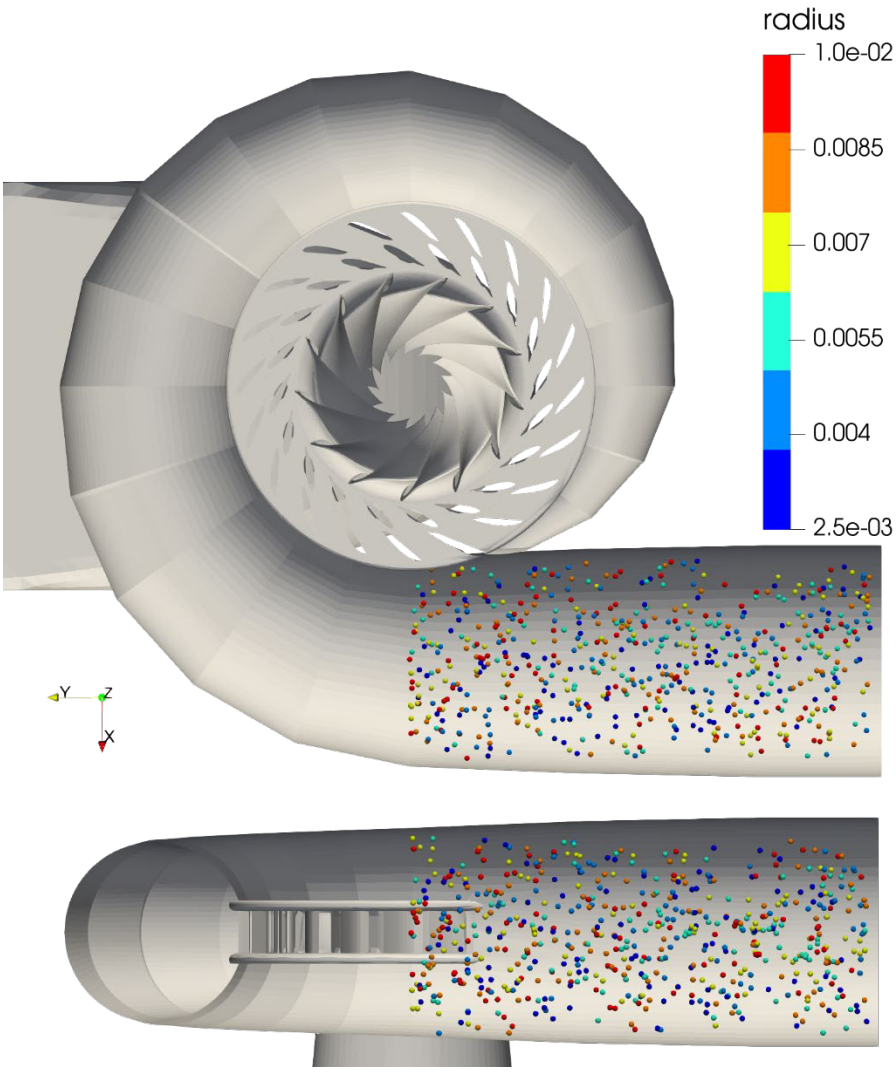

Figura 3.17 Distribución de partículas iniciales en el domino computacional.

Como resultado del acoplamiento se presenta en las figuras 3.18 3.19 y 3.20. Allí se puede ver el desarrollo del material particular dentro del dominio computacional desde diferentes puntos de vista cronológicamente desde 0.5 segundos hasta 12 segundos. Separándolos en 2 partes en donde se puede ver las partículas ingresar al rotor en el lapso desde 0.5 hasta 2.5 segundos, y su movimiento a través del tubo de descarga desde los 2.5 segundos hasta los 12.5 segundos. Se ha diferenciado a las partículas en base a su diámetro con colores para destacar algunos detalles de esta simulación.

Lo primero que se puede apreciar es el efecto de la gravedad sobre las partículas, en la Figura 3.19 y 3.20 inciso a) se puede ver la trayectoria de las partículas desde su creación en el tiempo 0 hasta su ubicación en el suelo de la caracola en el segundo 0.5.

Los efectos de la condición de no deslizamiento se pueden notar ya que a medida que el material particular se acerca a las paredes disminuyen su velocidad y empiezan a ser

arrastradas las partículas que tienen mayor diámetro, siendo estas las primeras que entran al rodete y descender por el tubo de descarga, mientras tanto las partículas más pequeñas se retrasan y disminuyen su velocidad al ser afectadas más por la condición fricción en las paredes.

Se puede ver también que las partículas con mayor masa son las primeras en ser arrastradas por el fluido a lo largo del mecanismo, mientras que las partículas con menor masa tienden a retrasarse y quedarse girando junto con el rodete, estancándose en las zonas de menor presión de los alabes directrices, como se puede ver en la figura 3.18 y 3.20 inciso h). Esto se debe a que las partículas con más masa tienen más inercia y mantienen su movimiento durante un periodo más extenso siendo las primeras que ingresan al rodete y por acción de la gravedad descienden hasta el tubo de descarga. Mientras que las partículas pequeñas con menos inercia se ven afectadas aún más por las velocidades reducidas del fluido circundante a las paredes.

En la figura 3.19 y 3.20 a partir del inciso e), se ve que las partículas salen aceleradas del rodete formando trayectorias caóticas pero manteniendo la tendencia de el vórtice con forma de espiral sobre las paredes del tubo de descarga, disminuyendo su velocidad a medida que llegan a las zonas de más baja presión, llegando a estancarse en el suelo del *draft tube*, lugar en el que el fluido reduce su velocidad por lo tanto ya no arrastra a las partículas lo cual podría generar una zona con retención de sedimentos a la salida de la turbina.

Las partículas más pequeñas se quedan rotando junto al rodete como se puede ver en la figura 3.18 y 3.20 inciso g) y h), generando zonas de impacto frecuente pudiendo generar desgaste en la parte posterior de los alabes de directores y en la zona frontal de los alabes del rotor.

Otra zona en donde se puede encontrar una alta concentración de choques de partículas de todos los tamaños es en el tramo final de la caracola, en donde todas las partículas cambian de dirección para entrar al área de los álabes fijos y directores. Esta zona se destaca en color rojo en la figura 3.18 inciso d). Este cambio de trayectoria en las partículas indica que esta zona posiblemente contenga una mayor cantidad de desgaste, sin embargo, esto se analizará en el siguiente punto.

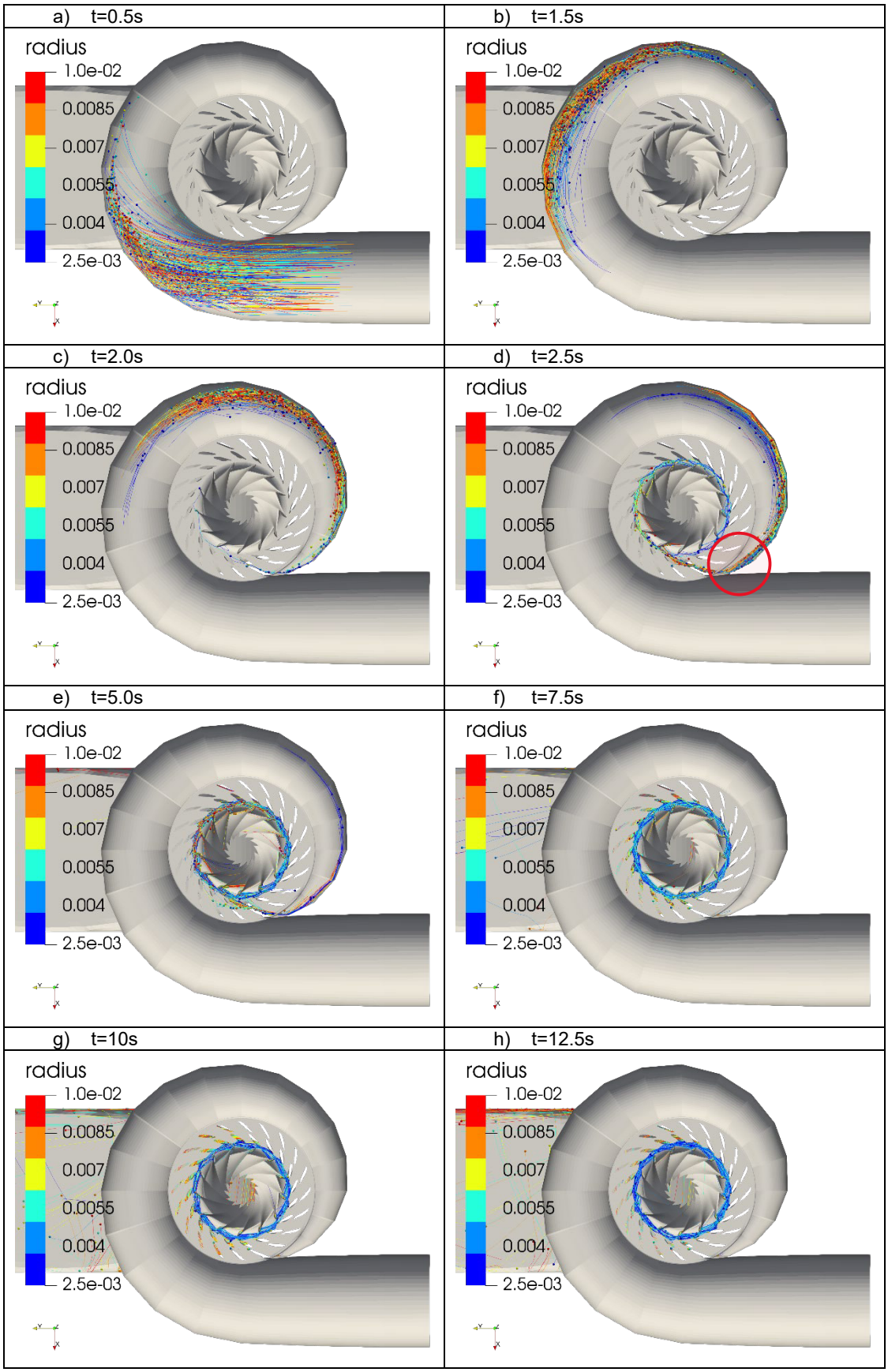

Figura 3.18 Desarrollo de partículas a través del dominio computacional, Vista superior. (Fuente: Propia)

