

SIMULACIÓN DE MODELOS DE PREDICCIÓN DE LA DEGRADACIÓN DEL
CONCRETO REFORZADO EXPUESTO A UN MEDIO MARINO SIMULADO EN
ARQUITECTURAS COMPUTACIONALES BASADAS EN MÚLTIPLES GPU.

FÉLIX ARMANDO MEJÍA CAJICÁ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
MAESTRÍA EN INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA

2018

SIMULACIÓN DE MODELOS DE PREDICCIÓN DE LA DEGRADACIÓN DEL
CONCRETO REFORZADO EXPUESTO A UN MEDIO MARINO SIMULADO EN
ARQUITECTURAS COMPUTACIONALES BASADAS EN MÚLTIPLES GPU.

FÉLIX ARMANDO MEJÍA CAJICÁ

TRABAJO DE GRADO PARA OPTAR EL TÍTULO DE MAGISTER EN
INGENIERÍA DE SISTEMAS E INFORMÁTICA

DIRECTOR

DARÍO YESID PEÑA BALLESTEROS

Ph.D. en Corrosión.

CODIRECTOR

CARLOS JAIME BARRIOS HERNÁNDEZ

Ph.D. en Informática.

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
MAESTRÍA EN INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA

2018

Dedicatoria

A Dios, por permitirme llegar a esta etapa de mi formación profesional. A mi querida madre, por su apoyo incondicional y por demostrarme su gran amor. A mi padre por enseñarme a valorar el conocimiento y el trabajo, aunque no estés presente este momento sería tan especial para ti como para mí. A mi querida esposa Yolanda por su paciencia y por acompañarme en el desarrollo de este proyecto.

Agradecimientos

Doy infinitas gracias a Dios, por haberme dado fuerzas y valor para culminar esta etapa de mi vida.

Agradezco la confianza y el apoyo brindado por parte de mi madre, que siempre demostrándome su infinito amor celebra mis triunfos y me da fuerzas para afrontar estos retos.

A mi querida esposa Yolanda, por su gran apoyo y su amor me impulsó a lograr este triunfo y me alienta a ser cada día mejor.

Al profesor Ph.D. en Corrosión, Darío Yesid Peña por toda la colaboración y el apoyo brindado para el desarrollo del presente proyecto de investigación.

Al profesor Ph.D. en Informática, Carlos Jaime Barrios por cada una de sus valiosas aportaciones que hicieron posible este proyecto de investigación y por la gran calidad humana que me ha demostrado.

CONTENIDO

INTRODUCCIÓN.....	13
1. DESCRIPCIÓN	15
1.1. EL FENÓMENO DE LA CORROSIÓN.....	15
1.1.1. Deterioro del concreto.....	19
1.2. OBJETIVOS	23
1.2.1. Objetivo General.....	23
1.2.2. Objetivos específicos	23
2. PROGRAMACIÓN EN PARALELO	25
2.1. Técnicas de programación en paralelo	27
2.1.1. Método de descomposición de dominio.....	27
2.1.2. Descomposición funcional	28
2.1.3. Descomposición recursiva	29
2.2. Paralelización en GPU	30
2.2.1. Arquitecturas de las GPU	33
2.2.2. GPU Computing acelerado	34
2.2.3. OpenACC	36
2.2.4. OpenMP.....	41
2.2.5. Herramientas de Profiling	41
3. Discretización de ecuaciones diferenciales.....	43
3.1. Método de Diferencias Finitas.....	43
3.1.1. Aproximación a las derivadas en 2D y 3D	46
4. Modelos de predicción de la vida útil del concreto reforzado	49
4.1. Modelos empíricos	49
4.2. Modelos según la comparación de desempeño.....	50
4.3. Modelos estocásticos.....	50
4.4. Modelos matemáticos	51
4.4.1. Modelos tradicionales	53
4.4.2. Modelos Multi-especie	54

5.	METODOLOGIA	56
5.1.	Estudio de la literatura	56
5.2.	Diseño e implementación de la versión secuencial.....	57
5.3.	Optimización de la versión secuencial	57
5.4.	Medición de los tiempos de ejecución.....	58
5.5.	Paralelización de funciones	58
5.6.	Evaluación del rendimiento	58
5.7.	Análisis de resultados	58
5.8.	Documentación	59
6.	Simulación de la predicción de la vida útil del concreto	60
6.1.	Modelo a simular.....	60
6.2.	Algoritmo de Simulación	76
6.2.1.	Diseño del algoritmo de Simulación.....	76
6.2.2.	Implementación del algoritmo de simulación	83
6.2.3.	Pruebas de rendimiento obtenido	94
6.2.4.	Resultados del modelo	96
7.	Conclusiones.....	104
	BIBLIOGRAFÍA.....	105

LISTA DE TABLAS

Tabla 1. Tiempos de ejecución.	95
Tabla 2. Tiempos de ejecución obtenidos en el algoritmo de simulación.	96
Tabla 3. Valores de entrada al algoritmo de simulación.	98
Tabla 4. Valores de entrada segunda simulación.	100
Tabla 5. Valores de entrada para la tercera simulación.	102

LISTA DE FIGURAS

Figura 1. Estrategias de descomposición de dominio de los datos. (a) por columnas (b) por filas (c) por bloques.....	28
Figura 2. Arquitectura básica de una GPU.	34
Figura 3. Series de Taylor.....	44
Figura 4. Modelo de vida útil de Tutti.	51
Figura 5. Fases de desarrollo	56
Figura 6. Cubo de datos de variables.	77
Figura 7. Variables de entrada del concreto.	78
Figura 8. Diagrama de flujo del algoritmo.	80
Figura 9. Formato del archivo de datos de VTK.....	83
Figura 10. Resultados de profiling versión secuencial.	87
Figura 11. Código paralelizado de la función get_multiply.....	89
Figura 12. Código paralelizado de la función get_Irc.....	89
Figura 13. Código paralelizado de la función fd3d_diffusion_2Fick_3D.....	91
Figura 14. Visualización de los tiempos de ejecución mediante PGI Profiler.....	92
Figura 15. Interface Paraview visualización de la difusión del cloruro en el concreto.....	94
Figura 16. Comparación de tiempos de ejecución.	95
Figura 17. Aceleración obtenida para la simulación en GPU.....	97
Figura 18. Concentración de cloruros para $a/c=0.4$	99
Figura 19. Concentración de cloruros para $a/c=0.5$	101
Figura 20. Concentración de cloruros para $a/c=0.6$	103

RESUMEN

TITULO: SIMULACIÓN DE MODELOS DE PREDICCIÓN DE LA DEGRADACIÓN DEL CONCRETO REFORZADO EXPUESTO A UN MEDIO MARINO SIMULADO EN ARQUITECTURAS COMPUTACIONALES BASADAS EN MULTIPLES GPU.*

AUTOR: FÉLIX ARMANDO MEJÍA CAJICÁ**

PALABRAS CLAVES: GPGPU, modelamiento, simulación, corrosión, concreto.

DESCRIPCIÓN: La simulación del proceso de la difusión de cloruros en el concreto reforzado expuesto a medios marinos, permite conocer la concentración del ion cloruro en una estructura de concreto en un tiempo determinado, por lo tanto, es posible estimar el tiempo necesario para que la estructura alcance la concentración umbral y así iniciar la despasivación del acero de refuerzo.

El concreto reforzado es uno de los materiales más utilizados en la construcción, debido a sus propiedades estructurales, bajo costo y gran durabilidad. El carácter alcalino presente en los poros del concreto, permite que el acero de refuerzo en un inicio se encuentre en estado pasivo en cuanto a la corrosión, siendo su velocidad casi nula, sin embargo, esta velocidad comienza a incrementarse debido al ingreso de agentes agresivos los cuales se encuentran presentes en el medio ambiente como son los iones de cloruro y el dióxido de carbono, los cuales llevan al inicio de la corrosión de las estructuras de acero que refuerzan el concreto.

En este trabajo se simula un modelo de difusión teniendo en cuenta los diferentes factores que afectan o inciden en la velocidad de inicio de la corrosión de la estructura de refuerzo, como son la relación agua-cemento, temperatura, densidad y capacidad de fijación del cloruro del concreto, dado que se manejan todas estas variables y una gran cantidad de información se hace necesario utilizar arquitecturas computacionales basadas en múltiples GPU con el fin de lograr la obtención de mejores resultados en tiempos más cortos y así poder minimizar este fenómeno cambiando ciertos parámetros de diseño y fabricación.

* Trabajo de grado

** Facultad de Ingenierías Físicomecánicas. Escuela de Ingeniería de Sistemas e Informática. Maestría en Ingeniería de Sistemas e Informática. Director: Darío Yesid Peña Ballesteros. Ph.D. Corrosión.

ABSTRACT

TITULO: SIMULATION OF PREDICTION MODELS OF THE DEGRADATION OF REINFORCED CONCRETE EXPOSED TO A MARINE ENVIRONMENT THROUGH GPU BASED ARCHITECTURES. *

AUTOR: FÉLIX ARMANDO MEJÍA CAJICÁ**

PALABRAS CLAVES: GPGPU, modeling, simulation, corrosion, concrete.

DESCRIPCIÓN: The simulation of the chlorides diffusion process in reinforced concrete exposed to marine environments, allows knowing the concentration of the chloride ion in a concrete structure in a determined time, therefore, it is possible to estimate the time necessary for the structure to reach the threshold concentration and thus initiate the depassivation of the reinforcing steel.

Reinforced concrete is one of the materials most used in construction, due to its structural properties, low cost and great durability. The alkaline character present in the pores of the concrete, allows the steel reinforcement passivate in terms of corrosion, with its velocity almost null, but this velocity begins to increase due to the entrance of aggressive agents that are present in the environment such as chloride ions and carbon dioxide, which lead to the beginning of corrosion of steel structures that reinforce concrete.

In this work a diffusion model is simulated taking into account the different factors that affect the corrosion initiation rate of the reinforcement structure, such as the water-cement ratio, temperature, density and chloride binding capacity of concrete, since all these variables are handled and a great amount of information becomes necessary to use computational architectures based on multiple GPUs in order to obtain better results in shorter times and thus be able to minimize this phenomenon by changing certain parameters of Design and manufacturing.

* Trabajo de grado

** Facultad de Ingenierías Físicomecánicas. Escuela de Ingeniería de Sistemas e Informática. Maestría en Ingeniería de Sistemas e Informática. Director: Darío Yesid Peña Ballesteros. Ph.D. Corrosion.

INTRODUCCIÓN

La necesidad de tener máquinas cada vez más rápidas se ve como consecuencia del surgimiento de aplicaciones más complejas que necesitan mucha capacidad de cómputo, pero hay que tener en cuenta que hay un límite al cual se va a llegar, por el cual nada puede moverse más rápido que la velocidad de la luz. Las computadoras monoprocesador han logrado grandes velocidades de varios miles de millones de instrucciones por segundo por eso la tecnología hardware se ha visto llevada a los límites físicos que limitan el poder calculador que puede ser logrado por una computadora de este tipo.

Una manera de evitar esta limitante es utilizar el cómputo paralelo aprovechando el uso de las unidades de procesamiento gráfico GPU. Para superar el punto de saturación de los requerimientos del cómputo realizable por una computadora serial, se conecta un número grande de computadores los cuales son programados para trabajar simultáneamente en un problema dado, así puede proporcionar rendimientos muchos más altos del cómputo que las computadoras seriales más rápidas.

Las computadoras de alto rendimiento ahora son esenciales en la investigación y el desarrollo científico y tecnológico, con su capacidad de proceso de alta velocidad, memoria extensa, las computadoras ahora son herramientas importantes para los experimentos de simulación y para el procesamiento de grandes volúmenes de información. Los progresos futuros en superordenadores de alta velocidad desempeñarán un papel significativo en la investigación y el desarrollo de la tecnología avanzada del presente siglo.

La simulación de problemas fisicoquímicos lleva inherente el análisis y procesamiento de gran cantidad de información, para esto se requiere una gran capacidad de cómputo por esto se hace necesario la aplicación de un esquema de procesamiento paralelo utilizando las GPU, que permita mejorar eficientemente la obtención de los resultados de la simulación.

La corrosión de las estructuras de acero que forman parte del concreto reforzado se origina por la presencia de oxígeno y humedad cerca a éstas, además de la existencia de cloruros libres en el medio que las rodea. En el agua de mar se encuentra un alto nivel de concentración de sales disueltas, debido a esto representa un elemento ofensivo y peligroso para el concreto reforzado debido a que propicia y acelera el fenómeno de la corrosión (Li L, 2005). En las costas la brisa marina lleva consigo importantes contenidos de humedad que de una u otra forma lleva cloruros, por lo cual estructuras que no están en contacto directo con el mar comienzan a sufrir por la acción de los cloruros.

Los cloruros son un elemento activo en el proceso de deterioro y degradación de las estructuras de concreto en las áreas marítimas, de acuerdo a los niveles de concentración que se encuentren en el agua de mar, el cual determina su nivel de agresividad (Vargas).

En este trabajo se desarrolló una simulación del proceso de la corrosión en el concreto reforzado expuesto a ambientes marinos, donde los cloruros y la carbonatación son los procesos que degradan el material, para esto se va a utilizar las técnicas de paralelización de los algoritmos sobre arquitecturas basadas en múltiples GPU, con el propósito de obtener una mayor resolución del problema y así poder utilizar mucha más información y así acelerar su procesamiento.

1. DESCRIPCIÓN

1.1. EL FENÓMENO DE LA CORROSIÓN

El fenómeno de la corrosión del acero de refuerzo es una de las causas más frecuentes del deterioro prematuro de las estructuras de concreto, aun sabiendo que el concreto por su alta alcalinidad con un promedio de pH de 12.5 y baja conductividad, debería ser un medio que proporciona buena protección al acero contra la corrosión, pero al encontrarse en un medio bastante agresivo esta protección no es suficiente por lo cual el fenómeno se produce, además de las condiciones que de origen la favorecen como son la excesiva porosidad del concreto, reducido espesor de recubrimiento del concreto sobre el refuerzo, la aparición de grietas en la estructura y la alta concentración de agentes corrosivos en los componentes del concreto (Vargas).

El concreto reforzado es uno de los materiales más utilizados en las construcciones actualmente, debido a sus propiedades estructurales, bajo costo y a su prolongada durabilidad. El carácter alcalino presente en los poros del concreto, permite que el acero de refuerzo en un inicio se encuentre en estado pasivo respecto a la corrosión, siendo su velocidad casi nula, sin embargo, esta velocidad puede irse incrementando debido al ingreso de ciertos agentes agresivos presentes en el medio ambiente como son los iones de cloruro, sulfatos y el dióxido de carbono, los cuales causan el inicio de la corrosión de las estructuras de acero (Selouma, 2009).

En un ambiente marino la principal causa de deterioro de las estructuras de concreto reforzado es la corrosión iniciado por iones de cloruro. Estas estructuras al estar expuestas al agua de mar, la brisa marina o el empleo de agregados contaminados con sales, lleva a la incorporación de éstos iones al concreto reforzado, los cuales

una vez lleguen a un nivel de concentración crítico en la superficie del acero, la capa pasiva de protección pierde la estabilidad y da inicio a la corrosión. El cloruro se puede incorporar al concreto reforzado ya sea durante la etapa de elaboración mediante alguno de sus componentes o mediante procesos como el de difusión, permeabilidad o capilaridad cuando éste se encuentra expuesto a atmósferas, aguas o suelos húmedos que contienen cloruros. Para mitigar esta penetración ya sea por cualquiera de los procesos anteriormente nombrados debe procurarse una estructura poco porosa para que así minimice la velocidad de ingreso, ya que la porosidad es uno de los principales factores que inciden en el inicio de la corrosión.

Entre las principales deficiencias que presenta el concreto están su baja resistencia a la tracción y la porosidad, la cual se debe a su propia naturaleza ya que es el producto de los procesos de hidratación de la reacción entre cemento y agua, la cual una vez endurecido el concreto los restos de agua se convertirán en poros de diferentes tamaños, los cuales en su gran mayoría serán poros capilares los cuales estarán interconectados entre sí, ya sea directamente o mediante poros de gel. Al tener un gran volumen de poros determinará una mayor capacidad de transporte de fluidos como el agua, gases y otras sustancias químicas las cuales son potencialmente perjudiciales para el concreto reforzado ya que puede iniciarse un proceso más o menos lento de deterioro de las estructuras de acero que lo conforman.

Esta simulación permite analizar y experimentar que ocurre al interior del concreto reforzado, logrando así un aprendizaje activo permitiendo verificar como se puede disminuir el fenómeno de deterioro mediante el cambio de ciertos parámetros de diseño y fabricación. La utilización de arquitecturas de computación basadas en GPU, permitió utilizar muchas más variables que intervienen en este fenómeno y por ende más información sin ver afectados los tiempos de respuesta, esto debido

a la alta capacidad y velocidad de procesamiento que se puede lograr de éstas arquitecturas y así determinar el tiempo de vida útil del concreto reforzado con una mayor exactitud y en el menor tiempo posible, con el fin de tomar decisiones oportunas en cuanto al mantenimiento, reparación o reemplazo estructural, siendo así una herramienta valiosa para la industria de la construcción ya que permitirá controlar el fenómeno de la corrosión y no llegar a situaciones irreparables.

Corrosión: es la reacción química producto de la unión del metal con el oxígeno, es decir, la corrosión es un deterioro observado en un objeto metálico a causa de un alto impacto electroquímico de carácter oxidativo y la velocidad degenerativa de dicho material dependerá de la exposición al agente oxidante, la temperatura presentada si se encuentra expuesto a soluciones salinizadas y de las propiedades químicas que presentan estos agentes metálicos, este proceso es espontáneo y natural y se puede presentar en materiales que no sean metálicos (Valderrama.).

Concreto reforzado: también llamado hormigón armado consiste en la utilización del concreto reforzado con barras o mallas de acero principalmente, aunque también se utiliza fibras plásticas, fibra de vidrio, fibras de acero o combinaciones de barras de acero con fibras dependiendo de los requerimientos a los que va estar sometido. Se utiliza en todo tipo de construcciones como son edificaciones, puentes, presas, túneles y obras industriales (Velázquez Gonzalez, 2005).

Concreto: es una mezcla de cemento, grava, arena, aditivos y agua, maleable en su forma líquida y de gran resistencia en su estado sólido. Los aditivos se utilizan con el fin de adicionarle otras propiedades al concreto. El concreto es de apariencia densa, pétreo compacta y porosa por lo cual permite la entrada de líquidos y gases que provienen del exterior.

Cemento: es un material pulverizado que contiene óxido de calcio, sílice, alúmina, óxido de hierro y otros componentes menores que al adicionarle cierta cantidad de agua forma una pasta conglomerante capaz de endurecer tanto en el agua como al

aire, dando lugar a productos hidratados mecánicamente resistentes y estables (Mejía de Gutiérrez, 1999).

El cemento Portland es una mezcla de piedra caliza y arcilla, previamente triturada y sometida a cocción hasta la sinterización a una temperatura alrededor de 1482°C, la cual se le llama Clinker, la cual posteriormente se pulveriza jun con el yeso, material regulador del fraguado, para formar un polvo fino. Los principales compuestos del Clinker son los silicatos, los cuales constituyen el 60 al 80% del peso del cemento portland y son los responsables de su resistencia mecánica.

Agua: ésta al utilizarse en la mezcla influye significativamente en el comportamiento y propiedades del concreto. El agua debe estar exenta de impurezas tales como aceite, ácidos, álcalis, sales, sedimentos, materia orgánica u otras sustancias perjudiciales.