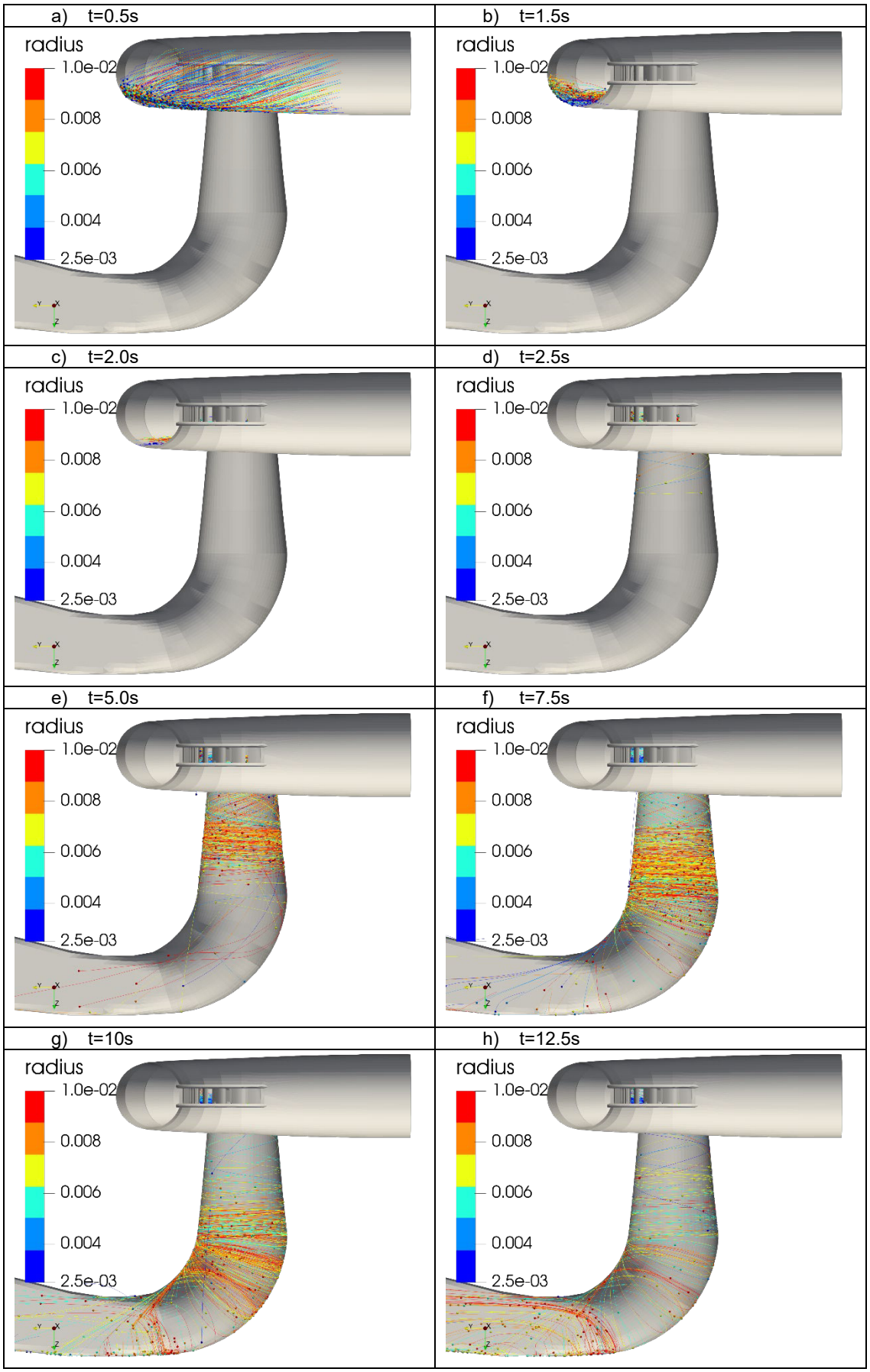

Figura 3.19 Desarrollo de partículas a través del dominio computacional, Vista lateral.

(Fuente: Propia)

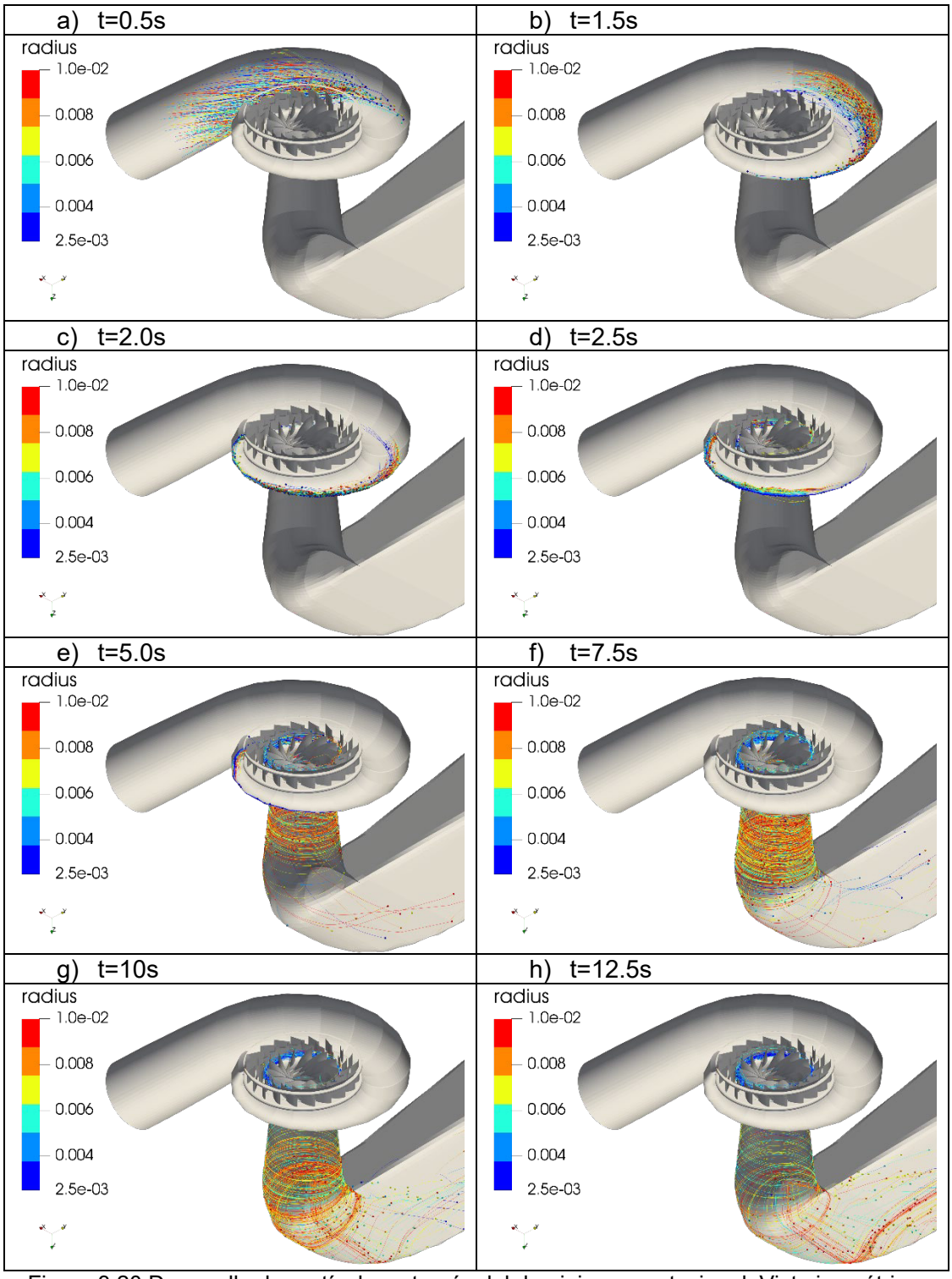

Figura 3.20 Desarrollo de partículas a través del dominio computacional, Vista isométrica. (Fuente: Propia)

El archivo de configuración de LIGGGHTS permite insertar el modelo de desgaste de Finnie [66] para analizar un perdida de material desde un punto de vista cualitativo y obtener zonas de posible daño debido a material particular sobre las superficies que limitan el dominio computacional.

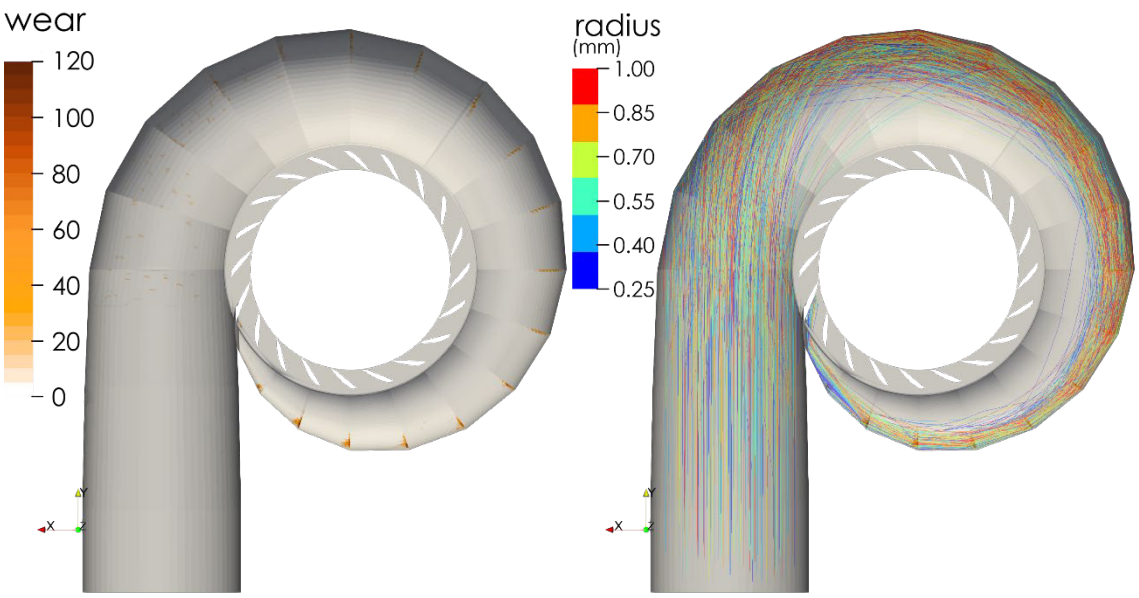

Figura 3.21 Zonas de desgaste por material particulado en la caracola.
(Fuente: Propia)

En la Figura 3.21 se presentan las trayectorias de las partículas en el lado derecho y las zonas que han sufrido un posible mayor desgaste en la izquierda, como se menciona anteriormente el desarrollo del flujo y la geometría de la caracola obliga a las partículas a ubicase en las zonas próximas a las superficies, generando impactos en cada cambio de sección de la caracola, como se puede ver en cada cambio de tramo existe un máximo de desgaste y este aumenta a medida que el radio de la caracola disminuye, sin embargo mantienen valores bajos de desgaste.