Aditivos: se definen como sustancias que son incorporadas al concreto durante el mezclado con el fin de mejorar sus propiedades como son:

- Retardar o acelerar el tiempo de fraguado inicial.
- Incrementar la trabajabilidad sin aumentar o disminuir el contenido de agua
- Modificar la rapidez o la capacidad de sangrado
- Reducir la exudación y segregación
- Modificar las propiedades del concreto endurecido (aumentar resistencia y durabilidad)

Durabilidad del concreto: se define como la capacidad para resistir a la intemperie, ataque químico, la abrasión o cualquier otro proceso o condición de servicio de las estructuras que produzcan deterioro del concreto. Existen diversos factores que afectan la durabilidad del concreto reforzado como son la calidad de sus componentes, dosificación de cada uno de sus componentes, mezclado durante el tiempo suficiente con el fin de obtener un material homogéneo, colocación correcta

de las estructuras de acero, compactación adecuada para evitar segregación y porosidad (Mejía de Gutiérrez, 1999).

1.1.1. Deterioro del concreto

Existen factores físicos, químicos o fisicoquímicos que conducen a la corrosión de las armaduras. En cualquiera de estas categorías se reconoce la influencia de los componentes del concreto y su ubicación geográfica sobre la vida útil de la estructura. Los mecanismos físicos de deterioro se asocian con la disolución de compuestos de la pasta en el medio, con pérdida de masa, aumento de porosidad y caída de resistencia. Los mecanismos químicos corresponden al intercambio de iones de la pasta con el medio dando lugar a compuestos solubles o no, pero de carácter no expansivo que finalmente causan efectos similares a los físicos. El fisicoquímico combina los dos conceptos, dando lugar a la formación de compuestos de tipo expansivo que provoca tensiones internas y que llevan a la fisuración y posible desintegración del componente. Dentro de los diferentes mecanismos de deterioro, se puede considerar relevantes los siguientes:

- Corrosión de la armadura
- Reacción árido – álcali
- Ataques químicos por cloruros, sulfatos o carbonatación
- Ciclos de hielo – deshielo (Mejía de Gutiérrez, 1999)

Ataque por cloruros

El cloruro puede penetrar el concreto reforzado desde el medio externo hacia su interior generando corrosión en las armaduras, mediante la combinación de varios mecanismos de transporte como son:

- Difusión iónica, debido al gradiente de concentración entre la superficie expuesta y el líquido de poro en la matriz cementícea, siempre que exista humedad dentro de la estructura de poro.

- La absorción de agua con cloruros en el concreto parcialmente saturado, absorción capilar.
- Por flujo de agua en el concreto debido a un gradiente de presión.
- Por dispersión de iones cloruro debido a efectos convectivos de la solución de poro.
- Por efecto de un potencial eléctrico externo y/o intrínseco.

Los iones cloruros llegan al concreto por medio de agregados contaminados, aguas, cementos y aditivos durante el amasado; también pueden penetrar desde el exterior al concreto endurecido a través de la red de poros como consecuencia de ambientes marinos, donde los iones cloruros están presentes en el agua de mar y es posible que el viento de la brisa marina los desplace a la zona costera y los deposite en estructuras de concreto cercanas a la línea de mar. También es posible la existencia de cloruros en el agua de amasado lo cual aumenta el alto riesgo de corrosión.

Cuando los cloruros penetran la estructura desde el exterior, lo hace a través de los poros del concreto, al avanzar una cantidad reacciona, otra se absorbe y otra queda disuelta.

Cloruro libre: es el que permanece disuelto en el agua que contiene los poros. Se desplaza por el interior del concreto mediante difusión u otros mecanismos.

Cloruro enlazado: es el que permanece unido a las fases sólidas del concreto. Este cloruro queda inmovilizado, sin posibilidad de moverse hacia el interior.

Cloruro total: Es la cantidad total de cloruro que contiene el concreto, sin hacer distinción entre el cloruro libre y enlazado.

Los iones cloruro que ocasionan daño al acero de refuerzo son los que se hallan disueltos o libres, aunque es posible que debido a los equilibrios que se presentan los absorbidos se incorporen a la disolución y se tornen peligrosos.

Solución poro concreto

La sinergia del concreto con el acero de refuerzo se basa en que el concreto provee al refuerzo de una protección tanto química como física contra la corrosión. La protección química se debe a la alcalinidad del concreto, la cual produce una capa de óxido en la superficie del acero impidiendo que este continúe corroyéndose. A este fenómeno se le conoce como pasividad, (Fontana, 1986) ya que la capa de óxido evita la propagación de la corrosión del acero. La alcalinidad del concreto se debe principalmente al hidróxido de calcio Ca(OH)_2 que se forma durante la hidratación de los silicatos del cemento y a los álcalis (sodio y potasio) que pueden estar incorporados como sulfatos en el Clinker (Mindess, 1981). Estas sustancias sitúan el pH de la fase acuosa contenida en los poros en valores entre 12 y 14, es decir en el extremo más alto de la escala de pH. El concreto también funciona como una capa física protectora contra los agentes ambientales (oxígeno, agua, cloruros, dióxido de carbono) que pueden despasivar el acero e iniciar su corrosión.

Difusión y absorción

El anión cloruro entra como ión a la estructura a través de los poros del concreto por un mecanismo de difusión pura o combinada con el transporte del agua en la forma de succión capilar y evaporación ubicándose en los poros interiores del concreto, parcial o totalmente llenos de agua, donde parte de estos iones son enlazados por los componentes del cemento mediante mecanismos químicos y físicos. El contenido del anión cloruro y su profundidad de penetración son funciones de algunos factores como son la temperatura, la relación agua/cemento, tipo de cemento, adiciones activas o no presentes (UIS, 2009).

Un factor crucial de estos fenómenos es sin duda la porosidad del concreto, ya que la penetración será más rápida a mayor porosidad, teniendo en cuenta que a relaciones bajas de agua/cemento es posible lograr velocidades de penetración de cloruros más lentas. La penetración del ion cloruro al interior del concreto no es solo función de la porosidad total del material sino también depende del diámetro

de los poros, su distribución y continuidad de los mismos. El tipo de cemento y su capacidad de retener o combinar cloruros es otro de los parámetros relevantes.

Coeficiente de difusión

El coeficiente de difusión de cloruro es altamente dependiente de la composición del concreto; su determinación experimental se realiza por métodos convencionales (ensayos de celda de difusión o ensayos de inmersión). En las pruebas de inmersión las muestras de concreto se mantienen sumergidas en una solución con cloruro, una vez retirada la muestra de la solución se procede a medir el ingreso del ión cloruro a ciertos periodos de tiempo, realizando un posterior análisis químico a la muestra de concreto pulverizada. El coeficiente de difusión aparente puede obtenerse mediante métodos basados en la segunda ley de Fick sobre los perfiles de cloruro obtenidos. También es posible determinar el coeficiente a partir de pruebas de migración, aplicando un campo eléctrico, los cuales consumen menor tiempo (Böhni, 2005).

Capacidad de fijación de cloro de concreto

Se entiende por fijación del cloro la capacidad del concreto de retener en la fase sólida los iones Cl^- libres presentes en la solución poro hasta que, bajo ciertas condiciones, la concentración de Cl^- libres y Cl^- fijos alcancen un equilibrio. El proceso de fijación de cloro ayuda a retardar los efectos de corrosión del acero de refuerzo al disminuir la concentración de Cl^- libre que difunden por el concreto (Ferreira, 2004).

Las isotermas de fijación del Cl^- describen la relación entre los Cl^- libres (C_f) y los Cl^- fijos (C_b) a una temperatura dada y dependen de las características del concreto como su composición, relación agua cemento y pH. (Qiang, 2009) (Pérez B., 2000)

Carbonatación del concreto

El concreto es un material muy poroso, el cual permite la penetración en su interior del CO_2 del aire, a través de los poros. Cuando esto sucede, se produce la reacción CO_2 con el hidróxido de calcio del concreto y los compuestos hidratados del cemento, formándose carbonato de calcio. Una vez todo el $\text{Ca}(\text{OH})_2$, KOH y NaOH presentes en los poros han sido carbonatados, el valor de pH empieza a disminuir, llegando a valores de pH menores a 9. (Del Valle Moreno, y otros, 2001)

1.2. OBJETIVOS

1.2.1. Objetivo General

Desarrollar e implementar una simulación de modelos de predicción de la difusión de cloruros y carbonatos en concreto reforzado, utilizando arquitecturas con múltiples GPU para acelerar eficientemente el procesamiento.

1.2.2. Objetivos específicos

- Analizar los modelos de predicción aplicado a la difusión de las especies iónicas en el concreto reforzado, para identificar oportunidades de concurrencia y paralelismo.
- Establecer mecanismos de implementación en arquitecturas basadas en múltiples GPU, para garantizar eficiencia computacional de la simulación de la corrosión del concreto reforzado.
- Evaluar el rendimiento computacional para identificar métricas, aceleración, escalabilidad, consumo energético y otras, que permitan

definir la eficiencia computacional de la simulación del proceso corrosivo en el concreto reforzado.

- Determinar el tiempo de vida útil del concreto reforzado a un medio marino simulado por medio del modelo de predicción de mejor desempeño.

2. PROGRAMACIÓN EN PARALELO

Cada día se le exige más y más al cómputo moderno, las investigaciones actuales requieren de mucho más poder de procesamiento, y a pesar del avanzado desarrollo de microprocesadores cada vez más rápidos, muchas veces el tener sistemas más rápidos no es muy accesible/conveniente; pensando en esto, el tener varios equipos encargados de la misma tarea puede resultar en sistemas mucho más poderosos y de menor costo.

La programación en paralelo es un paradigma de programación orientado al cómputo paralelo, es decir el correr partes de un mismo programa en distintos procesadores para lograr una mayor eficiencia. Cabe notar que la programación en paralelo difiere de la programación tradicional (o programación serial) enormemente, ya que además del código propio de problema, es necesario incorporar el código necesario para la comunicación entre los procesadores. Es una técnica de programación basada en la ejecución simultánea, bien sea en un mismo ordenador (con uno o varios procesadores) o en un clúster de ordenadores, en cuyo caso se denomina computación distribuida. Al contrario que en la programación concurrente, esta técnica enfatiza la verdadera simultaneidad en el tiempo de la ejecución de las tareas.

Los sistemas con multiprocesador y multicomputadores consiguen un aumento del rendimiento si se utilizan estas técnicas. En los sistemas monoprocesador el beneficio en rendimiento no es tan evidente, ya que la CPU es compartida por múltiples procesos en el tiempo, lo que se denomina multiplexación o multiprogramación. El mayor problema de la computación paralela radica en la complejidad de sincronizar unas tareas con otras, ya sea mediante secciones críticas, semáforos o paso de mensajes, para garantizar la exclusión mutua en las

zonas del código en las que sea necesario (Serrano, y otros, 2014). Es conveniente implementar una solución paralela cuando el problema en cuestión se presta a ser dividido en partes independientes o semi-independientes entre sí, es un error pensar que cualquier programa sería mejor implementarlo en paralelo, ya que hay que recordar que existe un costo de comunicación entre los procesadores.

En el campo de la computación de alto rendimiento se usa una terminología la cual es necesario tener muy clara y entender de qué se trata, entre estos conceptos tenemos:

- IPS (Instructions per second): se refiere al número de instrucciones por segundo que ejecuta un sistema de cómputo.
- FLOPS (Floating point operations per second): número de operaciones de coma flotante por Segundo.
- Operaciones de coma flotante: operaciones realizadas con número en notación científica.
- Núcleo: en computación se refiere a una unidad de procesamiento independiente, la cual puede ejecutar una serie de instrucciones y entregar un resultado.
- Pipelines (tubería): es un conjunto de elementos de procesamiento de datos conectados en serie, donde la salida de un elemento es la entrada del siguiente.
- Algoritmo secuencial: algoritmo que se ejecuta en un solo procesador y se ejecuta una sola instrucción cada vez.
- Operación aritmética: operación binaria que se ejecuta en el procesador e incluye las cuatro operaciones básicas como son suma, resta, multiplicación y división.
- Operación lógica: operación binaria que ejecuta el procesador e incluye las operaciones lógicas AND, OR, NOT entre otros.

- Acceso a memoria: operación que realiza el procesador y que consiste en acceder a un bloque de memoria para realizar una operación de lectura o escritura.

2.1. Técnicas de programación en paralelo

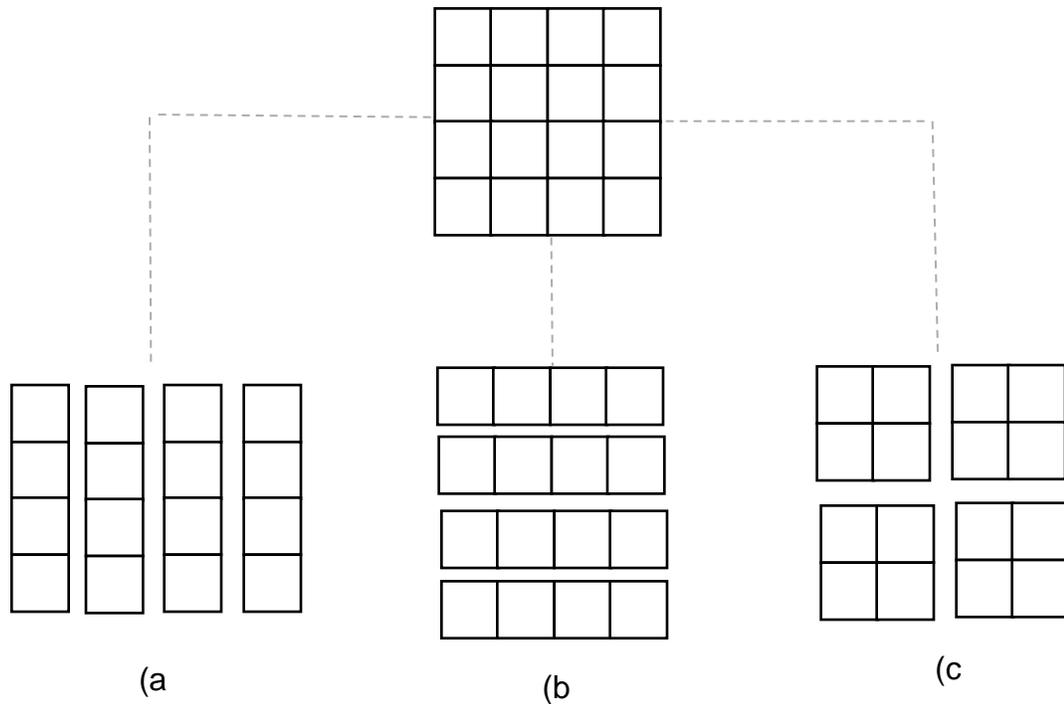
Debido al incremento en el volumen y resolución de los datos, la complejidad de los algoritmos de procesamiento se hace necesario aplicar técnicas o métodos que permiten explotar al máximo la computación paralela los cuales se describen a continuación.

2.1.1. Método de descomposición de dominio

Mediante este método se puede tratar problemas de tamaño considerable, aplicando las técnicas de paralelización. Este método se basa en que dado un dominio computacional Ω , este se puede particionar en subdominios Ω_i , donde $i = 1, 2, 3, \dots$, entre los cuales puede existir o no existir traslape. Por lo cual el problema es llevado a términos de cada subdominio, obteniendo así una serie de subproblemas de tamaño reducido independientes en principio entre sí, los cuales son asignados a cada procesador y una vez obtenidas las soluciones parciales solo hay que combinarlas para obtener la solución total del problema original. El equilibrio entre los procesadores se logra si se particionan los datos en subdominios del mismo tamaño de manera que haya uniformidad en la distribución y que no haya procesadores que operen con mayor cantidad de datos que otros. (ARMSTRONG, 1992)

En los algoritmos paralelos los datos se descomponen ya sea por columnas, descomposición por filas o descomposición por bloques como se muestra en la figura 1.

Figura 1. Estrategias de descomposición de dominio de los datos. (a) por columnas (b) por filas (c) por bloques



2.1.2. Descomposición funcional

Algunas veces la descomposición de dominio puede no ser lo más eficiente cuando las partes asignadas a los diferentes procesadores pueden requerir diferentes tiempos de procesamiento unos de otros significativamente, por lo cual el tiempo de ejecución viene determinado por el proceso que más tarde en terminar, mientras los otros procesadores permanecen ociosos un tiempo. La descomposición funcional también es conocida como paralelismo de tareas, en el cual el problema se divide en un gran número de partes más pequeñas y las subtareas son asignadas a los

procesadores disponibles. Tan pronto un procesador termina una subtarea, recibe otra hasta que se ejecuten todas. Este paralelismo se implementa sobre el paradigma maestro-esclavo, en donde el proceso maestro se encarga de ir asignando las subtareas a los procesos esclavos, recibiendo los resultados obtenidos y asignando las tareas faltantes (FOSTER, 2003).

2.1.3. Descomposición recursiva

Técnica secuencial divide-y-vencerás para descubrir concurrencia. El seudocódigo del Algoritmo refleja la estructura general de un algoritmo recursivo que utiliza la descomposición de dominio (Brewer., 1994).

Calcule_Divide(p: problema, n:tamaño): solución

Si (n es suficientemente pequeño) entonces

 Devolver solución de p por método directo

Sino

 Dividir p en subproblemas mas pequeños p_1, p_2, \dots, p_k

 Para (i=1 to k) hacer

$S_i = \text{Calcule_Divide}(p_i, n_i)$

 Fin Para

 Devolver_Combinar(S_1, S_2, \dots, S_k)

Fin Si

Fin Calcule_Divide

Para aplicar la técnica de descomposición recursiva se debe cumplir:

- El problema original debe poderse dividir fácilmente en un conjunto de subproblemas del mismo tipo que el problema original, pero con una resolución más pequeña.
- La solución de un subproblema debe obtenerse independientemente de la de los demás.
- Se necesita un método directo para la resolución de problemas de tamaños pequeños.
- Es necesario tener un método que combine las soluciones parciales hasta obtener la solución del problema original.

2.2. Paralelización en GPU

En las últimas dos décadas, la GPU ha tenido enormes avances en cuanto al rendimiento y capacidad, explotando al máximo el paralelismo masivo inherente en el procesamiento de gráficos. Avanzando de un pipeline de gráficos de función fija a un procesador paralelo programable para gráficos avanzados en 3D. El bajo costo y marcado incremento del rendimiento, permitió la incorporación en el mercado. Dejo de ser un procesador gráfico para convertirse en un co-procesador apto para el desarrollo de aplicaciones paralelas de propósito general con demanda de anchos de banda de procesamiento y de memoria sustancialmente superiores a los ofrecidos por la CPU. (XATAKA, 2014)

Una GPU o Graphic Processing Unit es un dispositivo de co-procesamiento gráfico el cual hoy en día es bastante habitual en los computadores personales, que está muy asociado a la ejecución de videojuegos de forma óptima. En los últimos años con el surgimiento de GPU cada vez más rápidas y con mayor capacidad de computo, ha surgido la computación de propósito general sobre unidades de

procesamiento gráfico con sus siglas en inglés GPGPU (General-Purpose computing Graphic Processing Units).

El exitoso mercado de las tarjetas gráficas impulsado por el sector de los videojuegos en el PC y consolas, han llevado a que los procesadores gráficos utilizados en las tarjeta gráficas han evolucionado desde ser unos coprocesadores que simplemente representaban caracteres o gráficos de mapas de bits, a ser complejos procesadores capaces de realizar complejos cálculos matemáticos y vectoriales sobre cada pixel para poder dar a las imágenes un mayor realismo y a su vez una mayor tasa de cuadros por segundo para suavizar las transiciones entre cada instantánea. Es así como el poder computacional de las GPU se ha multiplicado mucho más en estos últimos años que el de los procesadores de propósito general, CPU. Una tarjeta gráfica de gama alta puede tener fácilmente más 2800 procesadores individuales los cuales trabajan de forma independiente en paralelo y con capacidad de comunicarse a través de memorias rápidas DDR5 de varios GigaBytes de capacidad ubicadas en la misma tarjeta.

Debido al gran número de unidades aritmético-lógicas presentes en las unidades de procesamiento gráfico GPU, que hasta ahora se venían utilizando para el procesado de gran cantidad de vértices para la representación de figuras en tres dimensiones, los programadores vieron en estas plataformas masivamente paralelas que podían ser utilizadas para procesar grandes cantidades de otros tipos de datos. Es así como las empresas fabricantes de GPU como NVidia y ATI empezaron a desarrollar herramientas para abrir el acceso a sus dispositivos para los programadores que estuvieran interesados en utilizar este gran potencial incorporado.

NVidia lanzó al mercado en Febrero de 2007 su primera versión de su API GPGPU Compute Unified Device Architecture, más conocido como CUDA. La cual se ha

convertido en la base sobre la cual funcionan muchas de las aplicaciones científicas, de encoding de videos, entre otras como la aplicación de físicas aceleradas por GPU PhysX (Bazant, 1979). Cuda es un API que ofrece un entorno programación de bajo nivel, lo cual implica mayor conocimiento de parte de los programadores respecto al funcionamiento específico del chip, por lo cual ha sido poco difundido (NVIDIA, 2017).

En la actualidad miles de desarrolladores, científicos, estudiantes, creadores de videojuegos e investigadores desarrollan aplicaciones aprovechando el alto rendimiento de computación que ofrecen las GPU, en áreas bastantes diversas como son los juegos basados en la física, análisis de riesgos de activos, análisis de datos sísmicos y pronósticos climáticos, entre las que se encuentran SeismicCity, TechniScan, Badaboom, etc.

En cuanto a la ciencia de materiales existen varias aplicaciones que aprovechan la potencia de cálculo de las GPU, entre las que se destacan gWL-LSMS que permite investigar los efectos de la temperatura en el magnetismo de los materiales basado en el método de Wang-Landau, PEtot que calcula el comportamiento de las estructuras de electrones de los materiales, mediante cálculos pseudopotenciales de la onda plana de la teoría funcional de la densidad (DFT), QMCPACK que resuelve la ecuación de Schrodinger de muchos cuerpos para estructuras electrónicas usando el método cuántico de Monte Carlo, Quantum Espresso/PWcsf, que es un conjunto de aplicaciones integradas para el cálculo de las estructuras de electrones y el modelado de materiales a nanoescala incluyendo algebra lineal y FFTs 3D, y VASP calcula el comportamiento de las estructuras electrónicas basadas en la teoría cuántica. (NVIDIA, 2016)

El Dr. Joshua A. Anderson, especialista en el área de investigación de la Universidad de Michigan, es uno de los primeros investigadores en usar la

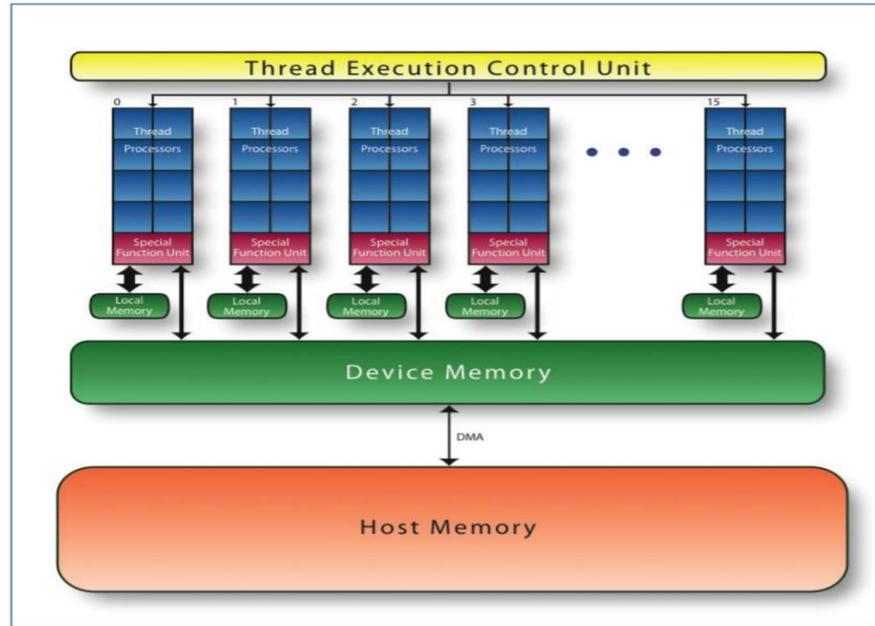
tecnología de computación basada en GPU, comenzó desde los primeros días de CUDA sobre una GTX 8800, en el año 2006, haciendo simulaciones de dinámica molecular. Desde el año 2015 se encuentra trabajando en un proyecto de simulación de materiales para descubrir los principios fundamentales de como los sistemas a nanoescala de bloques de construcción se auto ensamblan y así descubrir cómo controlar este proceso para diseñar nuevos materiales. Se concentra en el papel de como la forma de las partículas y el cambio de las mismas lleva a obtener diferentes propiedades del material. (NVIDIA, 2016)

Actualmente existe un estándar llamado OpenACC promovido por las principales compañías del sector de las GPU, que permite aprovechar el rendimiento que se obtiene de las GPU, sin la necesidad de aprender lenguajes específicos ligados al fabricante, permitiendo a los programadores obtener resultados bastantes rápidos reduciendo así los tiempos necesarios en los proyectos de investigación.

2.2.1. Arquitecturas de las GPU

Las GPU manejan una arquitectura basada en múltiples procesadores que se encargan de manejar los datos de forma paralela, el cual es mostrado en la figura 2.

Figura 2. Arquitectura básica de una GPU.



<http://www.nvidia.es/object/gpu-computing-es.html>

Como se puede ver en la figura la GPU tiene una unidad de control de ejecución de hilos que se encarga de recibir los datos que va a manejar y los entrega a cada uno de los procesadores que a su vez tienen varios procesadores de hilos. Cada procesador de hilo ejecuta el mismo algoritmo con diferentes datos de manera que tenemos trabajando en paralelo a nivel de programa ya que cada procesador puede estar realizando una o varias tareas diferentes y paralelismo a nivel de datos porque se trabaja el mismo algoritmo con diferentes datos al mismo tiempo (NVIDIA, 2017).

2.2.2. GPU Computing acelerado

El cálculo acelerado en la GPU puede definirse como el uso de una unidad de procesamiento gráfico (GPU) en combinación con una CPU para acelerar aplicaciones de deep learning, análisis e ingeniería. NVIDIA lo introdujo en 2007 y,

desde entonces, las GPU aceleradoras han pasado a instalarse en centros de datos energéticamente eficientes de laboratorios gubernamentales, universidades, grandes compañías y PYMEs de todo el mundo. Desempeñan un papel fundamental en la aceleración de aplicaciones que abarcan desde la inteligencia artificial hasta los coches, drones y robots. El cálculo acelerado en la GPU traslada las partes de la aplicación con mayor carga computacional a la GPU y deja el resto del código ejecutándose en la CPU. Desde la perspectiva del usuario, las aplicaciones simplemente se ejecutan más rápido. (NVIDIA, 2017)

Una forma sencilla de entender la diferencia entre la GPU y la CPU es comparar la forma en que procesan las tareas. Una CPU está formada por varios núcleos optimizados para el procesamiento en serie, mientras que una GPU consta de millares de núcleos más pequeños y eficientes diseñados para manejar múltiples tareas simultáneamente.

El grupo de investigación SPHysics está estudiando la propagación de las olas y su interacción con las estructuras costeras, utilizando el modelo SPH (Smoothed Particle Hydrodynamics) hidrodinámica de partículas suavizadas, el cual requiere una carga computacional muy elevada, dado que el tiempo de ejecución crece exponencialmente con el número de partículas se utilizaron técnicas de programación sobre GPUs con el propósito de que la simulación fuera bastante eficiente.

PyFR es un algoritmo o plataforma open source para CFD (Dinámica de fluidos por computadora), este programa utiliza métodos seguros conocidos como la técnica de reconstrucción Flux, para solucionar algunos de los problemas más representativos en fluidos, en particular los de turbulencia y flujos inestables. El líder aeroespacial BAE Systems en el Reino Unido ha colaborado de cerca en el proyecto, “consideramos que la precisión y desempeño numérico de PyFR será un

benchmark para otros códigos” comenta Oscar Neilson de BAE Sytems, un usuario del programa CFD.

En un mercado tan competitivo, es fundamental acortar los ciclos de diseño para sacar los productos al mercado en menos tiempo. Los prototipos digitales permiten a los diseñadores e ingenieros iterar y visualizar sus diseños para implicar mucho antes a los clientes en el ciclo de desarrollo y mejorar así la confianza en el producto.

Los sistemas multi-GPU aprovechan la combinación de múltiples GPU para acortar los tiempos de las simulaciones y eliminar cualquier obstáculo que ralentice la productividad. La aceleración de los procesos en la GPU ha beneficiado a aplicaciones CAE de empresas como ANSYS, Altair Engineering, Autodesk, FluiDyna, IMPETUS, MSC, Prometech, SIMULIA y Vrtis, entre otras. Ahora, los ingenieros consiguen velocidades de simulación muy superiores en sus sistemas y generan los modelos con una fidelidad antes imposible, de manera que pueden producir más variaciones del diseño en menos tiempo. (NVIDIA, 2017)

2.2.3. *OpenACC*

Es un estándar promovido por las principales compañías del sector de las unidades centrales de procesamiento GPU, que permite a los programadores de una forma más fácil aprovechar las ventajas de la computación paralela sobre arquitecturas híbridas CPU y GPU. Fue desarrollado por las compañías Cray, CAPS, NVIDIA y el PGI, conocido como Portland Group. La motivación para crear este estándar es facilitar el proceso de desarrollo en un entorno de programación en paralelo. Este nuevo estándar es abierto y está diseñado para permitir a millones de científicos y programadores técnicos que trabajan en química, biología, física, análisis de datos, tiempo y el clima, inteligencia y muchos otros campos aprovechar estos beneficios de la computación GPGPU. (HPCwire, 2012)

La API de OpenACC utiliza un conjunto de directivas de compilador que especifica las regiones de código que se puede ejecutar en GPU, aunque cabe resaltar que OpenACC tiene como principal característica la portabilidad del código a través de los múltiples aceleradores y sistemas operativos. Estas directivas facilitan la programación de programas en aceleradoras externas mediante instrucciones de alto nivel que pueden ser añadidos al código original, por lo cual el programador se ahorra las necesidades de conocimientos de las instrucciones de control del hardware a un bajo nivel como son la inicialización de la GPU, transferencia de información entre la GPU y la CPU, gestionar la memoria de la GPU, entre otras. Por lo cual todas estas tareas quedan enmascaradas bajo las directivas las cuales tratan todos estos problemas implícitamente en tiempo de compilación (OpenACC, 2017).

OpenACC permite a los desarrolladores con palabras sencillas, conocidas como directivas, indicar al compilador las áreas que deben ser acelerados, sin realizar modificaciones y adaptaciones al código existente. El objetivo de las directivas es indicar al compilador que paralelice específicamente ciertas secciones del código. El compilador se encarga del proceso de mover los datos desde y hacia la CPU y la GPU, y de mapear las tareas de cómputo en el procesador adecuado.

Las directivas OpenACC facilitan la programación de la GPU, ya que mediante pocas líneas de código y sin preocuparse de la ubicación de los datos permite paralelizar en GPU un programa. Estas directivas se adicionan al código de una manera bastante fácil y rápida pero no permite al programador controlar directamente el hardware en que se ejecuta por lo cual el programa puede comportarse de otra forma a como se esperaba, además de no permitir directamente manejar los detalles de la comunicación entre CPU-GPU puede

provocar que una optimización del software resulte en un empeoramiento del rendimiento.

Las directivas de OpenACC se basan en el uso de instrucciones de compilador `#pragma` siguiendo la siguiente sintaxis:

```
#pragma acc directive-name [clause-list] new-line
```

Estas directivas se aplican al bloque de código inmediatamente posterior a la etiqueta `#pragma`.

Resumen de directivas de OpenACC

PGI ahora soporta las siguientes directivas OpenACC:

Parallel Construct

Define la región del programa que se debe compilar para la ejecución paralela en el dispositivo acelerador.

Kernels Construct

Define la región del programa que se debe compilar en una secuencia de kernels para su ejecución en el dispositivo acelerador.

Data Directive

Define datos, generalmente matrices, que deben asignarse en la memoria del dispositivo durante la duración de la región de datos, si los datos deben ser copiados

del host a la memoria del dispositivo al ingresar a la región y copiados desde el dispositivo a la memoria del host al salir de la región.

Enter Data and Exit Data Directives

Esta directiva define datos, generalmente matrices, que deberían asignarse en la memoria del dispositivo durante la duración del programa o hasta una directiva de datos de salida que desasigna los datos, y si los datos deben copiarse del host a la memoria del dispositivo en la directiva enter data. La directiva Exit Data define los datos, generalmente matrices, que se deben desasignar en la memoria del dispositivo, y si los datos se deben copiar desde el dispositivo a la memoria del host.

Host_Data Construct

Hace que la dirección de los datos del dispositivo esté disponible en el host.

Loop Directive

Describe que tipo de paralelismo usar para ejecutar el ciclo y declarar variables y arreglos y operaciones de reducción privados dentro del ciclo. Se aplica al ciclo que debe aparecer en la siguiente línea de la directiva.

Combined Parallel and Loop Directive

Es un atajo para especificar una Loop Directive anidada inmediatamente dentro de una Parallel Directive de acelerador. El significado es idéntico a especificar explícitamente una construcción paralela que contiene una Loop Directive.

Combined Kernels and Loop Directive

Es un atajo para especificar una Loop Directive anidada inmediatamente dentro de una Kernel Directive. El significado es idéntico a la especificación explícita de núcleos que contiene una Loop Directive.

Cache Directive

Especifica los elementos de matriz o submatrices que se deben buscar en el nivel más alto de la memoria caché para el cuerpo de un Loop. Debe aparecer en la parte superior dentro del Loop.

Declare Directive

Especifica que una matriz o matrices deben asignarse en la memoria del dispositivo durante la duración de la región de datos implícita de una función, subrutina o programa. Especifica si los valores de los datos se van a transferir del host a la memoria del dispositivo al ingresar a la región de datos implícita y del dispositivo a la memoria del host al salir de la región de datos implícita.

Crea una copia de dispositivo visible de la variable o matriz.

Update Directive

Usada durante la vigencia de los datos del acelerador para actualizar toda o una parte del arreglo de memoria del host con los valores del arreglo correspondiente en la memoria del dispositivo o para actualizar todo o parte del arreglo de memoria del dispositivo con los valores del arreglo correspondiente en la memoria del host.

Routine Directive

Se utiliza para decirle al compilador que compile un procedimiento dado para un acelerador, así como también como para el host. En un archivo o una rutina con una llamada a procedimiento, la Routine Directive le dice a la implementación los atributos del procedimiento cuando es llamada sobre el acelerador. A partir de la versión PGI 14.9 una ACC Routine Directive sin la cláusula de paralelismo se tratará como si la cláusula seq estuviera presente.

Wait Directive

Especifica la espera hasta que todas las operaciones sobre un dispositivo específico asincrónico o todas las tareas asincrónicas estén completadas.

2.2.4. OpenMP

Es una interfaz de programación de aplicaciones (API) para la programación de multiprocesos de memoria compartida en múltiples plataformas. Es un modelo de programación portable y escalable que proporciona a los programadores una interfaz simple y flexible para el desarrollo de aplicaciones paralelas para las plataformas que van desde computadoras de escritorio hasta supercomputadores. OpenMP permite paralelizar secciones determinadas de código, por eso es fácil acelerar las secciones donde se requiere más capacidad de cómputo mediante las directivas de OpenMP para que éstas sean ejecutadas por un número determinado de procesos, permitiendo obtener una ejecución más rápida.

2.2.5. Herramientas de Profiling

Para la optimización del rendimiento de un programa se hace necesario saber las partes del programa en donde se concentra la mayor parte del tiempo de ejecución, por esto se hace necesario utilizar herramientas de profiling que controlan la ejecución del programa sobre el que se quiere obtener información y toma datos de tiempos de ejecución de cada una de las funciones que forman el programa para su posterior análisis por parte del programador. La herramienta más utilizada es gprof ya que proporciona reportes con información más clara y fácil de interpretar. GProf es el profiler de GNU y permite obtener en cada ejecución de un programa para cada una de las funciones el número de llamadas que se ha realizado, el tiempo consumido total y los segundos consumidos por cada llamada. Para poder utilizar

gprof es necesario utilizar un flag de compilación `-pg` como se muestra a continuación:

```
c++ -pg -std:c++11 fd3d_diffusion_prb.cpp fd3d_diffusion.cpp -o fd3d_serial
```

Donde:

C++: es la llamada al compilador.

`-pg`: inserta código al inicio y al final de cada función con el fin de contabilizar número de llamadas, tiempos, etc.

`-o`: especifica el archivo que contendrá el producto de la compilación es decir el ejecutable.

La ejecución del ejecutable obtenido (`fd3d_serial`), almacena en un archivo `gmon.out` la información necesaria para gprof. Con este archivo se puede visualizar esa información con la llamada a gprof, mediante la utilización del siguiente comando:

```
gprof -p -b fd3d_serial
```

Donde:

`-p`: muestra perfil plano.

`-b`: elimina verbosidad.

`fd3d_serial`: es el ejecutable que se desea analizar, el cual debe haberse compilado previamente con las flags `-pg` y `-g`.

3. Discretización de ecuaciones diferenciales

Para describir un determinado fenómeno, se plantea un sistema de ecuaciones diferenciales ordinarias o parciales, las cuales en algunos casos resulta muy complicado y a veces imposible encontrar la solución analítica, por lo cual se hace necesario encontrar valores a estas funciones en un conjunto discreto de puntos, mediante la aproximación de las derivadas a fracciones de incrementos de la función y de las variables, por lo cual se pasa de una expresión diferencial a una expresión algebraica (Jöhnk, 2004).

3.1. Método de Diferencias Finitas

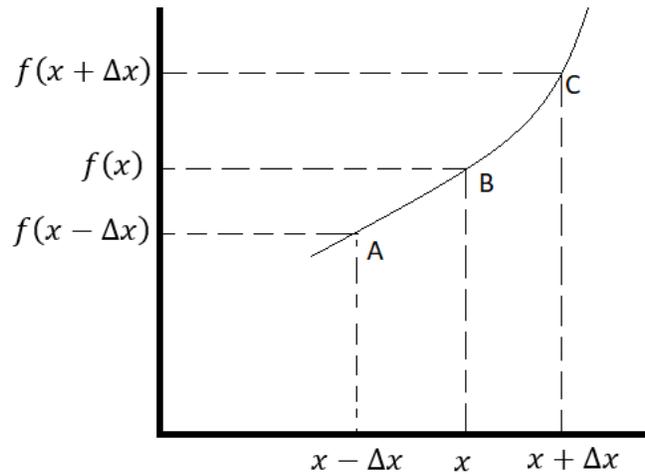
El método de diferencias finitas permite calcular el resultado aproximado de ecuaciones diferenciales en derivadas parciales definidas en espacios finitos. El método de diferencias finitas utiliza la matemática convencional y sustituye las derivadas por diferencias finitas hasta llegar a obtener un sistema de ecuaciones algebraicas para el área de estudio el cual se puede resolver por métodos directos o iterativos (Li, 2011).

Este método se basa en el desarrollo por series de Taylor Figura 3., de la siguiente forma:

$$f(x + \Delta x) = f(x) + \frac{df(x)}{dx} \frac{\Delta x}{1!} + \frac{d^2f(x)}{dx^2} \frac{(\Delta x)^2}{2!} + \dots + \frac{d^n f(x)}{dx^n} \frac{(\Delta x)^n}{n!} \quad (1)$$

$$f(x - \Delta x) = f(x) - \frac{df(x)}{dx} \frac{\Delta x}{1!} + \frac{d^2f(x)}{dx^2} \frac{(\Delta x)^2}{2!} - \dots + \frac{(-1)^n d^n f(x)}{dx^n} \frac{(\Delta x)^n}{n!} \quad (2)$$

Figura 3. Series de Taylor



Antonio Carrillo Ledesma, Introducción al Método de Diferencias Finitas y su Implementación Computacional.