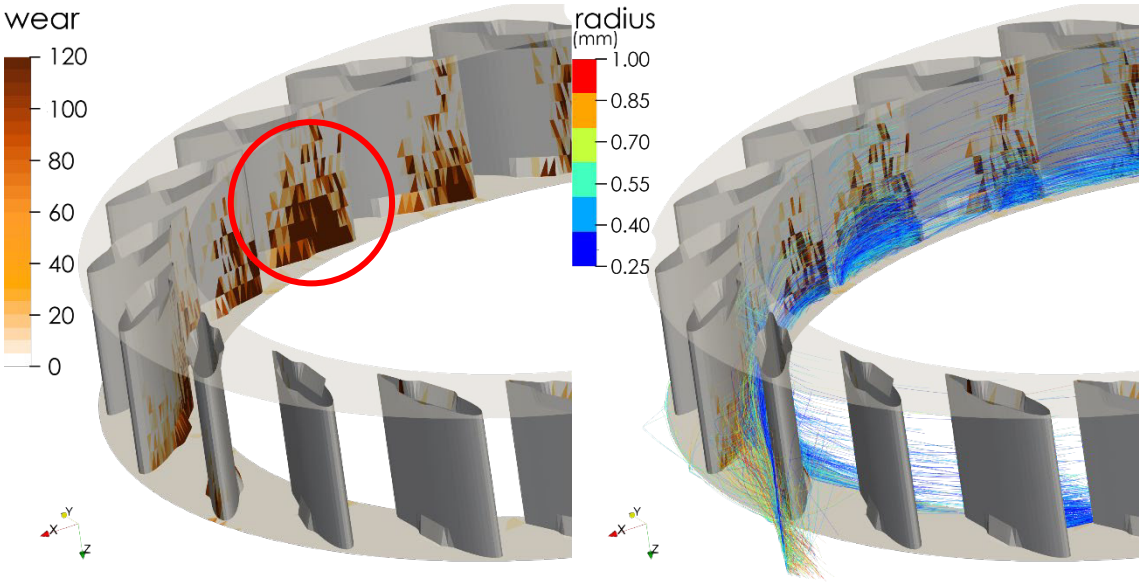

Figura 3.22 Zonas de desgaste por material particulado en los alabes directores.
(Fuente: Propia)

En la Figura 3.22 se ven al lado izquierdo las posibles zonas en el conjunto de los alabes directrices y en el lado derecho las trayectorias de las partículas durante el lapso de la simulación, como se puede ver las partículas de menor tamaño tienen más tiempo de permanencia en la zona entre el rodete y los alabes directores hecho que se puede apreciar también en las trayectorias de la Figura 3.23 lo cual produce más colisiones en el borde de fuga de los alabes directores y en el borde de ataque de los alabes del rodete como se muestra en el círculo rojo de la figura 3.22 y 3.23, también se puede determinar que la mayor cantidad de colisiones se generan en la base de estos elementos debido al asentamiento de las partículas por acción la gravedad, transmitiendo también colisiones a la parte baja del borde de ataque de los alabes del runner.

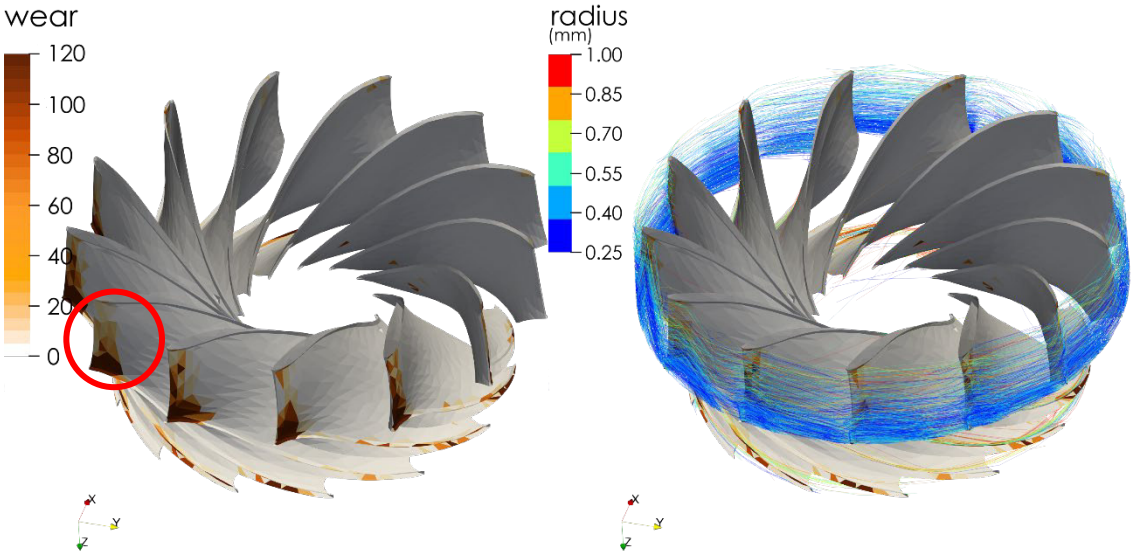

Figura 3.23 Zonas de desgaste por material particulado en el rodete.
(Fuente: Propia)

En la Figura 3.24 se puede ver el estado de uno de los alabes al terminar la simulación desde una vista frontal y posterior. De la primera vista se puede destacar que la zona que más se desgasta es el borde de ataque del alabe, con un ligero incremento en la esquina inferior, se ve que el desgaste se extiende por toda la parte inferior del aspa ya que esa es la ruta que las partículas toman junto con el fluido para descender hasta el tubo de descarga. En el otro lado se observa un incremento de desgaste en la esquina inferior que más se aleja del vórtice, este incremento era esperado debido a la naturaleza en espiral que siguen las partículas que hace que continúen su camino alrededor del vórtice desviándolas hacia las paredes del tubo de descarga encontrando en la trayectoria esta zona del alabe.

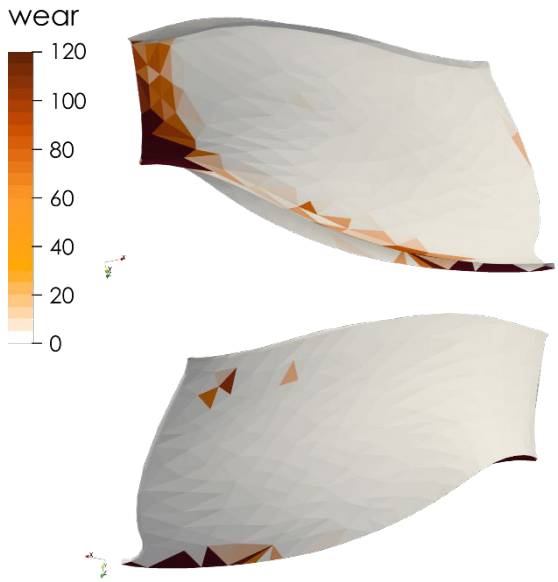

Figura 3.24 Zonas de desgaste por material particulado en los alabes del rodete.
(Fuente: Propia)

Como se explicó anteriormente el método de desgaste utilizado es puramente cualitativo y permite determinar las zonas que pierden material por la interacción con el material particular, permitiendo comparar esta información con la turbina real. Así se puede ver en la Figura 3.25 los resultados de la simulación y el estado real de un alabe directriz de la turbina Francis existente en la central hidroeléctrica San Francisco. Revelando que las predicciones de desgaste tienen se asemejan en gran medida al fenómeno físico, pues se puede apreciar que en la pieza real la pieza ha sufrido múltiples mantenimientos en la misma zona en donde el cálculo CFDEM sugiere que existiría un desgaste por material particular.

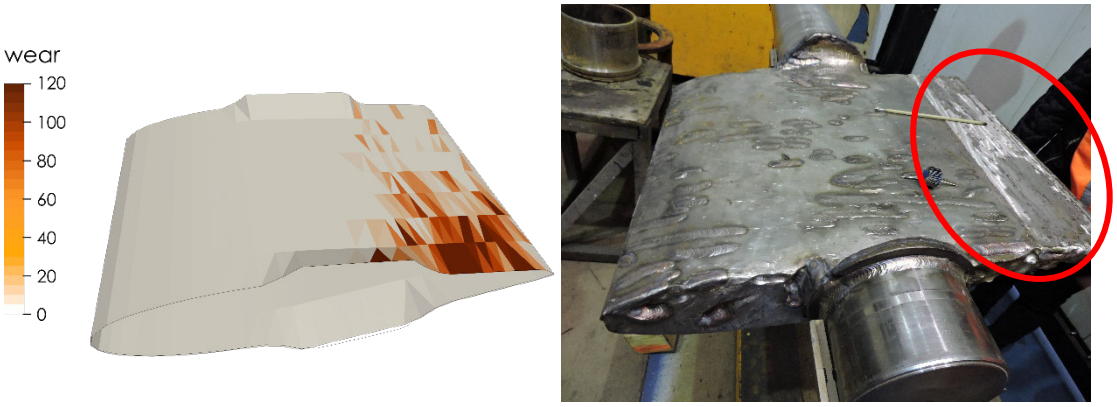

Figura 3.25 Comparación de daños en el material por desgaste en alabes directrices obtenidos mediante simulación numérica.
(Fuente: Propia)

De igual manera en la Figura 3.26 se presenta la salida del rodete en una vista inferior y se destacan las zonas en donde ha existido un mantenimiento y se ha reconstruido con soldadura en los lugares donde el cálculo sugiere que existiría perdida de material por el impacto de partículas en un periodo prolongado de tiempo.