Si la ecuación se resuelve para df/dx se tiene:

$$\frac{df(x)}{dx} = \frac{f(x+\Delta x) - f(x)}{\Delta x} + O(\Delta x) \quad (3)$$

Donde $O(\Delta x)$ es el término que falta de la serie.

Despreciando el término $O(\Delta x)$ de (3) se obtiene la aproximación de la derivada de f por diferencias finitas hacia adelante.

$$\frac{df(x)}{dx} \approx \frac{f(x+\Delta x) - f(x)}{\Delta x} \quad (4)$$

Resolviendo la ecuación (2) y despreciando el término $O(\Delta x)$ se obtiene la aproximación de la derivada de f por diferencias finitas hacia atrás.

$$\frac{df(x)}{dx} \approx \frac{f(x) - f(x - \Delta x)}{\Delta x} \quad (5)$$

El error que se obtiene en ambas aproximaciones es el término que se desprecia de la serie de Taylor, el cual es $O(\Delta x)$.

Para obtener la aproximación de la primera derivada por diferencias centradas a (1) le resta (2) y se obtiene la siguiente ecuación:

$$\frac{df(x)}{dx} = \frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x} + O\{(\Delta x)^2\} \dots \dots (6)$$

Donde el término $O\{(\Delta x)^2\}$ es el término que falta de la serie, despreciándolo se tiene:

$$\frac{df(x)}{dx} \approx \frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x} \dots (7)$$

El error que se tiene en esta aproximación es de $O\{(\Delta x)^2\}$.

Para obtener la aproximación a la segunda derivada se suma (1) y (2) obteniendo:

$$\frac{d^2f(x)}{dx^2} = \frac{f(x + \Delta x) - 2f(x) + f(x - \Delta x)}{(\Delta x)^2} + O\{(\Delta x)^2\} \dots \dots (8)$$

Despreciando el término $O\{(\Delta x)^2\}$ se tiene:

$$\frac{d^2 f(x)}{dx^2} \approx \frac{f(x+\Delta x) - 2f(x) + f(x-\Delta x)}{(\Delta x)^2} \quad (9)$$

3.1.1. Aproximación a las derivadas en 2D y 3D

Para obtener la aproximación a las derivadas parciales en dos variables x y y , se hace manteniendo una de las variables fija. Si se mantiene fija y se tiene:

$$\frac{df(x,y)}{dx} \approx \frac{f(x+\Delta x,y) - f(x-\Delta x,y)}{2\Delta x} \quad (10)$$

Y si se mantiene x fija se obtiene:

$$\frac{df(x,y)}{dy} \approx \frac{f(x,y+\Delta y) - f(x,y-\Delta y)}{2\Delta y} \quad (11)$$

Esto mismo se aplica para las derivadas de orden dos en dos dimensiones en donde se obtiene la aproximación a las segundas derivadas parciales en las mismas dos variables x y y . Manteniendo y fija se tiene:

$$\frac{d^2 f(x,y)}{dx^2} \approx \frac{f(x-\Delta x,y) - 2f(x,y) + f(x+\Delta x,y)}{\Delta x^2} \quad (12)$$

Y si ahora se mantiene x fija se tiene:

$$\frac{d^2 f(x,y)}{dy^2} \approx \frac{f(x,y-\Delta y) - 2f(x,y) + f(x,y+\Delta y)}{\Delta y^2} \quad (13)$$

Para el caso de tres dimensiones se obtiene la aproximación a las derivadas parciales de primer orden en tres variables x, y y z lo cual se hace manteniendo dos de las variables fijas.

Si se mantiene y y z fijas se tiene:

$$\frac{df(x,y,z)}{dx} \approx \frac{f(x+\Delta x,y,z)-f(x-\Delta x,y,z)}{2\Delta x} \quad (14)$$

Si se mantiene x y z fijas se tiene:

$$\frac{df(x,y,z)}{dy} \approx \frac{f(x,y+\Delta y,z)-f(x,y-\Delta y,z)}{2\Delta y} \quad (15)$$

Si se mantiene x y y fijas se tiene:

$$\frac{df(x,y,z)}{dz} \approx \frac{f(x,y,z+\Delta z)-f(x,y,z-\Delta z)}{2\Delta z} \quad (16)$$

En el caso de las derivadas parciales de orden dos en tres dimensiones se tiene que si se mantiene y y z fijas:

$$\frac{d^2 f(x,y,z)}{dx^2} \approx \frac{f(x-\Delta x,y,z)-2f(x,y,z)+f(x+\Delta x,y,z)}{\Delta x^2} \quad (17)$$

Si se mantiene x y z fijas:

$$\frac{d^2 f(x,y,z)}{dy^2} \approx \frac{f(x,y-\Delta y,z) - 2f(x,y,z) + f(x,y+\Delta y,z)}{\Delta y^2} \quad (18)$$

Y por último si se mantiene x, y fijas:

$$\frac{d^2 f(x,y,z)}{dz^2} \approx \frac{f(x,y,z-\Delta z) - 2f(x,y,z) + f(x,y,z+\Delta z)}{\Delta z^2} \quad (19)$$

4. Modelos de predicción de la vida útil del concreto reforzado

Las investigaciones del deterioro de las estructuras de concreto se iniciaron mediante métodos de valoración y predicción de la vida útil, se estudió su deterioro estudiando aisladamente un solo proceso de degradación. A través de estas investigaciones se fue entendiendo que el deterioro de las estructuras de concreto se debía a la combinación de efectos provenientes de varios factores y que las aproximaciones en donde solo se tenía en cuenta un factor resultaban con un grado de aproximación insuficiente (Officials, 2007). Por esto se hace complejo que exista un modelo unificado con el cual se pueda proveer el periodo de vida útil de una estructura.

Hasta hace unos años los modelos de predicción de la vida útil del concreto reforzado fueron empíricos, surgidos sobre la base de las evaluaciones técnicas que se hacen sobre un conjunto de estructuras afectadas, sin embargo, han surgido propuestas provenientes del campo científico e industrial.

4.1. Modelos empíricos

Estos modelos surgen a partir de la experiencia con el concreto reforzado en el campo académico, científico e industrial. Estos modelos son solo cuantitativos y asumen usualmente, que si un concreto se produce bajo determinado estándar de construcción las estructuras tendrán un determinado tiempo de vida útil. Muchos de los códigos de construcción especifican ciertos criterios que se debe tener en cuenta para la construcción en concreto y de ahí se presume que la vida útil de estas estructuras será la adecuada para la mayoría de los usos (Sísmica, 2012). En ambientes que no son agresivos esta presunción resulta adecuada para la mayoría de usos.

4.2. Modelos según la comparación de desempeño

Se asume que si una estructura fue fabricada con un concreto de similares características a uno de otra estructura el cual ha tenido un tiempo de vida útil determinado y se encuentra en un ambiente similar, se puede suponer que esta nueva estructura tendrá un tiempo de vida útil muy similar al de la otra estructura. Los inconvenientes que se tiene en este enfoque es que cada proyecto es diferente y particular en cuanto a su geometría, la variación de los materiales, microclimas particulares de cada ambiente, etc. También hay que tener en cuenta que las técnicas de producción, los aditivos del concreto utilizados en la industria no son los mismos que hace unas décadas (Committee 365, 2010).

4.3. Modelos estocásticos

Son conocidos como modelos probabilísticos, estos fueron propuestos al considerarse que la predicción de la vida útil de las estructuras de concreto reforzado constituye un problema al que no se puede dar un enfoque determinístico, es decir que este no puede ser calculado de manera precisa, debido a que los factores que afectan la vida útil de una estructura son numerosos, complejos y de difícil predicción (Pereira, 2003).

Estos modelos se alimentan de una amplia base de datos de resultados de ensayos obtenidos en laboratorio, usualmente ensayos de degradación acelerada en los cuales se obtiene valores de esfuerzo de falla.

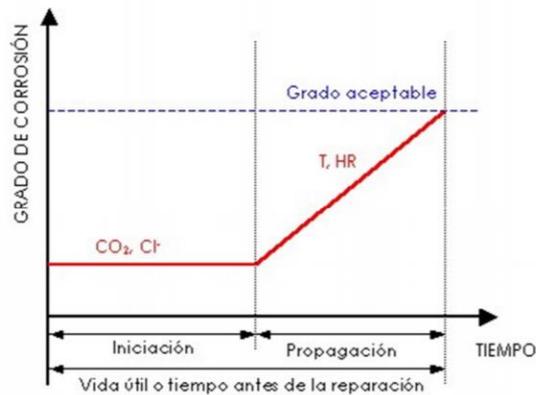
4.4. Modelos matemáticos

Las investigaciones se han concentrado en la formulación de modelos matemáticos que permiten describir el proceso de deterioro de las estructuras de concreto y llegar a predecir el periodo de vida útil de una estructura. Según la experiencia se ha demostrado que un modelo matemático no es confiable sino parte de una base experimental sólida en la obtención de los parámetros con los cuales trabaja el modelo (Khitab, 2005). Los modelos matemáticos se concentran principalmente en algunos de los factores primarios que atacan a las estructuras de concreto, para así predecir el tiempo de vida útil. Los factores primarios se refieren a aquellos que tienen una mayor incidencia en el deterioro de la estructura, es así como los más difundidos son aquellos que se basan en la corrosión del acero de refuerzo, el avance de frente de carbonatación y el ingreso de cloruros.

Modelos de corrosión del acero de refuerzo

En este modelo se vincula el concepto de vida útil al inicio y desarrollo de la corrosión en el acero de refuerzo como se muestra en la figura 4. (Tuutti, 1982)

Figura 4. Modelo de vida útil de Tutti.



K. Tuutti, Corrosion of Steel in Concrete

En esta gráfica se puede apreciar dos fases, una de iniciación, que es el tiempo en que tarda los agentes agresivos en alcanzar al acero de refuerzo a través del concreto, concentrarse y despasivar el acero de refuerzo, la segunda fase corresponde al periodo de propagación donde los iones agresivos continúan concentrando y aumentando el grado de corrosión y deterioro hasta que se alcanza un estado límite inaceptable. Este modelo no tiene una base matemática, pero es muy citado debido a su sencillez conceptual y valor didáctico y es la base de muchos modelos matemáticos.

Modelos de avance del frente de carbonatación

El frente de carbonatación es el límite entre dos concretos con una marcada diferencia entre sus valores de pH. El concreto de recubrimiento contiene este límite, siendo el concreto con contacto directo con el exterior el que posee un pH bajo ($\text{pH} < 7$), por efecto de los agentes agresivos del medio (Lizarazo Marriaga, 2010). El concreto que está en contacto directo con el acero de refuerzo mantiene un pH alto ($\text{pH} > 12$). Este límite avanza de manera progresiva hacia el acero de refuerzo hasta alcanzarlo, dando como resultado la despasivación de la capa protectora del acero e iniciando el proceso de la corrosión. Las ecuaciones en su mayoría describen el progreso de la carbonatación del concreto como función del tiempo, se basan principalmente en la primera ley de Fick, la cual se utiliza para describir cualquier proceso de difusión de la naturaleza.

$$X_c = k * \sqrt{t} \quad (20)$$

Donde X_c , es la profundidad del frente carbonatado y t el tiempo que la estructura ha estado expuesta a este agente agresivo.

Modelos de penetración de iones de cloruro

Los esfuerzos hacia la obtención de un modelo unificado que estime adecuadamente la vida útil de las estructuras de concreto han estado orientados a la modelación de los mecanismos de transporte del ión cloruro y otras especies iónicas. Durante las últimas dos décadas se han divulgado una serie de propuestas que permiten modelar el ingreso de los iones de cloruro en el concreto reforzado (Castellote, 2001). De este número de propuestas se reconocen dos tendencias, una que se basa principalmente en la Ley de Fick y los que consideran la interacción del ión cloruro con otros iones mientras se mueven por la fase acuosa del concreto, los cuales son conocidos como modelos multi-especie.

4.4.1. Modelos tradicionales

Leyes de Fick

Las leyes de Fick son la aproximación más clásica al problema de la difusión de la materia o energía en un medio en el que no existe un equilibrio químico o térmico. Aplicándolo al caso del movimiento de especies iónicas, se tiene que el transporte iónico es originado por la diferencia en el gradiente de concentración de la sustancia (Crank, 1976). El movimiento iónico se muestra en las Leyes de Fick en forma de ecuaciones diferenciales, las cuales describen este fenómeno sin dependencia y con dependencia del tiempo.

Primera ley de Fick

Cuando el fenómeno no depende del tiempo, esto es ni la concentración ni la velocidad de difusión cambia con el tiempo, se dice que el estado del fenómeno es estacionario (Alejo. , 2005). En esta condición la relación entre el flujo de iones, denotado como J_i y la concentración C_i está definida por la primera Ley de Fick:

$$J_i = -D_i \frac{\partial C_i}{\partial x} \quad (21)$$

Donde D_i , es el coeficiente de Difusión (m²/s), J_i es el flujo (mol/m²/s) y x (m) la profundidad de penetración.

Segunda Ley de Fick

En el fenómeno de la difusión para la evaluación de la variación de la concentración en el tiempo se supone que la concentración de la especie iónica decae proporcionalmente al flujo iónico de la misma especie iónica en un punto x (Alejo. , 2005). Así es que cuando las condiciones de transporte por difusión son no estacionarias, se utiliza la segunda Ley de Fick.

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial x} \left(D_{ns} \frac{\partial C}{\partial x} \right) \quad (22)$$

Según esta ley la velocidad de incremento de la concentración $\partial C/\partial t$ es proporcional a la variación del flujo de materia que existe en ese punto ($D_{ns} \partial C/\partial x$). Por lo cual el transporte de materia se produce debido al gradiente de concentraciones $\partial C/\partial x$.

4.4.2. Modelos Multi-especie

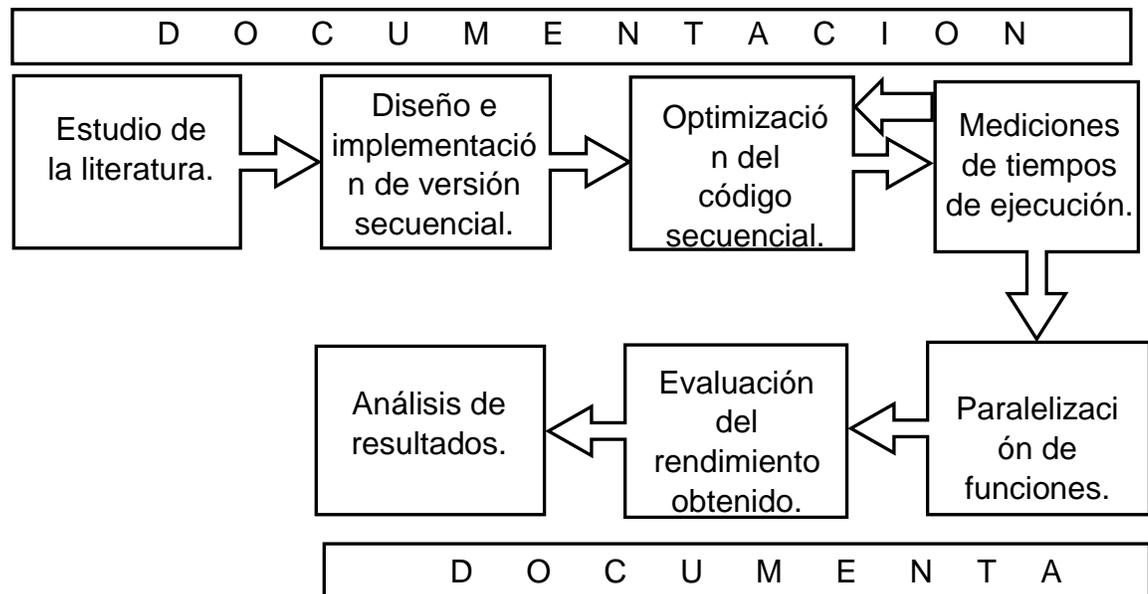
Debido a los extensos estudios realizados sobre la degradación del concreto se ha hecho evidente que los resultados obtenidos, muchas veces no reflejan la realidad, dando la impresión de que algo le hace falta a la formulación del problema, por esto diversos autores han llamado la atención de que un fenómeno tan complejo como el ingreso de cloruros en el concreto no puede ser analizado de manera exitosa con el uso de ecuaciones que simplifican el fenómeno por no considerar la interacción del cloruro con otros iones (Narsilio, 2007). Es así como las investigaciones en este

campo se concentran en encontrar resultados más satisfactorios, surgiendo las primeras aproximaciones hace un poco más de una década (Johannesson, 2003) (Truc, 2000) en las cuales se estudiaba el fenómeno de la corrosión del concreto de una manera más integral, teniendo en cuenta la interacción del ión cloruro con otros iones.

5. METODOLOGIA

En este capítulo se explica de forma resumida el proceso que se llevó a cabo para la realización del presente proyecto. Como se puede ver en la figura 5, cada una de las fases para el desarrollo del proyecto y a lo largo de estas fases la documentación, la cual se desarrolló paralelamente a las fases.

Figura 5. Fases de desarrollo



5.1. Estudio de la literatura

Para la realización del presente proyecto primero se analizó y estudió el modelo matemático de difusión de los cloruros combinado con la difusión de carbonatos y se miró otras investigaciones y desarrollos que aportaron conocimientos necesarios para la realización de este proyecto y como segundo se identificó las metodologías

y técnicas de programación sobre GPU más viables para el desarrollo de la simulación de la degradación del concreto reforzado.

5.2. Diseño e implementación de la versión secuencial

Como primer paso se diseñaron las estructuras de datos y los algoritmos de simulación de la difusión de cloruros y carbonatos, utilizando el método de diferencias finitas para la solución de ecuaciones diferenciales parciales que corresponde al modelo matemático estudiado en la fase anterior. Posteriormente se implementó la versión secuencial del algoritmo de simulación de la degradación del concreto, utilizando el lenguaje de programación C++ bajo un entorno Linux, ya que es el lenguaje de mayor uso a nivel científico y el cual permite incorporar las librerías de paralelización más utilizadas por la comunidad científica como son OpenACC, CUDA, OpenMP, MPI etc.

5.3. Optimización de la versión secuencial

El propósito de este paso es conseguir un código que sea lo más eficientemente posible, es decir, que no realice operaciones que no sean necesarias y que el tiempo de ejecución y el consumo de memoria sea lo más mínimo, identificando así la redundancia en cálculos matemáticos y las variables exclusivamente necesarias para la simulación y una vez no se necesiten sean eliminadas de la memoria, esto con el fin de que el programa no vaya a presentar caídas por falta de memoria para el almacenamiento de variables ya que el volumen de datos que maneja el algoritmo es bastante extenso.

5.4. Medición de los tiempos de ejecución

Una vez obtenida la versión secuencial optimizada, se procedió a la medición de tiempos de ejecución, para esto se utilizó la herramienta de profiling gprof, las cuales permiten obtener en detalle el consumo de tiempo de ejecución de cada una de las funciones con el fin de identificar las funciones que consumen más tiempo de ejecución y se aplicó mejoras en algunas de sus operaciones con el fin de reducir estos tiempos de ejecución.

5.5. Paralelización de funciones

Una vez identificadas y optimizadas las funciones que más tiempo de ejecución consumen, se procedió a aplicar las técnicas de paralelización sobre GPU para cada una de las funciones utilizando las librerías OpenAcc y OpenMP, logrando tiempos de ejecución menores y por ende una mayor aceleración.