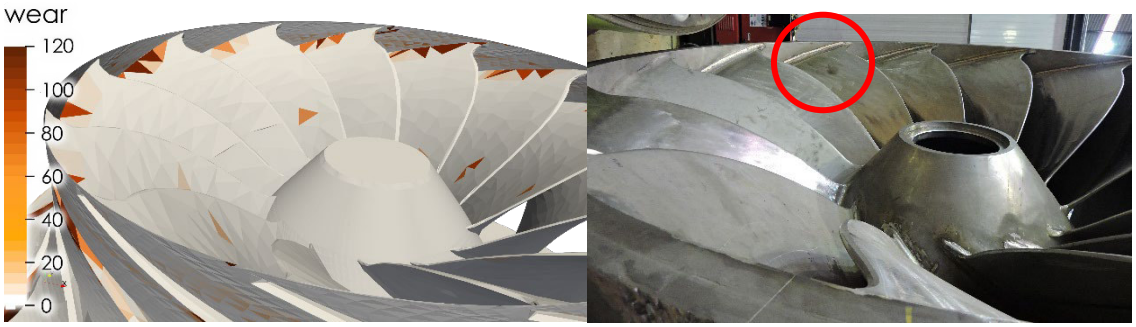

Figura 3.26 Comparación de daños en el material por desgaste en el rodete obtenidos mediante simulación numérica.

(Fuente: Propia)

En la Figura 3.27 se presenta la entrada del rodete y se marca en rojo las zonas que presenta desgaste. Se puede ver que esas mismas zonas en la simulación son las que presentaron mayor cantidad de perdida de material por la interacción con las partículas.

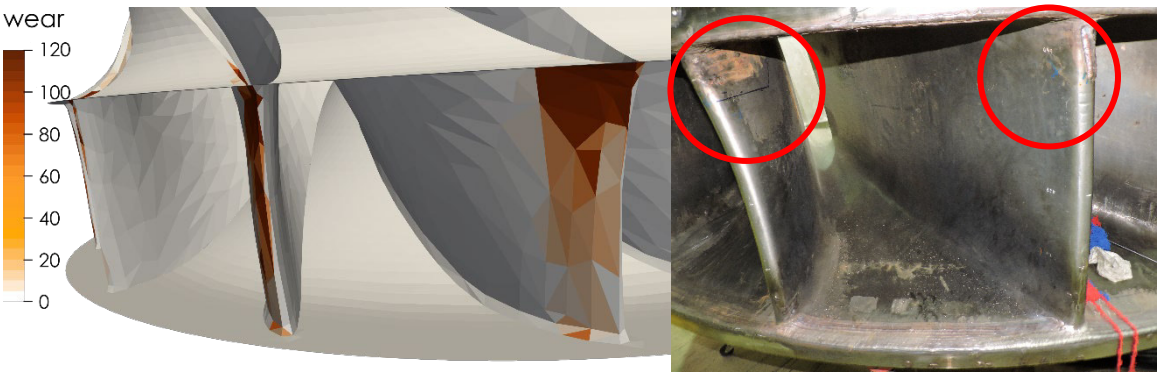

Figura 3.27 Comparación de daños en el material por desgaste en el rodete obtenidos mediante simulación numérica.

(Fuente: Propia)

4 CONCLUSIONES

4.1 Conclusiones

Se desarrolló una metodología válida para estudiar numéricamente el flujo con inclusión de material granular en una turbina tipo Francis operando en condiciones reales en la central hidroeléctrica San Francisco en la provincia de Tungurahua, utilizando el software de código abierto OpenFOAM, LIGGGHTS y CFDEM coupling.

Se pudo realizar una malla fija 3D capaz de reproducir correctamente el flujo particulado a través de una turbina tipo Francis en estado transitorio, utilizando ICEM y las aplicativos de manipulación de mallas incluidos en OpenFOAM.

Se optimizó los dominios computacionales propuestos por Velasco y Mora, logrando una reducción del 72% en el número total de elementos del mallado 3D obteniendo predicciones de eficiencia y potencia generada con un 5.8% de error comparados con el valor real.

El estudio de independencia de malla determinó la confiabilidad de los datos obtenidos con la malla optimizada, este dominio computacional cuenta con 1.3 millones de elementos y muestra una clara estabilidad en los resultados y la calidad necesaria para un correcto desarrollo del estudio propuesto. Por otro lado, las simulaciones realizadas con las mallas media y fina obtuvieron resultados con muy poca variación, pero con tiempos de ejecución mucho más extensos que proponían un gasto innecesario de poder computacional. Mientras que la malla muy fina presentaba errores debido a bajos valores de calidad en puntos cruciales en donde existían fuertes cambios en la geometría.

Se realizaron dos tipos de simulaciones, una en estado estable con el método SIMPLE y la segunda en estado transitorio con el método PISO. La primera simulación determinó la correcta reproducción del fenómeno en la malla optimizada en base a los resultados experimentales y los obtenidos en estudios anteriores. Por otro lado, en la segunda se implementó el método DEM para el estudio del material granular en estado inestable, obteniendo resultados satisfactorios en base a las aproximaciones realizadas.

Se concluye que el número de Courant es un valor adimensional importante en el acople de CFD y DEM, ya que las partículas que de manera errada atravesaron las superficies limítrofes lo hicieron en las zonas en que contienen altos números de Courant como en el rodete, alabes fijos y alabes directores.

La trayectoria de las partículas reveló información importante con respecto al desgaste del material por impactos, ya que se pudo apreciar con detalle los lugares en donde las partículas impactan y cambian su dirección a lo largo de todo su recorrido en la turbina tipo Francis.

Se llevó a cabo un análisis cualitativo de las posibles zonas de erosión debido a material particular comparándolas con los datos experimentales a disposición determinando que la simulación reproduce adecuadamente la naturaleza del fenómeno de desgaste en el rodete y alabes directores de la turbina Francis. Sin embargo, esta simulación fue llevada a cabo bajo la condición de malla fija, lo cual permite concluir que para un correcto análisis de desgaste en el rodete es necesario el uso de una metodología de malla móvil para poder estudiar completamente este fenómeno.

El uso del método de elemento discreto (DEM) con el acople a la dinámica de fluidos computacional (CFD) como modelo matemático de estudio de flujo particulado presenta una ventaja en el análisis de fenómenos complejos que tengan involucren estas dos fases, ya que captura adecuadamente la naturaleza discreta de la fase sólida y la naturaleza continua de la fase liquida.

Una vez evaluados los resultados obtenidos se concluye que, si se puede utilizar OpenFOAM en conjunto con LIGGGHTS para el estudio numérico de flujo particulado dentro de una turbina tipo Francis, presentando un avance en el estudio de CFD y DEM en software libre y de código abierto.

4.2 Trabajos futuros.

Este estudio propuso un dominio computacional estático para el estudio de un fenómeno transitorio, para lograr una primera aproximación del uso de software libre sobre el flujo particular en turbomaquinaria, sin embargo, es necesario continuar estudiando este caso implementando una malla rotativa móvil en la zona del rodete, acercando los resultados aún más al fenómeno físico real. De momento implementar este modelo móvil rotativo supone un gasto computacional exorbitante, lo cual limita el alcance de este estudio. Por tal motivo se espera que, con el avance tecnológico y con un acceso más fácil a tecnología más poderosa y eficiente en un futuro se pueda implementar esta característica en el estudio de flujo particular en turbomaquinaria.

Por otro lado, también es posible la implementación de partículas no esféricas que representen con mayor precisión la geometría de los sedimentos encontrados en el cauce del rio, y estudiar el efecto de la geometría de las partículas sobre turbomaquinaria.

Finalmente, se puede considerar analizar el desgaste desde un punto de vista cuantitativo para determinar con valores reales la perdida de materia y así también el periodo de vida útil de las piezas y proponer de manera puntual zonas de mantenimiento preventivo o intentar implementar mejoras en el material que sufra más desgaste debido al impacto de las partículas.

Referencias Bibliográficas

[1] V. H. Hidalgo, X. Luo, A. P. Peña, E. A. Valencia, R. Soto, and A. Yu, "Benefits of hydropower research in Ecuador using OpenFOAM based on CFD technology (A practical cavitation study for NACA0015)," *IX Congr. Cienc. y Tecnol. ESPE*, no. MAY, pp. 123–127, 2014, [Online]. Available: https://www.researchgate.net/publication/262068755_Benefits_of_hydropower_research_in_Ecuador_using_OpenFOAM_based_on_CFD_technology_%28A_practical_cavitation_study_for_NACA0015%29.

[2] ARCONEL, "Balance Nacional de Energía Eléctrica 2020," *Biblioteca*, 2020. https://www.regulacionelectrica.gob.ec/balance-nacional/ (accessed May 25, 2021).

[3] Ministerio de Energía y Recursos Renovables, "Plan Maestro de la Electricidad. Expansión de la generación," 2017, [Online]. Available: https://www.recursosyenergia.gob.ec/wp-content/uploads/2020/01/4.-EXPANSION-DE-LA-GENERACION.pdf.

[4] P. Guangjie, W. Zhengwei, X. Yexiang, and L. Yongyao, "Abrasion predictions for Francis turbines based on liquid-solid two-phase fluid simulations," *Eng. Fail. Anal.*, vol. 33, pp. 327–335, 2013, doi: 10.1016/j.engfailanal.2013.06.002.

[5] C. Mora, "Análisis de la eficiencia de una tipo turbina francis con caracteristicas similares a la central hidroeléctrica San Francisco-Ecuador.," 2018.

[6] H. Zhang and L. Zhang, "Numerical simulation of cavitating turbulent flow in a high head Francis turbine at part load operation with OpenFOAM," *Procedia Eng.*, vol. 31, no. 2011, pp. 156–165, 2012, doi: 10.1016/j.proeng.2012.01.1006.