5.6. Evaluación del rendimiento

Con el algoritmo de simulación paralelizado se procedió a medir los tiempos de ejecución, para esto se utilizó la herramienta de profiling PGI Profiler, que es la herramienta que proporciona las librerías de OpenACC de PGI Compiler & Tools de NVidia Corporation.

5.7. Análisis de resultados

Finalmente se ha realizado el análisis de resultados con el fin de analizar la aceleración y eficiencia obtenida al aplicar las técnicas de paralelización sobre GPU

e identificar que partes de los algoritmos tienen posibilidad de mejora para posibles trabajos futuros.

5.8. Documentación

A la par que fue desarrollada cada una de las anteriores fases de desarrollo del presente proyecto se desarrolló la fase de documentación donde se consignaron las experiencias desarrolladas, los conocimientos adquiridos y toda la información obtenida a lo largo de la realización del presente trabajo de investigación. Además del artículo presentado en el congreso internacional de Materiales CIM 2017 realizado en la ciudad de Barranquilla y el poster presentado en el Conferencia de Tecnología GPU GTC 2018, en donde se presentaron los resultados de la investigación realizada en el presente proyecto de investigación.

6. Simulación de la predicción de la vida útil del concreto

La corrosión del refuerzo debido a la penetración de iones cloruro y la carbonatación del hormigón es un problema importante que reduce la durabilidad de las estructuras de hormigón armado. Una vez que la concentración de cloruro alrededor de la superficie del refuerzo de acero excede una cierta concentración límite, o el valor de pH de la solución de poro de concreto disminuye a un valor umbral debido a la reacción de carbonatación, el refuerzo de acero se someterá al proceso de despasivación y luego a la corrosión metálica. (Bazant, 1979) (Caims, 1998) (D.W.S. Ho, 1987)

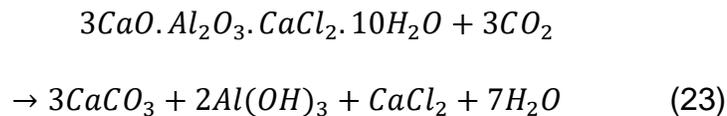
6.1. Modelo a simular

La entrada de iones cloruro a menudo es significativa en un medio atmosférico marino, donde el suministro de iones cloruro debido al rocío de la sal y la carbonatación del hormigón pueden ocurrir simultáneamente (Huang, 2013). Por ejemplo, las entradas de los túneles en un puerto están sujetas a condiciones severas de rocío de sal, y al mismo tiempo deben resistir la contaminación con dióxido de carbono con una concentración de cinco a seis veces más alta que en la mayoría de los otros entornos costeros. Dada la interacción de la carbonatación con la amenaza de durabilidad causada por la entrada de iones cloruro, ambos factores deben tenerse en cuenta en el desarrollo de una herramienta de simulación. La carbonatación influye significativamente en el transporte de iones cloruro en concreto. Numerosos experimentos se han llevado a cabo para estudiar esta influencia en el daño de la estructura (P. Chindaprasirt, 2008) - (V. Ngala, 1997), (UIS, 2009). De acuerdo con esas investigaciones experimentales, la influencia precisa de la carbonatación es tan complicada de estudiar, que es difícil notar si la carbonatación acelerará o desacelerará el daño de durabilidad del hormigón debido a los iones cloruro. La influencia de la carbonatación en el coeficiente de difusión de

iones cloruro depende de los tipos y las proporciones de mezcla de concretos considerados experimentalmente (P. Chindaprasirt, 2008), (V. Ngala, 1997), (C. Yuan, 2012).

En un trabajo experimental donde la carbonatación y el ingreso de iones de cloruro se cargaron alternativamente a la muestra (M. Lee, 2013)- (J. Backus, 2013), se determinó que la concentración de iones cloruro era máxima cerca del frente de carbonatación.

Aunque la carbonación y el ingreso de cloruro se producen simultáneamente, se debe tener en cuenta que la difusión de iones cloruro es mucho más rápida que la carbonatación. Es decir, antes de la carbonatación, el concreto por lo general contiene sal de Friedel debido al ion cloruro unido al concreto. Una vez que la sal reacciona con el dióxido de carbono durante el proceso de carbonatación, los iones cloruro se liberan a la solución de poro en el concreto como se muestra en la siguiente ecuación:



Estos iones de cloruro liberados aumentan la concentración de cloruro libre significativamente más alta que la de una simple difusión de iones de cloruro desde la superficie hacia el interior. Por lo tanto, para analizar y predecir los efectos combinados de la carbonatación y la entrada de cloruro, es esencial modelar de forma realista cómo la carbonatación interactúa con el transporte de cloro sin carbonatación.

Algunos trabajos propusieron funciones directas de influencia de la carbonatación sobre el transporte de cloro en el concreto (Yoon, 2009) (W. Puatatsananon, 2005) pero analizaron solamente la influencia de la carbonatación en el coeficiente de

difusión del cloruro. Esos modelos no pueden explicar los resultados de las pruebas alternas (M. Lee, 2013) (Yoon, 2007). Una serie de simulaciones del proceso de carbonatación se encuentran en la literatura (O. Isgor, 2004) (A. Saetta, 2004) (A. Saetta, 2005).

El proceso de carbonatación consiste en la difusión del dióxido de carbono y la reacción del dióxido de carbono con el hidróxido de calcio. En la mayoría de los trabajos, la influencia de otros productos de hidratos, como silicato cálcico hidrato, apenas se considera en la simulación del proceso de carbonatación.

Otros trabajos simplifican la entrada de iones de cloruro en sus modelos como un proceso de difusión simple. La capacidad de unión de los iones cloruro, la convección, la temperatura y la humedad, que influyen significativamente en el transporte de iones cloruro, no se consideran en la mayoría de los trabajos (B. Saassouh, 2012)- (Q. Suo, 2009). Mientras que la versión probabilística de la iniciación de la corrosión debido a la carbonatación se puede encontrar en la literatura (F. Duprat, 2014)- (T. Larrard, 2014). Sin embargo, estos estudios se basaron en la predicción empírica de la profundidad de la carbonatación para predecir el pH cerca del refuerzo de acero.

En este trabajo se utiliza un modelo integral para el efecto combinado de la carbonatación y el ingreso de cloruro, que se verificará con datos experimentales de un proyecto de Colciencias-Armada Nacional y la UIS, donde el proceso de la carbonatación hace que el pH de la solución poro concreto disminuya a un valor umbral y a su vez la concentración de cloruro alrededor de la superficie del refuerzo de acero excede una concentración umbral, por lo cual el acero de refuerzo es sometido al proceso de despasivación (Bazant, 1979) (D.W.S. Ho, 1987).

Al ingreso de oxígeno se producen reacciones electroquímicas que generan productos de corrosión en la superficie del refuerzo de acero, lo cual produce el

agrietamiento de la cubierta de concreto debido a la presión del hinchamiento de los productos de la corrosión.

Proceso de la carbonatación

La carbonatación del concreto es un proceso físico y químico complejo. En nuestro modelo, este proceso se divide en cuatro partes: (1) transporte de dióxido de carbono, (2) balance de masa de hidróxido de calcio disuelto, (3) disolución de hidróxido de calcio sólido en solución de poro de concreto y (4) reacción química de CSH con dióxido de carbono. Las ecuaciones gobernantes pueden ser dadas por:

$$\frac{\partial(\phi-\phi_{we})C_{CO_2}}{\partial t} + \nabla \cdot J_{CO_2} = -I_{ch} - I_{CSH} \quad (24)$$

$$\frac{\partial\phi_{we}C_{ch,d}}{\partial t} + \nabla \cdot J_{ch,d} = -I_{ch} + I_d \quad (25)$$

$$\frac{\partial C_{ch,s}}{\partial t} = -I_d \quad (26)$$

$$\frac{\partial C_{CSH}}{\partial t} = -r_{CSH} \quad (27)$$

donde C_{CO_2} es la concentración molar de dióxido de carbono en la fase gaseosa de los poros (mol/m^3 de aire poroso), ϕ es la porosidad actual del hormigón, ϕ_{we} es la fracción de volumen de agua en poro evaporable (m^3 de solución / m^3 de concreto), J_{CO_2} es el flujo de dióxido de carbono, I_{ch} e I_{CSH} es la tasa del consumo de dióxido de carbono debido a su reacción química con $\text{Ca}(\text{OH})_2$ y CSH, respectivamente, $C_{ch,d}$ es la concentración molar del hidróxido de calcio disuelto (mol / m^3 de solución), $J_{ch,d}$ es el flujo de iones hidróxido, I_d es la velocidad disuelta del hidróxido de calcio sólido al agua de poro, $C_{ch,s}$ es la concentración molar de hidróxido de

calcio sólido (mol / m³ de concreto), C_{CSH} es la concentración molar de CSH en concreto (mol / m³ de concreto), r_{CSH} es la velocidad de reacción de CSH con dióxido de carbono. Los lados derechos en las ecuaciones (24) - (27) están determinados por el modelo de carbonatación de Papadakis (V. Papadakis, 1991) (V. Papadakis, 1989) (V. Papadakis, 1991). La relación entre ϕ_{w_e} y h se puede estimar de acuerdo con el modelo BSB (S. Brunauer, 1969). Varios métodos de estimación para el coeficiente de difusión del dióxido de carbono $D_{CO_2}^{car}$ está disponible en la literatura (A. Saetta, 2004) (V. Papadakis, 1991) (A. Saetta, 1998).

La relación entre ϕ_{w_e} y h puede ser estimada mediante:

$$\phi_{w_e} = \frac{Ck_mV_mh}{(1-k_mh)[1+(C-1)]k_mh} \quad (28)$$

Donde C, k_m y V_m son parámetros. Para $t_e \geq 5$ días y $0.3 \leq a/c \leq 0.7$ éstos se pueden ser determinados como:

$$C = \exp\left(\frac{855}{T}\right) \quad (29)$$

$$k_m = \frac{c(1-1/n_w)^{-1}}{c-1} \quad (30)$$

$$n_w = \left(2.5 + 15/t_e\right)(0.33 + 2.2 a/c) \quad (31)$$

$$V_m = \left(0.068 - 0.22/t_e\right)(0.85 + 0.45 a/c) \quad (32)$$

Donde t_e es la edad de la hidratación en días, a/c relación agua/cemento y T es la temperatura relativa en grados Kelvin.

Hay que tener en cuenta que existe un gradiente de concentración entre a superficie expuesta y el interior del concreto, por lo cual la penetración y difusión del CO₂ a

través del concreto sigue el modelo de transporte de masa, que es la segunda ley de Fick la cual se representa por la ecuación (33):

$$\frac{\partial C_{CO_2}}{\partial t} = \frac{\partial}{\partial x} \left(D_{CO_2}^{car} \frac{\partial C_{CO_2}}{\partial x} \right) \quad (33)$$

Para el cálculo del coeficiente de difusión del dióxido de carbono se adoptó la siguiente fórmula simple en este trabajo:

$$D_{CO_2}^{car} = 1.64 * 10^{-6} \phi_{hc}^{1.8} (1 - h)^{2.2} \quad (34)$$

Donde ϕ_{hc} es la porosidad de la pasta aglutinante endurecida y h es la humedad. Tenga en cuenta que ϕ_{hc} es una función de la porosidad (V. Papadakis, 1991). Durante el proceso de carbonatación, se debe considerar la influencia del cambio de porosidad en el coeficiente de difusión del dióxido de carbono.

Algunos de los silicatos no hidratados contenidos en el concreto, como $3CaO.SiO_2$ (C_3S) y $2CaO.SiO_2$ (C_2S), pueden reaccionar con el dióxido de carbono. Sin embargo, las cantidades de C_3S y C_2S son pequeñas y difíciles de determinar con precisión. Su velocidad de reacción química con el dióxido de carbono es cercana a la velocidad de reacción de CSH. Todos son insensibles al grado de carbonatación (V. Papadakis, 1991). Por lo tanto, se tomó C_3S y C_2S en el contenido de CSH C_{CSH} para reflejar el contenido de constituyentes carbonatables sólidos totales de la pasta aglutinante endurecida en el hormigón por simplicidad.

El valor de pH del concreto puede estimarse por la concentración del hidróxido de calcio disuelto (V. Papadakis, 1991) (Neville, 1995).

$$pH = \begin{cases} 14 + \log(2 * 10^{-3} C_{ch,d}), & \text{para } C_{ch,d} \geq 1 * 10^{-3} \\ 8.3 & \end{cases} \quad (34)$$

En realidad, el límite bajo del valor de pH para un concreto completamente carbonizado también depende de la concentración de CO₂ del medio ambiente (Hans, 2005). Dado que el límite máximo de pH estaría muy por debajo del valor umbral que causa la despasivación del refuerzo, la dependencia del valor último en la ecuación (34) no se considera en este modelo.

Convencionalmente, el grado de carbonatación se estima mediante el consumo de solo el hidróxido de calcio disuelto (A. Saetta, 2004). Sin embargo, el dióxido de carbono también reacciona con constituyentes sólidos del aglutinante endurecido. Esto cambia la estructura de los poros e influye en la migración de iones. Por lo tanto, las reacciones de CSH y otros silicatos no hidratados durante el proceso de carbonatación también deben considerarse en el grado de carbonatación. Aquí, el grado de carbonatación α_c se define como:

$$\alpha_c = 1 - \frac{C_{ch,s} + \phi_{we} C_{ch,d} + 3C_{CSH}}{[C_{CaO}]_0} \quad (35)$$

Donde $[C_{CaO}]_0$ es la concentración molar inicial del óxido de calcio total en el concreto. Este valor se puede estimar aproximadamente usando las proporciones de la mezcla de concreto (D. Niu, 2007).

Cambio de estructura de poro debido a la carbonatación

Cuando ocurre la carbonatación, la estructura del poro y las propiedades del material del concreto también cambiarán (C. Yuan, 2012) (V. Papadakis, 1991) (C. Yuan, 2013) (E. Bastidas-Arteaga, 2013). Esto afectará seriamente el transporte de humedad, iones cloruro y dióxido de carbono.

Por lo tanto, es importante establecer un modelo matemático que pueda reflejar las variaciones de la estructura del poro del hormigón en diferentes grados de carbonatación. Aquí, consideramos el cambio de la porosidad y el tamaño del radio máximo de los poros capilares (T. Ishida, 2009) (K. Nakarai, 2006).

La porosidad \emptyset en diferentes grados de carbonatación puede ser dada por:

$$\emptyset = \emptyset_0 - \Delta\emptyset_c \alpha_c \quad (36)$$

donde \emptyset_0 es la porosidad del concreto no carbonatado y $\Delta\emptyset_c$ es la reducción máxima de la porosidad cuando el concreto está completamente carbonatado. $\Delta\emptyset_c$ puede ser estimado por : (V. Papadakis, 1991)

$$\Delta\emptyset_c = ([C_{CaO}]_0 - 3[C_{CSH}]_0)\Delta V_{ch} + [C_{CSH}]_0\Delta V_{CSH} \quad (37)$$

Donde $[C_{CSH}]_0$ es la concentración molar inicial del CSH en el concreto, $\Delta V_{ch} = 3.85 * 10^{-6} \text{ m}^3/\text{mol}$ y $\Delta V_{CSH} = 15.39 * 10^{-6} \text{ m}^3/\text{mol}$ son los cambios de volumen molar del hidróxido de calcio y CSH después de reaccionar con el dióxido de carbono respectivamente. $[C_{CSH}]_0$ puede ser obtenido mediante:

$$[C_{CSH}]_0 = \frac{P_{CSH}[C_{CaO}]_0}{3} \quad (38)$$

Donde P_{CSH} es la proporción molar del óxido de calcio en forma de CSH y puede ser estimado aproximadamente mediante una fórmula empírica obtenida de la literatura (F. Duprat, 2014).

La porosidad inicial \emptyset_0 del concreto puede ser estimada mediante la siguiente ecuación:

$$\emptyset_0 = \frac{a/c * \rho_c / \rho_a}{a/c * \rho_c / \rho_a + a/g / c * \rho_c / \rho_{ag} + 1} \quad (39)$$

Donde:

$\rho_{ag}, \rho_c, \rho_a$: Es la densidad de los agregados, el cemento y el agua respectivamente.

a/c : es la relación agua-cemento.

ag/c : es la relación agregados-cemento.

Debido a que la porosidad va disminuyendo por la formación de carbonato de calcio, se emplea la siguiente ecuación:

$$\phi(t) = \phi_0 * e^{-\beta * \eta_{min} * t} \quad (40)$$

Donde:

$\eta_{min} = 1.78 * 10^{-5} \text{ mol/cm}^3 \text{ día}$ representa el término de la cinética de la carbonatación.

$$\beta = 4.19 \text{ cm}^3/\text{mol}$$

El radio máximo de los poros capilares refleja la conectividad de la estructura de los poros que influye en la movilidad de los iones cloruro en el concreto (T. Ishida, 2009) (K. Nakarai, 2006). Se puede calcular en función de la distribución del tamaño de poro que se puede obtener mediante la porosimetría de intrusión de mercurio. El cambio del tamaño máximo del radio debido a la carbonatación es significativo, especialmente para los hormigones de cenizas volantes (C. Yuan, 2013). Algunos de los estudios previos sobre la influencia de la carbonatación en la distribución del tamaño de poro se resumen en la literatura (A. Morandea, 2014). Aquí, para simplificar, suponemos que el tamaño del radio máximo es linealmente proporcional al grado de carbonatación α_c . Entonces, el tamaño del radio pico r_p en diferentes grados de carbonatación puede ser dado por:

$$r_p = 5 * 10^{-8} r_{p,ref} [(\bar{r}_{p,c} - \bar{r}_{p,0})\alpha_c + \bar{r}_{p,0}] \quad (41)$$

donde $r_{p,ref} = 1\text{m}$ es un tamaño de referencia y $\bar{r}_{p,0}$ y $\bar{r}_{p,c}$ son parámetros adimensionales del radio máximo antes y después de la carbonatación.

$\bar{r}_{p,0}$ y $\bar{r}_{p,c}$ dependen de la relación agua-cemento, es decir:

$$\bar{r}_{p,0} = f_h(a/c) \quad (42)$$

$$\bar{r}_{p,c} = f_h(a/c)f_c(a/c) \quad (43)$$

Donde $f_h(a/c)$ es la influencia de la relación agua-cemento sobre la estimación del radio máximo del concreto no carbonatado y $f_c(a/c)$ son las funciones de influencia de la relación agua-cemento en el cambio del radio máximo debido a la carbonatación. Estas dos funciones determinadas por datos experimentales (V. Papadakis, 1991) (C. Yuan, 2013) (A. Morandea, 2014)- (M. Thiery, 2011) son las siguientes:

$$f_h(a/c) = -4.66(a/c)^2 + 8.72(a/c) - 1.78 \quad \text{para } 0.3 \leq (a/c) \leq 0.55 \quad (44)$$

$$f_c(a/c) = 10.59(a/c)^2 - 11.36(a/c) + 4.29 \quad \text{para } 0.3 \leq (a/c) \leq 0.55 \quad (45)$$

Transporte de los iones de cloruro con carbonatación

Se asume que la ecuación de transporte de los iones de cloruro después de la carbonatación todavía cumple con la segunda ley de difusión de Fick. La cantidad total de cloruro en una unidad de volumen de concreto consiste en el cloruro libre presente en la solución de poro y el cloruro enlazado de la sal de Friedel.

$$C_{tc} = \phi_{w_e} C_{fc} + C_{bc} \quad (46)$$

Donde C_{tc} es el contenido de cloruro total en una unidad de volumen de concreto (mol/m³ de concreto) y C_{fc} es el contenido de iones de cloruro libres (mol/m³ de solución poro). C_{bc} es el contenido de cloruro unido (mol/m³ de concreto). Una parte del cloruro unido puede participar en la reacción química mostrada en la ecuación (3) y libera iones de cloruro libres.

La cantidad de cloruro unido depende de la concentración del cloruro libre en la solución de poro, así como del grado de carbonatación porque consideramos la carbonatación. Por lo tanto, la variación instantánea del cloruro total puede ser expresado como:

$$\frac{\partial C_{tc}}{\partial t} = \frac{\partial \phi_{w_e} C_{fc}}{\partial t} + \frac{\partial C_{bc}}{\partial C_{fc}} \frac{\partial C_{fc}}{\partial t} + \frac{\partial C_{bc}}{\partial \alpha_c} \frac{\partial \alpha_c}{\partial t} \quad (47)$$

Donde $\frac{\partial C_{bc}}{\partial C_{fc}}$ es un isoterma entre el cloruro unido y el cloruro libre. $\frac{\partial C_{bc}}{\partial \alpha_c}$ también se puede describir con un isoterma en el cual la carbonatación se debe considerar. La ecuación que considera el término convectivo viene dada por:

$$\frac{\partial C_{tc}}{\partial t} = \nabla D_{fc}^{car} \nabla \phi_{w_e} C_{fc} + \nabla D_h^{car} \phi_{w_e} C_{fc} \nabla h \quad (48)$$

Donde D_{fc}^{car} y D_h^{car} son los coeficientes de difusión de los iones de cloruro libres y humedad dependiendo del grado de carbonatación.

Igualando las ecuaciones (47) y (48) tenemos:

$$\frac{\partial \phi_{w_e} C_{fc}}{\partial t} + \frac{\partial C_{bc}}{\partial C_{fc}} \frac{\partial C_{fc}}{\partial t} + \frac{\partial C_{bc}}{\partial \alpha_c} \frac{\partial \alpha_c}{\partial t} = \nabla D_{fc}^{car} \nabla \phi_{w_e} C_{fc} + \nabla D_h^{car} \phi_{w_e} C_{fc} \nabla h \quad (49)$$

Esta ecuación se puede simplificar en:

$$\frac{\partial \phi_{w_e} C_{fc}}{\partial t} + \nabla J_{fc} = I_{rc} \quad (50)$$

Donde J_{fc} es el flujo de iones de cloruro y I_{rc} es el término fuente que refleja los iones de cloruro libres liberados de la sal de Friedel debido a la carbonatación. Estos pueden ser obtenidos mediante:

$$J_{fc} = -\frac{1}{1 + \frac{1}{\phi_{w_e}} \frac{\partial C_{bc}}{\partial C_{fc}}} (D_{fc}^{car} \nabla \phi_{w_e} C_{fc} + D_h^{car} \phi_{w_e} C_{fc} \nabla h) \quad (51)$$

$$I_{rc} = -\frac{1}{1 + \frac{1}{\phi_{w_e}} \frac{\partial C_{bc}}{\partial C_{fc}}} \frac{\partial C_{bc}}{\partial \alpha_c} \frac{\partial \alpha_c}{\partial t} \quad (52)$$

La capacidad de unión del concreto reforzado se ve influenciada por la carbonatación, las investigaciones reportan que la capacidad de unión residual se reduce a menos del 20% de la capacidad después de la carbonatación, por lo cual el contenido de cloruro ligado es descrito mediante el modelo de isoterma de Langmuir, de esta forma se tiene:

$$C_{bc} = \frac{\alpha_L (1 - d\alpha_c) \phi_{w_e} C_{fc}}{1 + \beta_L M(Cl^-) \phi_{w_e} (C_{fc}/b)} \quad (53)$$

La pendiente del isoterma de unión $\frac{\partial C_{bc}}{\partial C_{fc}}$ que refleja la capacidad de unión residual del concreto reforzado en diferentes grados de carbonatación está dado por:

$$\frac{\partial C_{bc}}{\partial C_{fc}} = \frac{\alpha_L (1 - d\alpha_c) \phi_{w_e} C_{fc}}{[1 + \beta_L M(Cl^-) \phi_{w_e} (C_{fc}/b)]^2} \quad (54)$$

El término fuente I_{rc} de la ecuación 52 puede ser calculado mediante:

$$I_{rc} = -\frac{d\alpha_L \phi_{w_e} C_{fc} (I_{ch} + I_{CSH})}{\left(1 + \frac{1}{\phi_{w_e}} \frac{\partial C_{bc}}{\partial C_{fc}}\right) [1 + \beta_L M(Cl^-) \phi_{w_e} (C_{fc}/b)] [C_{ca0}]_0} \quad (55)$$

Donde $M(Cl^-)$ es la masa molar de los cloruros, b es el contenido de aglutinante por unidad de volumen de concreto, α_L y β_L son parámetros determinados por el tipo de concreto, d es el factor de reducción de la capacidad de unión del ión cloruro debido a la carbonatación, para este modelo se utilizó $d=0.88$, $\alpha_L= 11.8$ y $\beta_L = 4.0$. I_{ch} es la tasa de consumo del dióxido de carbono debido a la reacción de $Ca(OH)_2$ disuelto y I_{CSH} es la tasa de consumo de dióxido de carbono debido a la reacción con CSH.

El cambio de la estructura del poro debido a la carbonatación se considera en los coeficientes de difusión de los iones cloruro y la humedad D_{fc}^{car} y D_h^{car} como se muestra en las siguiente ecuaciones:

$$D_{fc}^{car} = D_{fc} f_p(\Omega, \delta) \quad (56)$$

$$D_h^{car} = D_h f_p(\Omega, \delta) \quad (57)$$

Donde D_{fc} y D_h son los coeficientes iniciales del cloruro y la humedad antes de la carbonatación.

El coeficiente de difusión del cloruro en el concreto no carbonatado D_{fc} puede ser obtenido mediante:

$$D_{fc} = D_{fc,ref} h_1(h) h_2(T) h_3(t_e) \quad (58)$$

Donde $D_{fc,ref}$ es el coeficiente de difusión del cloruro de referencia (m^2/s) y $h_1(h)$, $h_2(T)$ y $h_3(t_e)$ son las funciones de influencia de humedad relativa del poro, la temperatura y la edad, respetivamente. Estas son obtenidas como:

$$h_1(h) = \left[1 + \frac{(1-h)^4}{(1-h_c)^4} \right] \quad (59)$$

$$h_2(T) = \exp\left(\frac{U_c}{RT_{c,ref}} - \frac{U_c}{RT}\right) \quad (60)$$

$$h_3(t_e) = \left(\frac{t_{ref}}{t_e}\right)^m \quad (61)$$

Donde U_c es la energía de activación del proceso de difusión de cloruros, este valor es de 41.8, 44.6 y 32 kJ/mol para las relaciones a/c de 0.4, 0.5 y 0.6 respectivamente (B. Martin-Perez, 2001), R es la constante de gases 8.314×10^{-3} kJ/K mol, $T_{c,ref}$ y t_{ref} son la temperatura (K) y el tiempo (s) de referencia en los que se ha evaluado la difusividad del cloruro $D_{fc,ref}$, T es la temperatura actual en el concreto (K), m es el factor de reducción de la edad (M.D.A. Thomas, 1999), donde se emplea:

$$m = 0.2 \quad (62)$$

h es la humedad en el poro del concreto, h_c es el nivel de humedad para este modelo se toma 0.75.

El valor de $D_{fc,ref}$ depende de la relación a/c y se puede obtener con:

$$D_{fc,ref} = \frac{10^{[1.776+1.364(a/c)]}}{3.1536 \cdot 10^{13}} \quad (63)$$

De igual forma el coeficiente de difusión de la humedad es obtenido por:

$$D_h = D_{h,ref} f_1(h) f_2(T) f_3(t_e) \quad (64)$$

Donde $D_{h,ref}$ es el coeficiente de difusión de la humedad de referencia (m^2/s) y $f_1(h)$, $f_2(T)$ y $f_3(t_e)$ son las funciones de influencia de humedad relativa del poro, la temperatura y la edad, respetivamente. Estas son obtenidas como:

$$f_1(h) = \alpha_0 + \frac{1 - \alpha_0}{1 + \left[\frac{1-h}{1-h_c}\right]^n} \quad (65)$$

$$f_2(T) = \exp\left(\frac{U_m}{RT_{h,ref}} - \frac{U_m}{RT}\right) \quad (66)$$

$$f_3(t_e) = 0.3 + \sqrt{\frac{13}{t_e}} \quad (67)$$

Donde α_0 es un valor entre 0.025 y 0.1, para este modelo se tomó $\alpha_0=0.05$, $h_c = 0.75$ es la humedad relativa, el valor del exponente n está entre 6 y 16 caracterizando la extensión de la caída en D_h , $U_m \approx 20.3 \text{ kJ/mol}$ es la energía de activación por hidratación, R es la constante de gases $8.314 \times 10^{-3} \text{ kJ/K mol}$ y $T_{h,ref} = 296K$ es la temperatura de referencia.

$f_p(\Omega, \delta)$ es la influencia del cambio de la estructura del poro sobre los coeficientes de difusión. Esta influencia puede ser caracterizada por la tortuosidad Ω y la constrictividad δ de la estructura del poro.

$$f_p(\Omega, \delta) = \frac{\delta}{\delta_0} \frac{\Omega_0}{\Omega} \quad (68)$$

Donde δ_0 y Ω_0 son la inicial constrictividad y la inicial tortuosidad del concreto respectivamente.

La tortuosidad es una medida de que tan tortuoso son los canales de poros y es obtenida mediante una función de la porosidad. La constrictividad mide que tan estrechos son los canales de poros para el transporte de los iones y se define utilizando el tamaño máximo de radio r_p . La tortuosidad y la constrictividad pueden obtenerse a partir de las siguientes ecuaciones: (T. Ishida, 2009) (K. Nakarai, 2006)

$$\Omega = -b_1 \tanh[b_2(\phi - b_3)] + b_4 \quad (69)$$

$$\delta = c_1 \tanh[c_2(\log(\bar{r}_p) + c_3)] + c_4 \quad (70)$$

Donde b_1 a c_4 son parámetros determinados y $\bar{r}_p = r_p/r_{p,ref}$ es un adimensional tamaño del radio máximo. De acuerdo a la literatura (T. Ishida, 2009) (C. Yuan, 2012) (C. Yuan, 2013) (K. Nakarai, 2006) se toma:

$$b_1 = 1.5, b_2 = 8.0, b_3 = 0.25, b_4 = 2.5,$$

$$c_1 = 0.395, c_2 = 4.0, c_3 = 5.95, c_4 = 0.405$$

Transferencia de calor y humedad

Debido a que la temperatura y la humedad tienen un papel importante en el análisis de la difusión de los iones de cloruro, el transporte del calor y la humedad deben ser consideradas en forma de ecuaciones de conservación.

$$\rho C_q \frac{\partial T}{\partial t} = \lambda \nabla^2 T \quad (71)$$

$$\frac{\partial h}{\partial t} = \nabla [D_h^{car} \nabla h] \quad (72)$$

Donde T es la temperatura (K), ρ es la densidad de masa del concreto (kg/m^3), C_q es el calor específico (W/K Kg), λ es la conductividad térmica, h es la humedad y D_h^{car} es el coeficiente de difusión de la humedad en el concreto carbonatado.

6.2. Algoritmo de Simulación

En este capítulo se describe el diseño e implementación del algoritmo de simulación primero en su versión secuencial y luego en su versión paralela aplicando las técnicas de cómputo paralelo con el fin de hacerlo eficiente, es decir, permita dar resultados en el menor tiempo posible y sea de gran utilidad a las empresas constructoras ya que podrán tomar decisiones oportunas en cuanto a tiempos de mantenimiento, reparación o cambios de las estructuras que están cercanas a presentar fallas y daños irreparables.

6.2.1. Diseño del algoritmo de Simulación

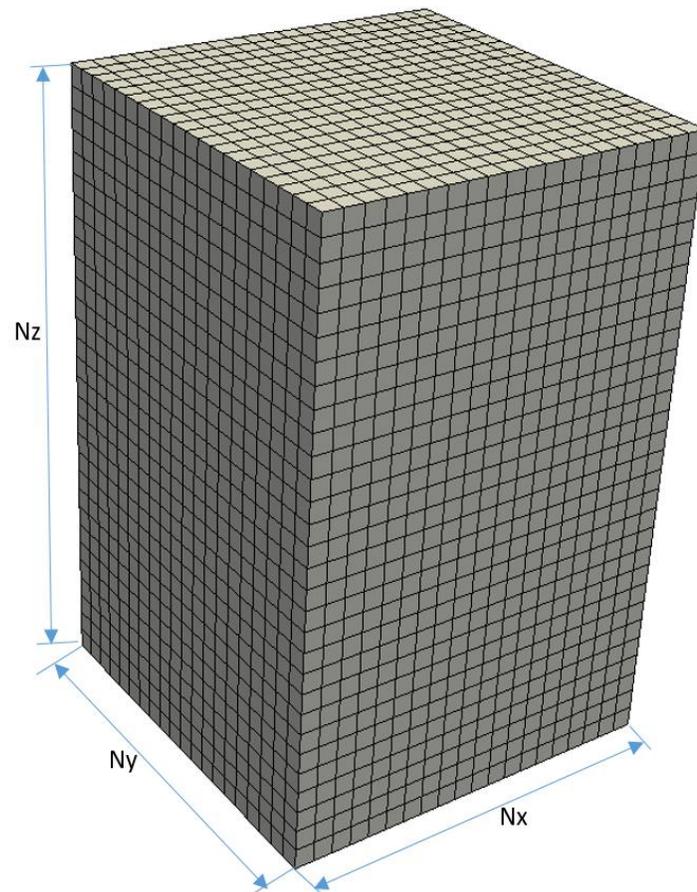
Diseño de las estructuras de datos

Para la simulación del fenómeno de la corrosión del concreto reforzado, cada una de las variables que intervienen en este proceso se representan mediante un cubo de datos, por lo cual el bloque de concreto se divide para cada una de sus coordenadas en un determinado número de segmentos, por lo cual se obtiene el cubo de datos con dimensiones $N_x \times N_y \times N_z$ de datos double como se muestra en la figura 6. Para facilitar el manejo de este tipo de variables en c++ se maneja como un puntero de datos double con $N_x \times N_y \times N_z$ posiciones. Las variables que se necesitan almacenar en memoria utilizando un puntero de datos double para utilizarlas en la simulación son las siguientes:

- Concentración de carbonatos [CCarbonates]
- Concentración de cloruros [CChlorides]
- Porosidad inicial [CPorosityIni]
- Porosidad actual [CPorosity]
- Temperatura [CTemperature]
- Humedad [CHumidity]
- Relación Tortuosidad-Constrictividad Inicial [CRel_Tort_Const_Ini]

- Relación Tortuosidad-Constrictividad [CRel_Tort_Const]
- Volumen de agua evaporable en poro [CPhi_we]
- Coeficiente de Difusión de humedad [CDh_car]
- Coeficiente de Difusión de cloruros [CDfc_car]
- Coeficiente de Difusión de carbonatos [CDco2_car]

Figura 6. Cubo de datos de variables.

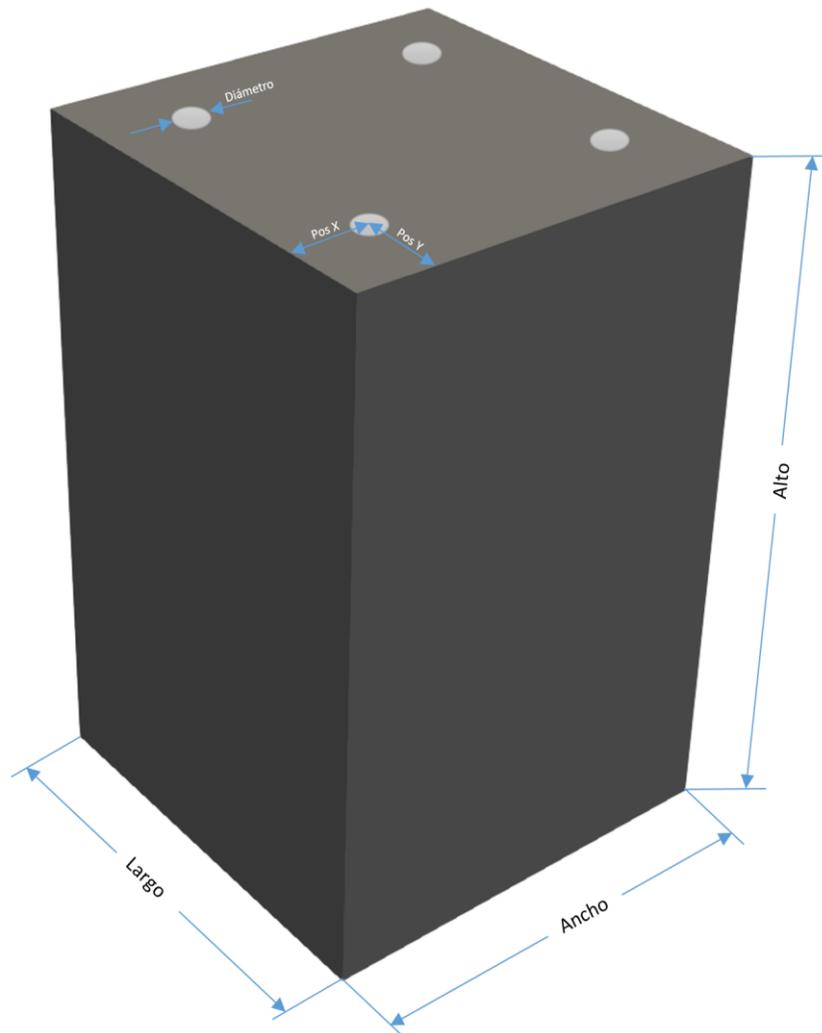


Variables de entrada

Entre las variables de entrada se tiene:

- Dimensiones del bloque de concreto a simular, largo, ancho y alto. [CLarge, CWidth, CHight]
- Número de vigas de la estructura de fortalecimiento del concreto. [NBeam]
- Coordenadas x, y de cada una de las varillas (ver figura 7). [PosBeam]

Figura 7. Variables de entrada del concreto.