[7] R. Betancourt, "Estudio De Mallado Estructurado 3D Rotativo En Openfoam Para Aplicaciones En Turbinas Tipo Francis," 2020.

[8] OpenFOAM, *User guide*, no. December. 2020.

[9] "¿Qué es el software libre? - Proyecto GNU - Free Software Foundation." https://www.gnu.org/philosophy/free-sw.es.html (accessed May 31, 2021).

[10] R. Yepez, Y. Li, M. Hallack, and L. David, "The energy path of latin america and the caribbean," *Interam. Dev. bank*, 2018.

[11] I. Renewable and E. Agency, *GLOBAL ENERGY*. .

[12] J. Smeaton, "CHAPTER 4 Hydropower Turbines," 2018, doi: 10.1016/B978-0-12-812906-7.00004-1.

[13] S. L. Dixon and C. A. Hall, "Hydraulic Turbines," *Fluid Mech. Thermodyn. Turbomach.*, pp. 303–355, Jan. 2010, doi: 10.1016/B978-1-85617-793-1.00009-2.

[14] R. Goyal, *Flow field in a high head Francis turbine draft tube during transient operations*. 2017.

[15] "Three Gorges - the world's biggest hydropower plant." https://www.nsenergybusiness.com/projects/three-gorges-dam-hydropower-station/ (accessed Oct. 11, 2021).

[16] "FAQ | ITAIPU BINACIONAL." https://www.itaipu.gov.py/es/sala-de-prensa/faq (accessed Oct. 11, 2021).

[17] CELEC, “San Francisco,” *Centrales*, 2016. https://www.celec.gob.ec/hidroagoyan/index.php/centrales/san-francisco (accessed Nov. 25, 2021).

[18] S. M. Weber, “Granulare Materie I Schüttgut in Ruhe - ein ungewöhnlicher Festkörper.”

[19] J. Duran, “Sands, Powders, and Grains,” 2000, doi: 10.1007/978-1-4612-0499-2.

[20] C. R. Woodcock and J. S. Mason, “Bulk Solids Handling,” 1988, doi: 10.1007/978-94-009-2635-6.

[21] M. Sperl, “Experiments on corn pressure in silo cells - Translation and comment of Janssen’s paper from 1895,” *Granul. Matter*, vol. 8, no. 2, pp. 59–65, May 2006, doi: 10.1007/S10035-005-0224-Z.

[22] K. Nilay, “High Temperature Airflow and Particle Deposition in Human Upper Lung-airway Models with a Focus on the Mucus-layer Dynamics.,” 2019.

[23] J. A. Maza and M. García, “Transporte de sedimentos,” *Informacion Tecnologica*, vol. 13, no. 6. pp. 137–142, 2002.

[24] I. Martín, R. Salcedo, and R. Font, “Mecánica De Fluidos Transporte Cantidad Movimiento Sólido-Fluido,” *Mecánica de fluidos*, p. 50, 2011, [Online]. Available: https://rua.ua.es/dspace/bitstream/10045/20299/9/tema4_flujo externo.pdf.

[25] A. M. Valverde, “Estimación De Sedimentos, Mediante El Modelo Unidimensional Hec-Ras, En El Río Costero Tropical Hato - Panamá,” p. 79, 2019, [Online]. Available: http://repositorio.lamolina.edu.pe/handle/UNALM/3878.

[26] R. Mejia-Alvarez and C. Nieto, “Dinámica de fluidos computacional aplicada al estudio de regeneradores térmicos,” *Dyna*, vol. 71, 2004.

[27] Grupo Termotecnia, “Fluidodinámica Computacional,” p. 8, 2016, [Online]. Available: http://bibing.us.es/proyectos/abreproy/4882/fichero/Análisis+del+sistema+de+ventilación+de+un+secadero+solar+mediante+tecnica+fluidodinámica+computacional%252FCapitulo+5.+Fluidodinámica+Computacional.pdf.

[28] S. Jamshed, “Introduction to CFD,” *Using HPC Comput. Fluid Dyn.*, no. December 2015, pp. 1–20, 2015, doi: 10.1016/b978-0-12-801567-4.00001-5.

[29] J. Pedlosky, “Geophysical fluid dynamics.,” *Geophys. fluid Dyn.*, 1979.

[30] E. P. A. Ojeda, P. M. López, S. A. Farah, and R. A. C. Ramirez, “Cfd como una herramienta para diseño de productos,” *Memorias Del Xv Congr. Int. La Somim*, pp. 431–442, 2009.

[31] J. Sodja, “Turbulence models in CFD,” *Univ. Ljubljana*, no. March, pp. 1–18, 2007, [Online]. Available: http://mafija.fmf.uni-lj.si/seminar/files/2006_2007/Turbulence_models_in_CFD.pdf.

[32] G. Richmond, “Modelos de turbulencia introductorio,” *Res. J. Environ. Sci.*, no. September, pp. 1–16, 2019, [Online]. Available: https://www.researchgate.net/publication/335611158_Modelos_de_turbulencia_introductorio.

[33] M. Shukla, R. Jain, V.Prasad, and S.N.Shukla, “CFD analysis of 3-D flow for Francis turbine,” *MIT Int. J. Mech. Eng.*, vol. 1, pp. 93–100, 2011.

[34] A. Molina, “Modelamiento de una turbina de Pelton en CFD e impresión 3D del

prototipo," pp. 1–16, 2000.

[35] R. Navthar, J. Tejas, D. Saurabh, D. Nitish, and A. Anand, "Cfd Analysis of Francis Turbine," vol. 4, no. 07, pp. 3194–3199, 2012.

[36] A. K. Sanjay Jain, R.P.Saini, "Cfd Approach for Prediction of Efficiency of Francis Turbine," *Ighem*, no. November, pp. 1–7, 2010.

[37] H. Aref and S. Balachandar, "Spatial Discretization," in *A First Course in Computational Fluid Dynamics*, Cambridge: Cambridge University Press, 2017, pp. 118–165.

[38] J. Sarrate, E. Ruiz-Gironés, and X. Roca, "Unstructured and Semi-Structured Hexahedral Mesh Generation Methods," *Comput. Technol. Rev.*, vol. 10, pp. 35–64, 2014, doi: 10.4203/ctr.10.2.

[39] I. Sadrehaghighi, "Structure Meshing for CFD," no. 20, pp. 1–89, 2020, [Online]. Available: https://www.researchgate.net/publication/339295881_Structure_Meshing_for_CFD.

[40] J. Sánchez, "Dinámica de fluidos computacional," *Simulación numérica directa en paralelo las ecuaciones Navier.Stokes en flujo con capa límite desprendida. Apl. en Instal. Deport. con gradas cubiertas*, pp. 65–72, 2004.

[41] F. GARCÍA, C. PALACIO, and U. GARCÍA, "GENERACIÓN DE MALLAS NO ESTRUCTURADAS PARA LA IMPLEMENTACION DE MODELOS NUMÉRICOS ," *DYNA* , vol. 76. scieloco , pp. 17–25, 2009.

[42] Z. Ali, P. G. Tucker, and S. Shahpar, "Optimal mesh topology generation for CFD," *Comput. Methods Appl. Mech. Eng.*, vol. 317, pp. 431–457, 2017, doi: 10.1016/j.cma.2016.12.001.

[43] T. J. Baker, "Mesh generation: Art or science?," *Prog. Aerosp. Sci.*, vol. 41, no. 1, pp. 29–63, 2005, doi: 10.1016/j.paerosci.2005.02.002.

[44] C. E. Guamushig Gualotuña, "Modelamiento y simulación multifísica de materiales granulares," Apr. 2016, Accessed: Oct. 31, 2021. [Online]. Available: http://bibdigital.epn.edu.ec/handle/15000/15864.

[45] H. P. Zhu, Z. Y. Zhou, R. Y. Yang, and A. B. Yu, "Discrete particle simulation of particulate systems: A review of major applications and findings," *Chem. Eng. Sci.*, vol. 63, no. 23, pp. 5728–5770, 2008, doi: 10.1016/j.ces.2008.08.006.

[46] P. A. Cundall and O. D. L. Strack, "A discrete numerical model for granular assemblies," *Géotechnique*, vol. 29, no. 1, pp. 47–65, 1979, doi: 10.1680/geot.1979.29.1.47.

[47] S. P. Timoshenko and J. N. Goodier, "Theory of elasticity by Timoshenko and Goodier.pdf." 1970.

[48] M. A. El-Emam, L. Zhou, W. Shi, C. Han, L. Bai, and R. Agarwal, *Theories and Applications of CFD–DEM Coupling Approach for Granular Flow: A Review*, no. 0123456789. Springer Netherlands, 2021.

[49] H. Zhou, G. Wang, C. Jia, and C. Li, "A novel, coupled CFD-DEM model for the flow characteristics of particles inside a pipe," *Water (Switzerland)*, vol. 11, no. 11, 2019, doi: 10.3390/w11112381.

[50] E. M. Smuts, D. A. Deglon, and C. J. Meyer, "Methodology for coupled CFD-DEM modelling of particulate suspension rheology," *26th Int. Miner. Process. Congr.*

IMPC 2012 Innov. Process. Sustain. Growth - Conf. Proc., no. May, pp. 5068–5079, 2012.

[51] A. Hager, C. Kloss, and C. Goniva, *Combining Open Source and Easy Access in the field of DEM and coupled CFD-DEM: LIGGGHTS® CFDEM®coupling and CFDEM®workbench*, vol. 43. Elsevier Masson SAS, 2018.

[52] G. Liu, F. Yu, H. Lu, S. Wang, P. Liao, and Z. Hao, "CFD-DEM simulation of liquid-solid fluidized bed with dynamic restitution coefficient," *Powder Technol.*, vol. 304, pp. 186–197, 2016, doi: 10.1016/j.powtec.2016.08.058.

[53] E. Guascal and P. Quispe, "Desarrollo y estudio de un mallado estructurado optimizado para una turbina tipo Francis con validacion experimental," p. 157, 2019, [Online]. Available: https://bibdigital.epn.edu.ec/handle/15000/786%0Ahttps://bibdigital.epn.edu.ec/bitstream/15000/786/1/CD-1222.pdf.