La relación agua-cemento del concreto. [a_c]

La relación agregados-cemento del concreto. [ag_c]

Las densidades del cemento, agregados y agua respectivamente. [dC], [dAg], [dA]

El contenido umbral de cloruro en el concreto no carbonatado. [Cfc_th]

La concentración de cloruro en la superficie del concreto. [Cfc_b]

La temperatura de la superficie del hormigón. [Tb]

El factor de reducción de la capacidad de unión de los iones cloruros debido a la carbonatación. [d]

El factor de cambio de la porosidad del concreto debido a la carbonatación. [k_phi]

El coeficiente de difusión del cloruro de referencia del concreto no carbonatado. [Dfc_ref]

El coeficiente de difusión de la humedad de referencia del concreto no carbonatado. [Dh]

La concentración molar inicial del óxido de calcio total en el concreto. [Ccao_0]

La concentración molar de dióxido de carbono en la superficie del concreto. [Cco2_b]

La humedad relativa del aire en la superficie del concreto. [hc]

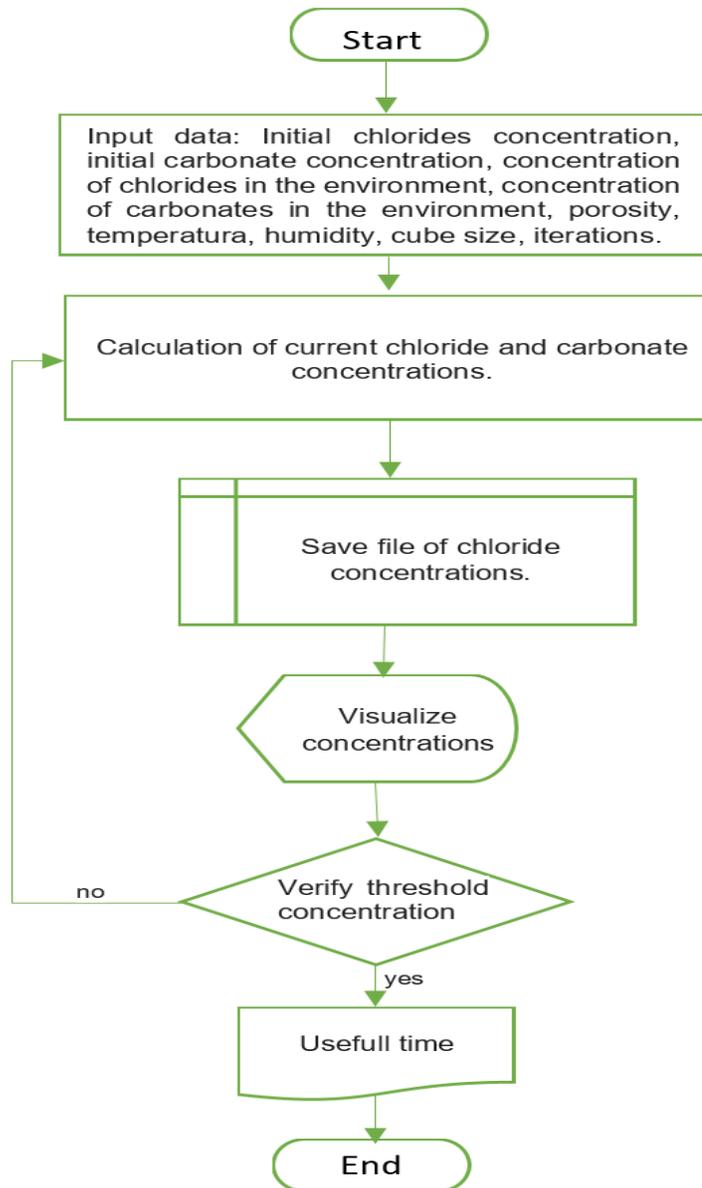
La proporción molar inicial del óxido de calcio en CSH de concreto. [Pcsh]

Diseño de los algoritmos

La primera parte del algoritmo se dedica exclusivamente a la carga de variables de entrada y a la inicialización de las estructuras de datos con los valores iniciales. Posteriormente se inicia las iteraciones donde varía la variable tiempo y se calculan las concentraciones en carbonatos y cloruros actuales, simultáneamente con la transferencia de temperaturas y humedad en el concreto. Una vez calculadas las concentraciones se almacena esta variable en un archivo en formato VTK para su

posterior visualización en la herramienta de visualización ParaView. El diagrama de flujo del algoritmo se puede visualizar en la figura 8.

Figura 8. Diagrama de flujo del algoritmo.



Generación de archivos VTK.

Las librerías de VTK son un kit de herramientas de visualización científica desarrolladas bajo el criterio de software libre basadas en la programación orientada a objetos, diseñadas para permitir la visualización y el procesamiento de imágenes, así como la creación de objetos en 2D y 3D mediante un computador. VTK consiste en una biblioteca de clases de C++ y varias capas de interfaz interpretadas como Tcl/Tk, Java y Python.

VTK admite una amplia variedad de algoritmos de visualización que incluyen escalares, vectores, tensores, texturas y métodos volumétricos, así como técnicas de modelado avanzadas como modelado implícito, reducción de polígono, suavizado de malla, corte, contorneado y triangulación de Delaunay. VTK tiene un amplio marco de visualización de la información y un conjunto de widgets de interacción 3D. El kit de herramientas admite el procesamiento en paralelo y se integra con varias bases de datos en kits de herramientas GUI como Qt y Tk. VTK es multiplataforma y se ejecuta en plataformas Linux, Windows, Mac y Unix, es además parte de la colección de plataformas de código abierto compatibles con el desarrollo de software de Kitware. (Kitware, 2017)

Formato de los archivos VTK.

Los formatos de archivos VTK consta de cinco partes básicas:

- La primera parte consta de la versión del archivo y el identificador. Esta parte contiene la siguiente línea única:

```
#vtk DataFile Version x.x
```

Esta línea debe ser exactamente como se muestra, con la excepción del número de la versión x.x que varía con las diferentes versiones de VTK. (El

número de la versión actual es la 4.2, aunque los archivos de las anteriores versiones son compatibles con la versión actual.

- La segunda parte es el encabezado, que consiste en una cadena de caracteres terminada por el carácter de fin de línea \n, el cual puede tener un máximo de 256 caracteres, se usa usualmente para describir los datos e incluir cualquier otra información pertinente.
- La siguiente parte es el formato del archivo donde se describe el tipo de archivo, ya sea ASCII o binario.
- La cuarta parte es la estructura del conjunto de datos. La parte de geometría describe la geometría y la topología del conjunto de datos. Esta parte comienza con una línea que contiene la palabra clave DATASET seguida de una palabra clave que describe el tipo de conjunto de datos. Luego dependiendo del tipo de conjunto de datos, otras combinaciones de palabras clave/datos definen los datos reales.
- La parte final describe los atributos del conjunto de datos. Esta parte comienza con la palabra clave POINT_DATA o CELL_DATA, seguida de un número entero que especifica el número de puntos o celdas respectivamente. Otras combinaciones de palabras clave/datos definen los valores del actual conjunto de datos (es decir escalares, vectores, tensores, normales, coordenadas de texturas o datos de campo).

En la figura 9 se muestra el formato general de un archivo VTK. Las tres primeras partes son obligatorias, y las otras dos son opcionales. Por lo tanto, tiene la flexibilidad de mezclar y combinar los atributos y la geometría del conjunto de datos, ya sea mediante la manipulación de archivos del sistema operativo o mediante el uso de filtros de VTK para fusionar datos. Las palabras claves son insensibles a las mayúsculas y minúsculas y pueden estar separadas por espacios en blanco.

Figura 9. Formato del archivo de datos de VTK.

```
# vtk DataFile Version 3.0 ](1)
Valores de cloruros ](2)
ASCII | BINARY ](3)
DATASET type ](4)
...
POINT_DATA n ]
...
CELL_DATA n ](5)
...
```

6.2.2. Implementación del algoritmo de simulación

Se implementó inicialmente el algoritmo de simulación en versión secuencial, la cual incluye toda la funcionalidad necesaria para la ejecución de la simulación. Esta versión consta de dos archivos los cuales contienen las funciones y procedimientos de la simulación, estos fueron implementados utilizando el estándar ISO/IEC 14882:2011, también conocido como C++11, este estándar es utilizado debido a la cantidad de optimizaciones que ya trae implementados.

El primer archivo `fd3d_diffusion.cpp` y su respectivo archivo de encabezados `fd3d_diffusion.hpp` contiene las siguientes funciones principales:

float get_fd3d_diffusion_cfl: calcula la condición de estabilidad para el método de diferencias finitas.

float get_dtdx2: calcula el valor de dt/dx^2

float get_dt2dx: calcula el valor de $dt/2dx$

*float *get_vec_new*: calcula el vector de posiciones para cada una de las coordenadas x,y y z.

*float *get_IC_3D*: crea un puntero con Nx*Ny*Nz posiciones y le establece un valor inicial a cada uno de los valores.

void write_vtk3D: esta función tiene como salida un archivo VTK en formato ASCII de tipo UnstructuredGrid, que contiene las celdas que forman el modelo del concreto y los valores de las concentraciones, esto con el fin de generar la animación de difusión del cloruro sobre el concreto. Mediante el software Paraview de distribución gratuita se reforzado.

*float *get_fd3d_diffusion_Temp_3D*: Calcula la difusión de temperatura para un tiempo t+1.

*float *get_fd3d_diffusion_2Fick_3D*: Calcula la difusión de la variable enviada como parámetro para un tiempo t+1.

*float *get_Porosity_Time*: calcula la porosidad actual.

*float *get_dCbc_dCfc*: calcula la relación entre el diferencial parcial $\frac{\delta C_{bc}}{\delta C_{fc}}$

*float *get_fd3d_gradiente*: calcula el gradiente de la variable enviada como parámetro.

*float *get_Irc*: calcula el valor Irc del modelo matemático.

El archivo fd3d_diffusion_prb contiene las siguientes funciones:

void set_bc_3D: establece los valores de frontera.

*float *set_value_3D*: esta función crear un puntero en forma de vector que contiene el cubo de datos y asigna un valor inicial.

*float *get_rp*: Calcula el valor Rp.

*float *set_porosity_ini*: calcula la porosidad inicial de acuerdo a los parámetros iniciales de entrada al algoritmo.

*float *get_Tortuosity*: calcula la tortuosidad.

*float *get_Constrictivity*: calcula la constrictividad.

float get_Dfc_reference: calcula el coeficiente de difusión de referencia de los cloruros.

*float *get_Dh_3D*: Calcula el coeficiente de difusión de la humedad.

*float *get_Dfc_3D*: Calcula el coeficiente de difusión de los cloruros.

*float *get_Dco2ca_3D*: calcula el coeficiente de difusión de los carbonatos.

*float *get_Phi_we*: calcula la porción de agua evaporable en poro.

*float *get_multiply*: multiplica los valores de dos vectores de datos.

*float *get_divide*: divide los valores de dos vectores de datos.

void Add: suma los valores de dos vectores de datos y almacena en el primer vector enviado como parámetro.

void Subtract: resta los valores de dos vectores de datos y almacena los resultados en el primer vector enviado como parámetro.

Optimización del algoritmo secuencial:

Una de las etapas en el desarrollo de un programa consiste en la optimización de su rendimiento. Se verificó cada una de las funciones que realizan cálculos sobre el cubo de datos y se evidenció que existen operaciones que se realizan repetidamente con los mismos datos dentro de los ciclos, las cuales fueron sacadas fuera de estos con el fin de que no se realicen operaciones iguales varias veces ya que se estaría perdiendo tiempo de máquina calculando los mismos resultados.

Aprovechando las novedades del estándar C++11 como es la semántica de movimiento, se eliminó las copias profundas de valores temporales mediante la

operación simple de mover el puntero al nuevo bloque de datos, mediante esto se ahorra una gran cantidad de tiempo especialmente cuando se asigna los valores de retorno de las funciones que realizan operaciones sobre los cubos de datos ya que no necesita tener al mismo tiempo dos variables del mismo tamaño en la memoria.

Se identificó las variables que son necesarias mantener constantemente en memoria las cuales son:

- ✓ Vector de posiciones X, Y y Z. [x_num], [y_num] y [z_num]
- ✓ Vector tiempo Vt.
- ✓ Concentración de carbonatos [CCarbonates]
- ✓ Concentración de cloruros [CChlorides]
- ✓ Porosidad inicial [CPorosityIni]
- ✓ Porosidad actual [CPorosity]
- ✓ Temperatura [CTemperature]
- ✓ Humedad [CHumidity]
- ✓ Relación Tortuosidad-Constrictividad Inicial [CRel_Tort_Const_Ini]
- ✓ Relación Tortuosidad-Constrictividad [CRel_Tort_Const]
- ✓ Volumen de agua evaporable en poro [CPhi_we]
- ✓ Coeficiente de Difusión de humedad [CDh_car]
- ✓ Coeficiente de Difusión de cloruros [CDfc_car]
- ✓ Coeficiente de Difusión de carbonatos [CDco2_car]

Las variables que se usan temporalmente para cálculos matemáticos, cuando ya no es necesario tenerlas almacenadas en memoria se procede a eliminarlas y así liberar esa memoria, esto con el fin de que el programa no vaya a presentar falla por no tener espacio para asignar memoria a las nuevas variables que se necesiten o el programa vaya incrementando ilimitadamente el uso de memoria hasta su caída.

Para determinar que partes del algoritmo de simulación consumen el mayor tiempo de máquina se utilizó la herramienta de profiling gprof, la cual controla la ejecución del programa sobre el cual se requiere obtener información y así permitir analizar y determinar las funciones que requieren más tiempo de máquina y por ende aplicar una mayor optimización. Para que el programa genere el archivo de salida gmon.out que es el archivo que permite generar el archivo de profiling del programa se necesita compilar el programa con la opción `-pg` como se muestra a continuación el comando de compilación debe ser como se muestra a continuación:

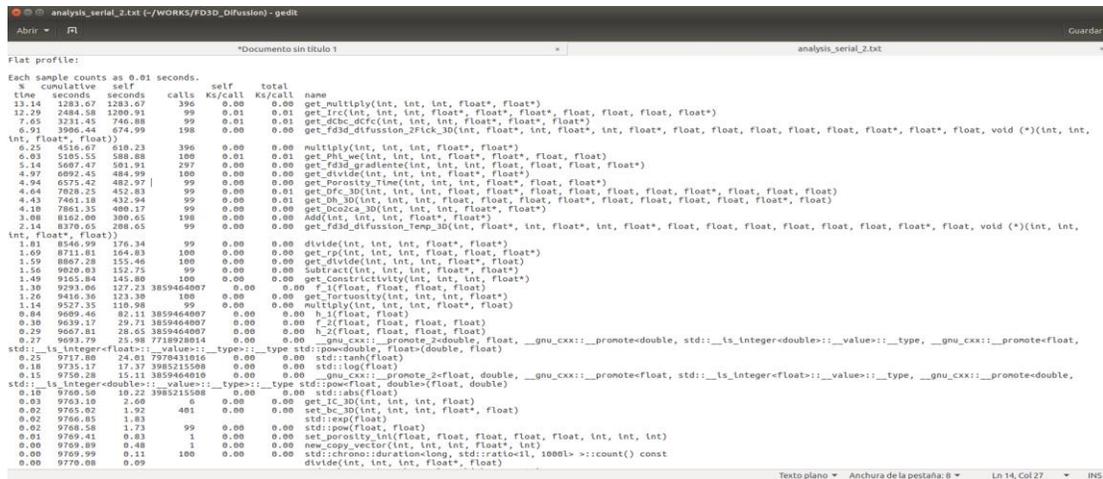
```
c++ -Wall -no-pie -std=c++11 -pg fd3d_diffusion_prb.cpp fd3d_diffusion.cpp -o fd3d_serial
```

Para la generación del archivo `analysis_serial.txt` el cual contiene los tiempos de ejecución y el uso de procesador de cada una de las funciones se necesita ejecutar el comando:

```
gprof fd3d_serial gmon.out > analysis_serial.txt
```

El archivo generado `analysis_serial.txt` el cual es el archivo de profiling es mostrado en la figura 10, donde se puede visualizar los resultados obtenidos del análisis de la versión C++ secuencial.

Figura 10. Resultados de profiling versión secuencial.



Como se puede identificar más del 70% del tiempo de ejecución lo consumen las primeras 10 funciones, entre las que se encuentran principalmente las funciones multiply, get_Irc, fd3d_diffusion_2Fick_3D, get_dCbc_dCfc, get_Phi_we, siendo la función multiply la que utiliza más tiempo de CPU, ya que utiliza el 15.63 % del tiempo total de ejecución, seguida por get_Irc con el 12.36% y fd3d_diffusion_2Fick_3D con el 7.58% por lo cual la implementación paralelizada utilizando OpenAcc de la simulación en versión GPU se concentró principalmente en estas funciones, ya que al ser optimizadas para GPU producirán una mejora sustancial sobre el tiempo de ejecución total de la simulación.

Implementación del algoritmo paralelizado en OpenACC

Utilizando las técnicas de paralelización y las librerías OpenACC, mediante el compilador suministrado por PGI ya que es el más robusto que se encuentra actualmente, además de estar instalado en la máquina Guane de la Universidad Industrial de Santander, se desarrolló el proceso de implementación de los códigos utilizando técnicas de paralelización del algoritmo de simulación basado en el modelo de difusión de cloruros combinado con la difusión de carbonatos en el concreto reforzado expuesto a un ambiente marino, para su respectiva ejecución en arquitecturas basadas en múltiples GPU.

Se observó que el mayor tiempo de ejecución lo consume la función de get_multiply donde multiplica uno a uno dos vectores que tienen la dimensión del cubo de datos. Esta función se paralelizó utilizando las directivas de OpenAcc, el código es mostrado en la figura 11.

Figura 11. Código paralelizado de la función get_multiply.

```
float *get_multiply(int nx, int ny, int nz, float *A, float *B)
{
    float *result;
    long index;

    result = ( float * ) malloc ( nx * ny * nz * sizeof ( float ) );

    #pragma acc kernels copyin(A[0:nx * ny * nz], B[0:nx * ny * nz]) copyout(result[0:nx * ny * nz])
    #pragma acc loop independent collapse(2) gang
    for(int i = 0; i < nx ; i++ )
    {
        for (int j = 0; j < ny ; j++ )
        {
            #pragma acc loop independent vector
            for (int k = 0; k < nz ; k++ )
            {
                index=i*ny+j+k*(nx*ny);
                result[index] = A[index]*B[index];
            }
        }
    }
    return result;
}
```

Otra función que consume gran cantidad de tiempo de máquina es get_Irc, la cual también se optimizó quedando de la forma como se muestra en la figura 12.

Figura 12. Código paralelizado de la función get_Irc.

```
float *get_Irc(int nx, int ny, int nz, float *dCbC_dCfc, float *PhiWe, float *Carbonate, float Ccao0, float Ich, float Icsh, float *denIrc)
{
    float *result;
    float alphaL = 11.8;
    float bbetaL = 4.0;
    float d = 0.88;
    float Mcl = 0.35453 ; // Kg/mol
    float b = 1.0;
    long index;

    result = ( float * ) malloc ( nx * ny * nz * sizeof ( float ) );

    #pragma acc kernels copyin(dCbC_dCfc[0:nx * ny * nz], PhiWe[0:nx * ny * nz], Carbonate[0:nx * ny * nz]) copyout(result[0:nx * ny * nz]) \
        copy(denIrc[0:nx * ny * nz])
    #pragma acc loop independent collapse(2) gang
    for(int i = 0; i < nx ; i++ )
    {
        for (int j = 0; j < ny ; j++ )
        {
            #pragma acc loop independent vector
            for (int k = 0; k < nz ; k++ )
            {
                index = i*ny+j+k*(nx*ny);
                denIrc[index] = 1.0+1.0/PhiWe[index]*dCbC_dCfc[index];
                result[index] = (d*alphaL*PhiWe[index]*Carbonate[index]*(Ich+Icsh))/(denIrc[index]*(1.0+bbetaL*Mcl*PhiWe[index]*Carbonate[index]/b)*Ccao0);
            }
        }
    }
    return result;
}
```

Y así se hizo con las funciones que tenían más consumo de tiempo de ejecución como se puede ver en la figura 13 parte del código de la paralelización de la función `fd3d_diffusion_2Fick_3D`, la cual tiene como parámetros de entrada:

- `Int x_num`: número de valores en el eje x del concreto.
- `float x[]` : vector con las coordenadas x correspondiente a cada división.
- `y_num`: número de valores en el eje y del concreto.
- `float y[]` : vector con las coordenadas y correspondiente a cada división.
- `z_num`: número de valores en el eje z del concreto.
- `float z[]` : vector con las coordenadas z correspondiente a cada división.
- `Int t`: tiempo actual de la ejecución.
- `float dt`: delta de tiempo.
- `float cflx`: valor equivalente a $\frac{dt}{dx^2}$
- `float cfly`: valor equivalente a $\frac{dt}{dy^2}$
- `float cflz`: valor equivalente a $\frac{dt}{dz^2}$
- `float *coeff_Diff`: llamado a la función que calcula el coeficiente de difusión de ión que se está calculando.
- `float *Cion`: cubo de datos que contiene la concentración del ión a procesar.
- `float valueBond`: valor de frontera del ión procesado.
- `void bc` : llamado a la función que establece los valores de frontera del ion que se está calculando en el concreto.