[54] X. L. Liu, W. Q. Tao, and Y. L. He, "A simple method for improving the SIMPLER algorithm for numerical simulations of incompressible fluid flow and heat transfer problems," *Eng. Comput. (Swansea, Wales)*, vol. 22, no. 8, pp. 921–939, 2005, doi: 10.1108/02644400510626488/FULL/XML.

[55] D. C. Wilcox, "Turbulence modeling for CFD," p. 522, 2006.

[56] L. Stoessel, "Numerical simulations of the flow in the Francis-99 turbine," 2014, Accessed: Jan. 03, 2022. [Online]. Available: https://odr.chalmers.se/handle/20.500.12380/204724.

[57] B. Aakti, O. Amstutz, E. Casartelli, G. Romanelli, and L. Mangani, "On the performance of a high head Francis turbine at design and off-design conditions," *J. Phys. Conf. Ser.*, vol. 579, no. 1, p. 012010, Jan. 2015, doi: 10.1088/1742-6596/579/1/012010.

[58] A. Abdoli, "Optimization of Cooling Protocols for Hearts Destined for Transplantation," 2014.

[59] T. Belamri, A. Braune, P. Galpin, and C. Cornelius, "CFD Analysis of a 15 Stage Axial Compressor: Part II — Results," *Proc. ASME Turbo Expo*, vol. 6 PART B, pp. 1009–1017, Nov. 2008, doi: 10.1115/GT2005-68262.

[60] R. Courant, K. Friedrichs, and H. Lewy, "Über die partiellen Differenzengleichungen der mathematischen Physik," *Math. Ann.*, vol. 100, no. 1, pp. 32–74, Dec. 1928, doi: 10.1007/BF01448839.

[61] V. M. Cock, "Hidro-abrasión en turbinas Francis: caso de estudio en cinco aprovechamientos hidroeléctricos instalados en la zona del Batolito antioqueño.," pp. 1–21, 2019.

[62] M. Castro and P. Paúl, "Bases para el diseño hidráulico de la estructura de entrada, cámara de desarenación y estructuras de limpieza, con caudales entre 20 m3/s ≥ Q ≥ 2m3/s para desarenadores en aprovechamientos hidroeléctricos.," 2013, Accessed: Nov. 02, 2021. [Online]. Available: http://www.dspace.uce.edu.ec/handle/25000/2210.

[63] Changjiang Institute of Survey Planning Design and Research (CISPDR), "Plan hidráulico regional de lademarcación hidrográfica pastaza," 2016.

[64] S. Olivella Pastallé, "Mecánica de suelos : problemas resueltos," 2001, Accessed: Jan. 06, 2022. [Online]. Available: https://upcommons.upc.edu/handle/2099.3/36251.

[65] “Introducción — Blender Manual.” https://docs.blender.org/manual/es/2.82/getting_started/about/introduction.html (accessed Jan. 07, 2022).

[66] I. Finnie, “Erosion of surfaces by solid particles,” *Wear*, vol. 3, no. 2, pp. 87–103, 1960, doi: 10.1016/0043-1648(60)90055-7.

ANEXOS

ANEXO I.

SCRIPT PARA AUTOPOCISIONAMIENTO Y COSTURA DE DOMINIOS COMPUTACIONALES.

Este script utiliza como entradas los 4 archivos de extensión .msh obtenidos tras la modificación de las mallas en ICEM y los convierte automáticamente en una sola malla en formato OpenFOAM.

```
#!/bin/sh
cd ${0%/*} || exit 1    # Run from this directory

# Source tutorial run functions
. $WM_PROJECT_DIR/bin/tools/RunFunctions
#-------------------------------------------------------------------------------
-
#1. LIMPIAMOS EL CASO
    rm -rf 1*
    rm -rf 2*
    rm -rf 3*
    rm -rf 4*
    rm -rf 5*
    rm -rf log*

    cp -r ../2* .
    cp -r ../3* .
    cp -r ../4* .
    cp -r ../5* .
    mkdir log
#-------------------------------------------------------------------------------
-
#-------------------------------------------------------------------------------
-
#2. UBICAMOS (ROTAMOS Y TRSLADAMOS) LAS PIEZAS A SU SITIO
runApplication transformPoints -yawPitchRoll "(0 0 0)" -case 2.draft/
mv log.transformPoints log/log.transformPoints.yawPitchRoll.draft

runApplication transformPoints -translate "(0 0 0)" -case 2.draft/
mv log.transformPoints log/log.transformPoints.translate.draft

runApplication transformPoints -yawPitchRoll "(0 0 -90)" -case 3.runner/
mv log.transformPoints log/log.transformPoints.yawPitchRoll.runner

runApplication transformPoints -translate "(0 0 -0.3805)" -case 3.runner/
mv log.transformPoints log/log.transformPoints.translate.runner

runApplication transformPoints -yawPitchRoll "(0 0 -90)" -case 4.guide/
mv log.transformPoints log/log.transformPoints.yawPitchRoll.guide

runApplication transformPoints -translate "(0 0 -0.04031637)" -case 4.guide/
mv log.transformPoints log/log.transformPoints.translate.guide

runApplication transformPoints -yawPitchRoll "(90 0 180)" -case
5.spiralCasing/
mv log.transformPoints log/log.transformPoints.yawPitchRoll.spiralCasing

runApplication transformPoints -translate "(0 0 -0.0367)" -case
5.spiralCasing/
mv log.transformPoints log/log.transformPoints.translate.spiralCasing
```

```
#-----------------------------------------------------------------------------
-
#-----------------------------------------------------------------------------
-
# 3. PREPARAMOS LA MALLA GENERAL
rm -rf 1.francis
rm -rf STL/

cp -rf 5.spiralCasing 1.francis
mv 1.francis/5.spiralCasing.foam 1.francis/francis.foam
#-----------------------------------------------------------------------------
-
#-----------------------------------------------------------------------------
-
# 4. COSEMOS SPIRALCASING CON GUIDE
runApplication mergeMeshes -overwrite 1.Francis/ 4.guide/
runApplication stitchMesh -partial -case ./1.francis -overwrite IN_GUIDE_VANE
OUT_SPIRAL_CASE
mv log.stitchMesh log/log.stitchMesh.1
mv log.mergeMeshes log/log.mergeMeshes.1
rm -rf 1.francis/0
#-----------------------------------------------------------------------------
-
#-----------------------------------------------------------------------------
-
# 5. COSEMOS GUIDE CON RUNNER
 runApplication mergeMeshes -overwrite 1.Francis/ 3.runner/

 rm -rf  1.francis/constant/polyMesh/*Zones
 rm -rf  1.francis/constant/polyMesh/meshModifiers

 runApplication stitchMesh -overwrite -case ./1.francis OUT_GUIDE_VANE
IN_RUNNER
 mv log.stitchMesh log/log.stitchMesh.2
 mv log.mergeMeshes log/log.mergeMeshes.2
 rm -rf 1.francis/0
#-----------------------------------------------------------------------------
-
#-----------------------------------------------------------------------------
-
# 6. COSEMOS RUNNER CON DRAFT
runApplication mergeMeshes -overwrite 1.Francis/ 2.draft/

 rm -rf  1.francis/constant/polyMesh/*Zones
 rm -rf  1.francis/constant/polyMesh/meshModifiers

runApplication stitchMesh -case ./1.francis -overwrite IN_DRAFT_TUBE
OUT_RUNNER
mv log.mergeMeshes log/log.mergeMeshes.3
mv log.stitchMesh log/log.stitchMesh.3
rm -rf 1.francis/0
#-----------------------------------------------------------------------------
-
#-----------------------------------------------------------------------------
-
# 7. RETIRAMOS PATCHES EXTRAS
 runApplication createPatch -overwrite -case 1.francis/ -dict
../createPatchDict
```

ANEXO II

EJECUTABLE CFDEM

```
#!/bin/bash

#=================================================================#
# Codigo para ejecutar caso CFDEM Turbina Francis 2021 - Mateo N.
#=================================================================#
casePath="$(dirname "$(readlink -f ${BASH_SOURCE[0]})")"
runOctave="true"
postproc="false"

#Verificamos si la malla fue creada antes
if [ -f "$casePath/CFD/constant/polyMesh/points" ]; then
    echo "La malla si existe - Se usa la malla existente"
else
    echo "No existe la malla, se copia del caso de mallado"
    cp -rf polyMesh CFD/constant/
fi

if [ -f "$casePath/DEM/post/restart/liggghts.restart" ];  then
    echo "LIGGGHTS fue ejecutado antes, letendo CFDEM.restart"
else
    #- run DEM in new terminal
    $casePath/parDEMrun.sh
fi

     #-------Comentar toda esta seccion si se quiere reponer la
simulacion ---------------------------------

     # Adaptamos archivos para simular por primera vez
     cp $casePath/CFD/constant/liggghtsCommands_run
$casePath/CFD/constant/liggghtsCommands
     cp $casePath/CFD/constant/couplingProperties_run
$casePath/CFD/constant/couplingProperties
     cp $casePath/CFD/system/controlDict_run
$casePath/CFD/system/controlDict
     . $casePath/parCFDDEMrun.sh
        #- Esperamos a que termine
    echo "la ismulacion termino"
    read
     #-----------------------------------------------------------------
-------------------------------------


# Adaptamos archivos para reponer simulacion.
cp $casePath/CFD/constant/liggghtsCommands_restart
$casePath/CFD/constant/liggghtsCommands
cp $casePath/CFD/constant/couplingProperties_restart
$casePath/CFD/constant/couplingProperties
cp $casePath/CFD/system/controlDict_restart
$casePath/CFD/system/controlDict
. $casePath/parCFDDEMrun.sh
    #- Esperamos a que termine
    echo "la ismulacion termino"
    read
#-------------------------------------------------------#


#- Limpiamos el caso
```

```
echo "deleting data at: $casePath :\n"
source $WM_PROJECT_DIR/bin/tools/CleanFunctions
cd $casePath/CFD
cleanCase
rm -r $casePath/CFD/constant/liggghtsCommands
rm -r $casePath/CFD/constant/couplingProperties
rm -r $casePath/CFD/clockData
rm -r $casePath/DEM/post/*.*
#rm -r $casePath/DEM/post/restart/*.*
touch $casePath/DEM/post/.gitignore
touch $casePath/DEM/post/restart/.gitignore
echo "done"
```

ANEXO III

ligggthsCommands

```
/*-------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  1.4                                   |
|   \\  /    A nd           | Web:      http://www.openfoam.org               |
|    \\/     M anipulation  |                                                 |
\*-------------------------------------------------------------------------*/

FoamFile
{
    version         2.0;
    format          ascii;

    root            "";
    case            "";
    instance        "";
    local           "";

    class           dictionary;
    object          liggghtsCommands;
}

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

liggghtsCommandModels
(
    runLiggghts
    writeLiggghts
);
// ************************************************************************* //
writeLiggghtsProps
{
    writeLast off;
    //path "../DEM"; // optional setting
    writeName "post/restart/liggghts.restartCFDEM";
    overwrite on;
}
```

ANEXO IV

couplingProperties

```
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  1.4                                   |
|   \\  /    A nd           | Web:      http://www.openfoam.org               |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/

FoamFile
{
    version         2.0;
    format          ascii;

    root            "";
    case            "";
    instance        "";
    local           "";

    class           dictionary;
    object          couplingProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
//===========================================================================//
// sub-models & settings

modelType "A"; // A or B

couplingInterval 5;

voidFractionModel divided;//centre;//

locateModel engine;//engineIB;//

meshMotionModel noMeshMotion;

IOModel basicIO;

probeModel off;

dataExchangeModel twoWayMPI;

averagingModel dense;//dilute;//

clockModel off;//standardClock;//

smoothingModel off;// localPSizeDiffSmoothing;// constDiffSmoothing; //

forceModels
(
    gradPForce
    viscForce
    KochHillDrag
    fieldStore
);

momCoupleModels
(
    implicitCouple
);
//turbulenceModelType "RASProperties";//"LESProperties";//OFversion24x
turbulenceModelType turbulenceProperties;//"LESProperties";//OFversion30x

//===========================================================================//
```

```
// sub-model properties
localPSizeDiffSmoothingProps
{
    lowerLimit 0.1;
    upperLimit 1e10;
    dSmoothingLength 1.5e-3;
    Csmoothing    1.0;
}
constDiffSmoothingProps
{
    lowerLimit 0.1;
    upperLimit 1e10;
    smoothingLength 1.5e-3;
}
implicitCoupleProps
{
    velFieldName "U";
    granVelFieldName "Us";
    voidfractionFieldName "voidfraction";
}
ArchimedesProps
{
    gravityFieldName "g";
}
gradPForceProps
{
    pFieldName "p";
    voidfractionFieldName "voidfraction";
    velocityFieldName "U";
    interpolation true;
}
viscForceProps
{
    velocityFieldName "U";
    interpolation true;
}
volWeightedAverageProps
{
    scalarFieldNames
    (
        voidfraction
    );
    vectorFieldNames
    (
    );
    upperThreshold 0.999;
    lowerThreshold 0;
    verbose true;
}
totalMomentumExchangeProps
{
    implicitMomExFieldName "Ksl";
    explicitMomExFieldName "none";
    fluidVelFieldName "U";
    granVelFieldName "Us";
}
GidaspowDragProps
{
    verbose true;
    velFieldName "U";
    voidfractionFieldName "voidfraction";
    granVelFieldName "Us";
    interpolation true;
    phi 1;
}
DEMbasedDragProps
{
    velFieldName "U";
```

```
    voidfractionFieldName "voidfraction";
}
DiFeliceDragProps
{
    velFieldName "U";
    voidfractionFieldName "voidfraction";
    granVelFieldName "Us";
    interpolation true;
}
KochHillDragProps
{
    velFieldName "U";
    voidfractionFieldName "voidfraction";
    interpolation true;
    implForceDEM true;
}
BeetstraDragProps
{
    velFieldName "U";
    gravityFieldName "g";
    rhoParticle     2000.;
    voidfractionFieldName "voidfraction";
    interpolation true;
    useFilteredDragModel ;
    useParcelSizeDependentFilteredDrag ;
    k       0.05;
    aLimit  0.0;
//    verbose true;
}
RongDragProps
{
    verbose true;
    velFieldName "U";
    voidfractionFieldName "voidfraction";
    interpolation true;
    implForceDEM true;
    implForceDEMaccumulated true;
    granVelFieldName "Us";
}
virtualMassForceProps
{
    velFieldName "U";
}
particleCellVolumeProps
{
    upperThreshold 0.999;
    lowerThreshold 0.;
    verbose true;
}
fieldStoreProps
{
    scalarFieldNames
    (
    );

    vectorFieldNames
    (
        "U"
    );
}
fieldVolWeightedProps
{
    startTime 0;

    scalarFieldNames
    (
    );
```

```
    vectorFieldNames
    (
        "U"
    );
    useVolumeFraction true;
    volumeFractionName "voidfraction";
}
oneWayVTKProps
{
    couplingFilename "vtk_out%4.4d.vtk";
    maxNumberOfParticles 30000;
}
twoWayFilesProps
{
    maxNumberOfParticles 10100;
}
centreProps
{
    alphaMin 0.1;
}
"(engineProps|turboEngineProps|turboEngineM2MProps)"
{
    treeSearch true;
}
engineIBProps
{
    treeSearch true;
    zSplit 8;
    xySplit 16;
}
dividedProps
{
    alphaMin 0.1;
    scaleUpVol 1.0;
    //procBoundaryCorrection true;
}
"(twoWayMPIProps|M2MProps)"
{
    liggghtsPath "../DEM/in.liggghts_restart";
}
```

ANEXO V

in.liggghts_init

```
echo            both
log             ../DEM/log.liggghts
thermo_log      ../DEM/post/thermo.txt

atom_style      granular
atom_modify     map array
communicate     single vel yes

boundary        f f f
newton          off

units           si
processors      2 2 4

region          reg block -4.2 5.1 -6.2 20.1 -1.6 9.4 units box
create_box      1 reg

neighbor        0.1 bin
neigh_modify    delay 0


# Material properties required for granular pair styles

fix        m1 all property/global youngsModulus peratomtype 5.e6
fix        m2 all property/global poissonsRatio peratomtype 0.45
fix        m3 all property/global coefficientRestitution peratomtypepair 1 0.3
fix        m4 all property/global coefficientFriction peratomtypepair 1 0.5
fix        m5 all property/global k_finnie peratomtypepair 1 0.2

# pair style
pair_style  gran model hertz tangential history # Hertzian without cohesion
pair_coeff  * *

# timestep, gravity
timestep    0.0001
fix         gravi all gravity 9.81 vector 0 0 1

# walls
    fix     Walls_Spiral    all mesh/surface/stress file
../STL/Zalls_Spiral.stl type   1  element_exclusion_list read
../STL/ZSpiral.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Runner    all mesh/surface/stress file
../STL/Zalls_Runner.stl type   1  element_exclusion_list read
../STL/ZRunner.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Blades    all mesh/surface/stress file
../STL/Zalls_Blades.stl type   1  element_exclusion_list read
../STL/ZBlades.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Guide     all mesh/surface/stress file
../STL/Zalls_Guide.stl type    1  element_exclusion_list read ../STL/ZGuide.ex
curvature_tolerant yes stress on wear finnie
    fix     Walls_Draft     all mesh/surface/stress file
../STL/Zalls_Draft.stl type    1  element_exclusion_list read ../STL/ZDraft.ex
curvature_tolerant yes stress on wear finnie

    fix     wall all wall/gran model hertz tangential history mesh n_meshes 5
meshes Walls_Spiral  Walls_Runner Walls_Blades Walls_Guide Walls_Draft

# particle distributions and insertion
region  bc cylinder y 3.55 0 1.3 -6 0  units box
    fix     pts1 all particletemplate/sphere 15485863 atom_type 1 density
constant 1600 radius constant 0.0100
    fix     pts2 all particletemplate/sphere 32452843 atom_type 1 density
constant 1600 radius constant 0.0085
```

```
    fix     pts3 all particletemplate/sphere 32452867 atom_type 1 density
constant 1600 radius constant 0.0070
    fix     pts4 all particletemplate/sphere 49979687 atom_type 1 density
constant 1600 radius constant 0.0055
    fix     pts5 all particletemplate/sphere 67867967 atom_type 1 density
constant 1600 radius constant 0.0040
    fix     pts6 all particletemplate/sphere 86028121 atom_type 1 density
constant 1600 radius constant 0.0025

fix     pdd1 all particledistribution/discrete/numberbased 15485867 6 pts1
0.1667 pts2 0.1667 pts3 0.1667 pts4 0.1667 pts5 0.1667 pts6 0.1667

fix     ins all insert/pack seed 86028157 distributiontemplate pdd1 vel
constant 0. 8.827 0. insert_every once overlapcheck yes all_in yes
particles_in_region 600 region bc

# apply nve integration to all particles that are inserted as single particles
fix     integr all nve/sphere

# screen output
compute         rke all erotate/sphere
thermo_style    custom step atoms ke c_rke vol
thermo          1000
thermo_modify   lost ignore norm no
compute_modify  thermo_temp dynamic yes

# insert the first particles so that dump is not empty
run         1

# dump Particles
dump        dmp1 all custom/vtk 50 ../DEM/post/particles*.vtk id type x y z vx
vy vz fx fy fz radius


# dump Walls
dump        dmpw1 all mesh/gran/VTK 50 ../DEM/post/1.Walls_Spiral*.vtk stress
vel stresscomponents wear Walls_Spiral
dump        dmpw2 all mesh/gran/VTK 50 ../DEM/post/2.Walls_Runner*.vtk stress
vel stresscomponents wear Walls_Runner
dump        dmpw3 all mesh/gran/VTK 50 ../DEM/post/3.Walls_Blades*.vtk stress
vel stresscomponents wear Walls_Blades
dump        dmpw4 all mesh/gran/VTK 50 ../DEM/post/4.Walls_Guide*.vtk stress
vel stresscomponents wear Walls_Guide
dump        dmpw5 all mesh/gran/VTK 50 ../DEM/post/5.Walls_Draft*.vtk stress
vel stresscomponents wear Walls_Draft

run         1 upto #10000
write_restart   post/restart/liggghts.restart
```

ANEXO VI

in.liggghts_run

```
echo            both
log             ../DEM/log.liggghts
thermo_log      ../DEM/post/thermo.txt

atom_style      granular
atom_modify     map array
communicate     single vel yes

boundary        f f f
newton          off

units           si
processors      2 2 4

# read the restart file
read_restart    ../DEM/post/restart/liggghts.restart

neighbor        0.1 bin
neigh_modify    delay 0

# Material properties required for granular pair styles

fix         m1 all property/global youngsModulus peratomtype 5.e6
fix         m2 all property/global poissonsRatio peratomtype 0.45
fix         m3 all property/global coefficientRestitution peratomtypepair 1
0.3
fix         m4 all property/global coefficientFriction peratomtypepair 1 0.5
fix         m5 all property/global k_finnie peratomtypepair 1 0.2

# pair style
pair_style  gran model hertz tangential history # Hertzian without cohesion
pair_coeff  * *

# timestep, gravity
timestep    0.0001
fix         gravi all gravity 9.81 vector 0 0 1

# walls

    fix     Walls_Spiral    all mesh/surface/stress file
../STL/Zalls_Spiral.stl type   1  element_exclusion_list read
../STL/ZSpiral.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Runner    all mesh/surface/stress file
../STL/Zalls_Runner.stl type   1  element_exclusion_list read
../STL/ZRunner.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Blades    all mesh/surface/stress file
../STL/Zalls_Blades.stl type   1  element_exclusion_list read
../STL/ZBlades.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Guide     all mesh/surface/stress file
../STL/Zalls_Guide.stl type    1  element_exclusion_list read ../STL/ZGuide.ex
curvature_tolerant yes stress on wear finnie
    fix     Walls_Draft     all mesh/surface/stress file
../STL/Zalls_Draft.stl type    1  element_exclusion_list read ../STL/ZDraft.ex
curvature_tolerant yes stress on wear finnie

    fix     wall all wall/gran model hertz tangential history mesh n_meshes 5
meshes Walls_Spiral  Walls_Runner Walls_Blades Walls_Guide Walls_Draft


# cfd coupling
fix         cfd all couple/cfd couple_every 50 mpi
fix         cfd2 all couple/cfd/force/implicit
#fix         cfd2 all couple/cfd/force/accumulator RongDrag 10 1.5e-3
```

```
#fix         cfd2 all couple/cfd/force/implicit/accumulated #CrankNicolson 0.5

# apply nve integration to all particles that are inserted as single particles
fix         integr all nve/sphere

# center of mass
compute     centerOfMass all com

# compute explicit dragforce
compute     explDrag all reduce update_on_run_end yes sum f_dragforce[1]
f_dragforce[2] f_dragforce[3]

# sum of explicit and implicit drag force given from CFD to DEM
variable    totalDragX equal f_cfd2[1]
variable    totalDragY equal f_cfd2[2]
variable    totalDragZ equal f_cfd2[3]

# explicit drag force given from CFD to DEM
variable    explicitDragX equal c_explDrag[1]
variable    explicitDragY equal c_explDrag[2]
variable    explicitDragZ equal c_explDrag[3]

variable    time equal step*dt
fix         extra all print 10 "${time} ${explicitDragX} ${explicitDragY}
${explicitDragZ} ${totalDragX} ${totalDragY} ${totalDragZ}" file
../DEM/post/forces.txt title "# time expDrag(X Y Z) totDrag(X Y Z)"

# screen output
compute         rke all erotate/sphere
thermo_style    custom step atoms ke c_rke vol c_centerOfMass[3] c_explDrag[1]
c_explDrag[2] c_explDrag[3] f_cfd2[1] f_cfd2[2] f_cfd2[3]
thermo          10
thermo_modify   lost ignore norm no
compute_modify  thermo_temp dynamic yes

# dump Particles
dump        dmp1 all custom/vtk 50 ../DEM/post/particles*.vtk id type x y z vx
vy vz fx fy fz radius


# dump Walls
dump        dmpw1 all mesh/gran/VTK 50 ../DEM/post/1.Walls_Spiral*.vtk stress
vel stresscomponents wear Walls_Spiral
dump        dmpw2 all mesh/gran/VTK 50 ../DEM/post/2.Walls_Runner*.vtk stress
vel stresscomponents wear Walls_Runner
dump        dmpw3 all mesh/gran/VTK 50 ../DEM/post/3.Walls_Blades*.vtk stress
vel stresscomponents wear Walls_Blades
dump        dmpw4 all mesh/gran/VTK 50 ../DEM/post/4.Walls_Guide*.vtk stress
vel stresscomponents wear Walls_Guide
dump        dmpw5 all mesh/gran/VTK 50 ../DEM/post/5.Walls_Draft*.vtk stress
vel stresscomponents wear Walls_Draft

run             1
```

ANEXO VII

in.liggghts_restart

```
# Pour granular particles into chute container, then induce flow
log             ../DEM/log.liggghts
thermo_log      ../DEM/post/thermo.txt

atom_style      granular
atom_modify     map array
communicate     single vel yes

boundary        f f f
newton          off

units           si
processors      2 2 4

# read the restart file
read_restart    ../DEM/post/restart/liggghts.restartCFDEM

neighbor        0.1 bin
neigh_modify    delay 0

# Material properties required for granular pair styles

fix         m1 all property/global youngsModulus peratomtype 5.e6
fix         m2 all property/global poissonsRatio peratomtype 0.45
fix         m3 all property/global coefficientRestitution peratomtypepair 1
0.3
fix         m4 all property/global coefficientFriction peratomtypepair 1 0.5
fix         m5 all property/global k_finnie peratomtypepair 1 0.2

# pair style
pair_style  gran model hertz tangential history # Hertzian without cohesion
pair_coeff  * *

# timestep, gravity
timestep    0.0001
fix         gravi all gravity 9.81 vector 0 0 1

# walls
#fix     zwalls1  all wall/gran model hertz tangential history primitive type
1 zplane 0.0
#fix     zwalls2  all wall/gran model hertz tangential history primitive type
1 zplane 0.0553
#fix     cylwalls all wall/gran model hertz tangential history primitive type
1 zcylinder 0.01385 0. 0.
    fix     Walls_Spiral    all mesh/surface/stress file
../STL/Zalls_Spiral.stl type   1  element_exclusion_list read
../STL/ZSpiral.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Runner    all mesh/surface/stress file
../STL/Zalls_Runner.stl type   1  element_exclusion_list read
../STL/ZRunner.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Blades    all mesh/surface/stress file
../STL/Zalls_Blades.stl type   1  element_exclusion_list read
../STL/ZBlades.ex curvature_tolerant yes stress on wear finnie
    fix     Walls_Guide     all mesh/surface/stress file
../STL/Zalls_Guide.stl type    1  element_exclusion_list read ../STL/ZGuide.ex
curvature_tolerant yes stress on wear finnie
    fix     Walls_Draft     all mesh/surface/stress file
../STL/Zalls_Draft.stl type    1  element_exclusion_list read ../STL/ZDraft.ex
curvature_tolerant yes stress on wear finnie

    fix     wall all wall/gran model hertz tangential history mesh n_meshes 5
meshes Walls_Spiral  Walls_Runner Walls_Blades Walls_Guide Walls_Draft
```

```
# cfd coupling
fix         cfd all couple/cfd couple_every 50 mpi
fix         cfd2 all couple/cfd/force/implicit
#fix         cfd2 all couple/cfd/force/accumulator RongDrag 10 1.5e-3
#fix         cfd2 all couple/cfd/force/implicit/accumulated #CrankNicolson 0.5

# apply nve integration to all particles that are inserted as single particles
fix         integr all nve/sphere

# center of mass
compute     centerOfMass all com

# compute explicit dragforce
compute     explDrag all reduce update_on_run_end yes sum f_dragforce[1]
f_dragforce[2] f_dragforce[3]

# sum of explicit and implicit drag force given from CFD to DEM
variable    totalDragX equal f_cfd2[1]
variable    totalDragY equal f_cfd2[2]
variable    totalDragZ equal f_cfd2[3]

# explicit drag force given from CFD to DEM
variable    explicitDragX equal c_explDrag[1]
variable    explicitDragY equal c_explDrag[2]
variable    explicitDragZ equal c_explDrag[3]

variable    time equal step*dt
fix         extra all print 10 "${time} ${explicitDragX} ${explicitDragY}
${explicitDragZ} ${totalDragX} ${totalDragY} ${totalDragZ}" file
../DEM/post/forces.txt title "# time expDrag(X Y Z) totDrag(X Y Z)"

# screen output
compute         rke all erotate/sphere
thermo_style    custom step atoms ke c_rke vol c_centerOfMass[3] c_explDrag[1]
c_explDrag[2] c_explDrag[3] f_cfd2[1] f_cfd2[2] f_cfd2[3]
thermo          10
thermo_modify   lost ignore norm no
compute_modify  thermo_temp dynamic yes

# dump Particles
dump        dmp1 all custom/vtk 50 ../DEM/post/particles*.vtk id type x y z vx
vy vz fx fy fz radius


# dump Walls
dump        dmpw1 all mesh/gran/VTK 50 ../DEM/post/1.Walls_Spiral*.vtk stress
vel stresscomponents wear Walls_Spiral
dump        dmpw2 all mesh/gran/VTK 50 ../DEM/post/2.Walls_Runner*.vtk stress
vel stresscomponents wear Walls_Runner
dump        dmpw3 all mesh/gran/VTK 50 ../DEM/post/3.Walls_Blades*.vtk stress
vel stresscomponents wear Walls_Blades
dump        dmpw4 all mesh/gran/VTK 50 ../DEM/post/4.Walls_Guide*.vtk stress
vel stresscomponents wear Walls_Guide
dump        dmpw5 all mesh/gran/VTK 50 ../DEM/post/5.Walls_Draft*.vtk stress
vel stresscomponents wear Walls_Draft


run             1
```