Figura 13. Código paralelizado de la función fd3d_diffusion_2Fick_3D.

```
float *fd3d_diffusion_2Fick_3D ( int x_num, float x[], int y_num, float y[], int z_num, float z[], float t, float dt,
float cflx, float cfly, float cflz, float * Coeff_Diff, float *V, float valueBond,
void bc ( int x_num, int y_num, int z_num, float *V, float value ) )
{
    float *V_new;
    int j, i, k;
    long index_ijk, index_ijpk, index_ijnk;
    long index_injk, index_ipjk;
    long index_ijkn, index_ijkn2, index_ijkp, index_ijkp2;
    V_new = ( float * ) malloc ( x_num * y_num * ( z_num ) * sizeof ( float ) );
    k=0;
    #pragma acc kernels copyin(V[0:x_num * y_num * z_num], Coeff_Diff[0:x_num * y_num * z_num]) copyout(V_new[0:x_num * y_num * z_num])
    #pragma acc loop independent
    for(i = 1; i < x_num - 1; i++)
    {
        #pragma acc loop independent collapse(2) gang
        for(k = 1; k < z_num - 1; k++)
        {
            for(i = 1; i < x_num - 1; i++)
            {
                #pragma acc loop independent vector
                for ( j = 1; j < y_num - 1; j++)
                {
                    {
                }
            }
        }
        k=z_num-1;
        #pragma acc loop independent
        for(i = 1; i < x_num - 1; i++)
        {
            #pragma acc loop independent
            for ( j = 1; j < y_num - 1; j++)
            {
                {
            }
        }
    }
    bc ( x_num, y_num, z_num, V_new, valueBond );
    return V_new;
}
```

Para identificar el rendimiento obtenido por cada función se utilizó la herramienta profiling para aplicaciones OpenACC llamada pgprof. El comando para compilar el programa y permite generar el archivo de profiling es:

```
pgc++ -std:c++11 -acc -ta=multicore,tesla -Minfo=accel -Mprof=ccff
fd3d_diffusion_prb.cpp fd3d_diffusion.cpp -o FD3D_Diffusion_OpenACC
```

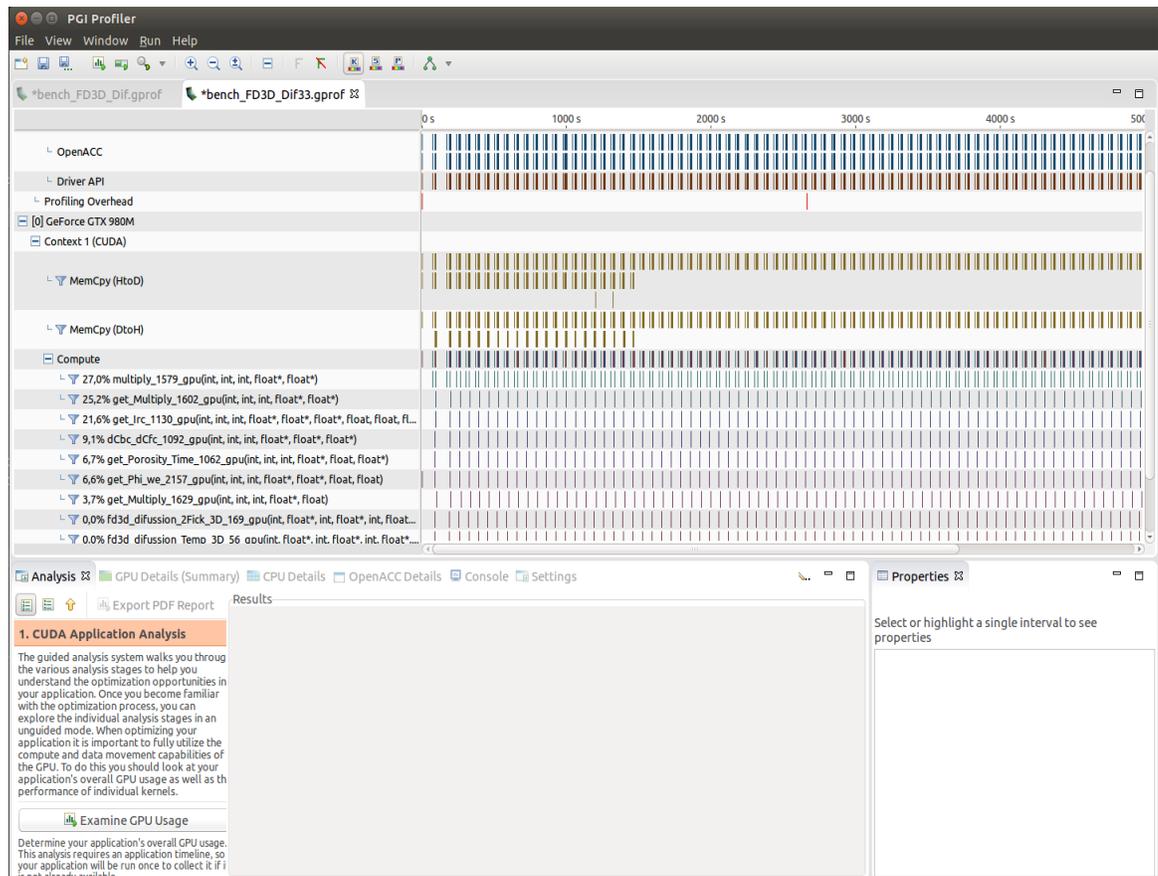
Una vez compilado es necesario ejecutar el programa compilado con la opción de generación del archivo de profiling mediante el comando:

```
pgprof --cpu-profiling off --unified-memory-profiling off -f -o bench_FD3D_Dif.pgprof
./FD3D_Diffusion_OpenACC
```

El programa se ejecuta y una vez finaliza su ejecución se genera el archivo de profiling bench_FD3D_Dif.pgprof enviado como parámetro de salida el cual contiene

los tiempos de ejecución de cada una de las funciones, para visualizarlo se usa la herramienta PGI Profiler, la cual muestra una gráfica del tiempo de ejecución del programa diferenciando los tiempos para cada una de las funciones que intervienen en el algoritmo como se muestra en la figura 14.

Figura 14. Visualización de los tiempos de ejecución mediante PGI Profiler.



La función write_vtk3D que se encarga de escribir el archivo de visualización en formato VTK para cada una de las iteraciones y permite la visualización de cómo avanza el cloruro en el concreto, no se le aplicó paralelización ya que es imposible aplicársela pues lo que hace es escribir un archivo VTK a disco duro.

En las pruebas de velocidades y aceleración se comentó la escritura a disco de los archivos VTK ya que ralentizaban la ejecución, además de requerir enorme capacidad de disco, llegando a ocupar más de 100 gigas de disco cada archivo cuando se usaban cubos de datos de 1024x1024x400.

La función `write_vtk3D` tiene como parámetros de entrada:

`output_filename` : nombre del archivo VTK a escribir.

`n`: número de datos en x.

`x[]`: vector que contiene las coordenadas x.

`m`: número de datos en y.

`y[]`: vector que contiene las coordenadas y.

`p`: número de datos en z.

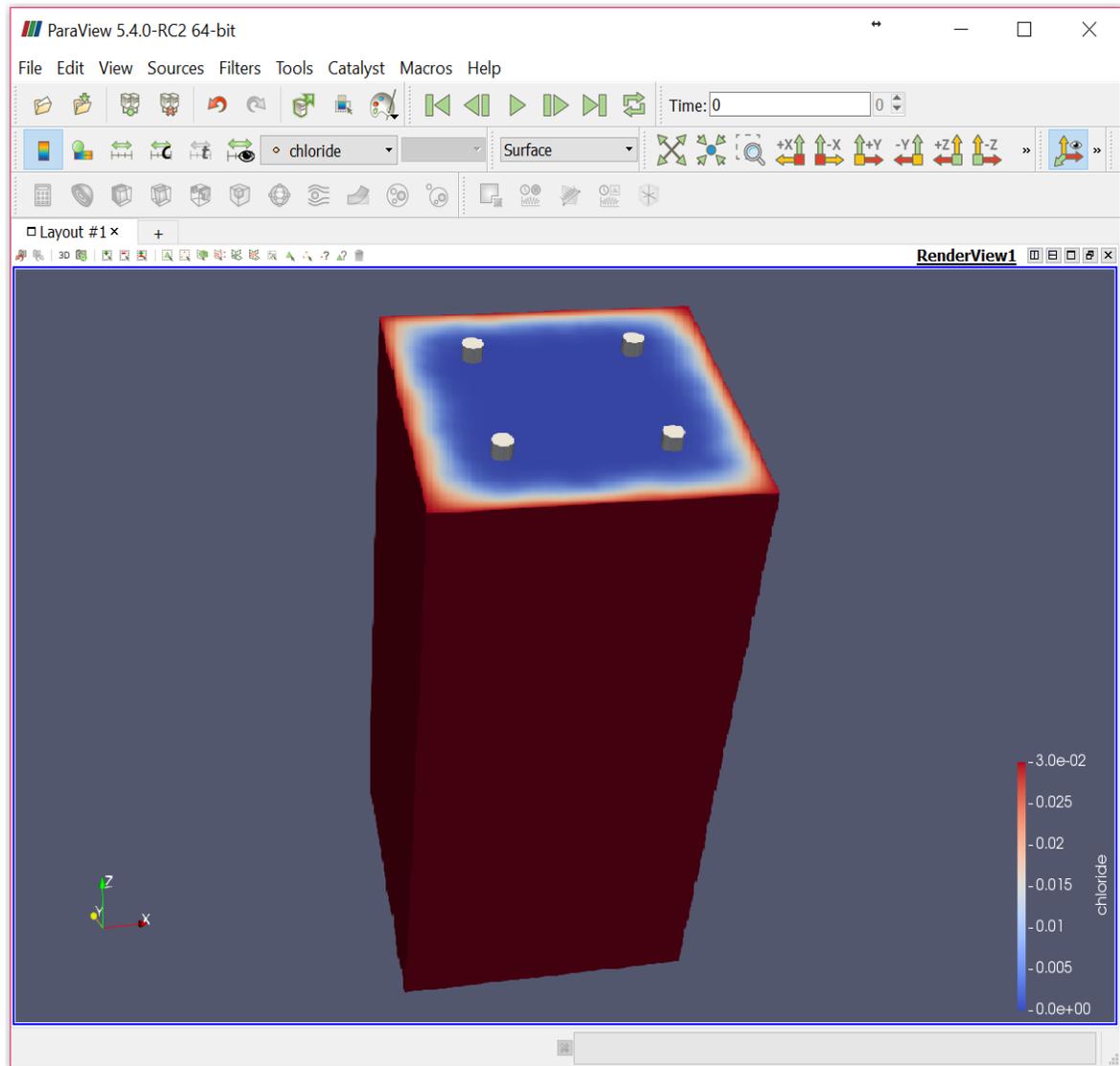
`z[]`: vector que contiene las coordenadas z.

`*c`: puntero a los datos que contienen el cubo de datos para cada una de las variables que almacena propiedades del concreto.

Visualización mediante ParaView

Paraview es una aplicación de distribución gratuita desarrollada por Kitware, que permite el análisis de datos y visualización de los mismos. Los usuarios de ParaView pueden construir rápidamente visualizaciones en 2D o 3D para analizar sus datos utilizando técnicas cualitativas y cuantitativas. ParaView fue desarrollada para analizar conjunto de datos extremadamente grandes usando recursos de memoria distribuida, por lo cual funciona en superordenadores para analizar conjunto de datos a nivel de teraescala, así como en computadores portátiles para conjunto de datos más pequeños. En la figura 15 se puede visualizar la interface de Paraview en donde se está visualizando los archivos VTK generados por el algoritmo de simulación de la corrosión del concreto.

Figura 15. Interface Paraview visualización de la difusión del cloruro en el concreto.



6.2.3. Pruebas de rendimiento obtenido

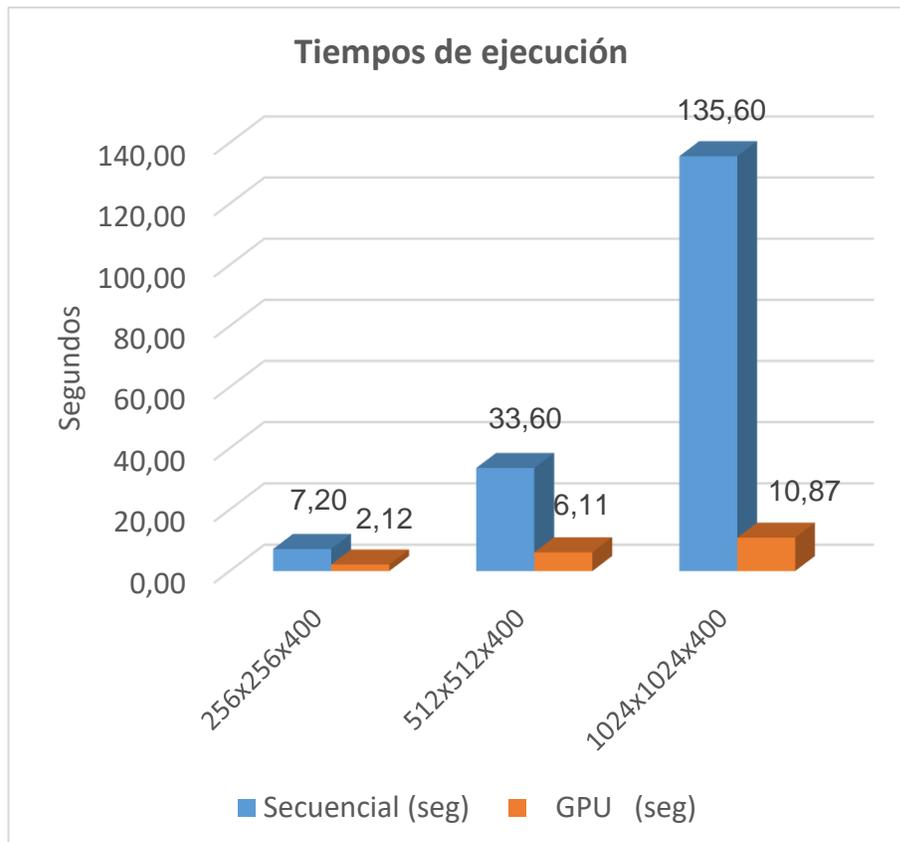
Una vez implementado el algoritmo en versión secuencial como versión paralela se procedió a realizar pruebas de rendimiento mediante las herramientas de profiling anteriormente mostradas, como son pgprof para análisis de rendimiento de la versión secuencial y PGI Compiler para la versión paralela, con el fin de hacer las

comparaciones pertinentes en cuanto a velocidad de ejecución para un ciclo, realizando corridas para diferentes dimensiones del cubo de datos, obteniendo los tiempos de ejecución mostrados en la tabla 1 y en la figura 16.

Tabla 1. Tiempos de ejecución.

Tamaño Cubo	Secuencial (seg)	GPU (seg)
256x256x400	7.20	2.12
512x512x400	33.60	6.11
1024x1024x400	135.60	10.87

Figura 16. Comparación de tiempos de ejecución.



Se extendió las pruebas para un determinado número de iteraciones y se lograron aceleraciones bastante notables en el total de los cálculos involucrados en el desarrollo del modelo de difusión de cloruros. Obteniendo los tiempos de ejecución mostrados en la tabla 2.

Tabla 2. Tiempos de ejecución obtenidos en el algoritmo de simulación.

Tamaño Cubo	Iteraciones	Secuencial (seg)	GPU (seg)
256x256x400	8000	57600	16457
256x256x400	16000	115200	33882
256x256x400	32000	230400	64901
256x256x400	64000	460800	132795
512x512x400	8000	268800	48873
512x512x400	16000	537600	94316
512x512x400	32000	1075200	194079
512x512x400	64000	2150400	396387
1024x1024x400	8000	1084800	86784
1024x1024x400	16000	2169600	173986
1024x1024x400	32000	4339200	346859
1024x1024x400	64000	8678400	692610

En la figura 17, se muestra las aceleraciones obtenidas con la paralelización del algoritmo de simulación.

6.2.4. Resultados del modelo

Se ejecutó el algoritmo de simulación utilizando diferentes proporciones de la mezcla del concreto en una primera ejecución se utilizó 0.4 para la relación agua/cemento y los valores de entrada mostrados en la tabla 3, la concentración de cloruro y el umbral de corrosión calculado en la ubicación del refuerzo se muestran

en la figura 18. Según la definición, si la concentración de cloruro C_{ic} , llega a ser igual al valor umbral, C_{th} , se iniciará la corrosión del refuerzo, debido al cambio inducido por la carbonatación de la estructura del poro y la capacidad de unión de los iones cloruro, el coeficiente de difusión aparente de los iones cloruro en la zona carbonatada. Mientras tanto, los iones de cloruro unidos se liberan a la solución de poro durante el proceso de carbonatación, estos efectos aumentan la concentración de iones de cloruro libres.

Figura 17. Aceleración obtenida para la simulación en GPU.

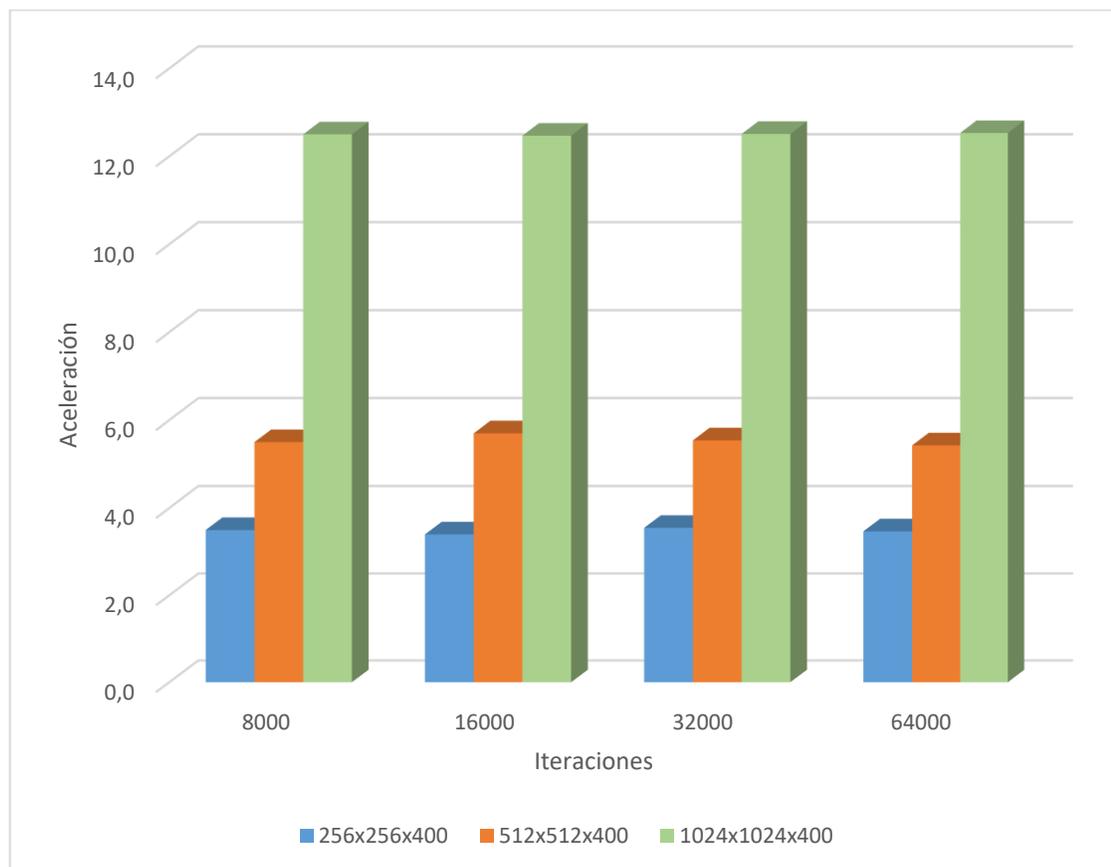
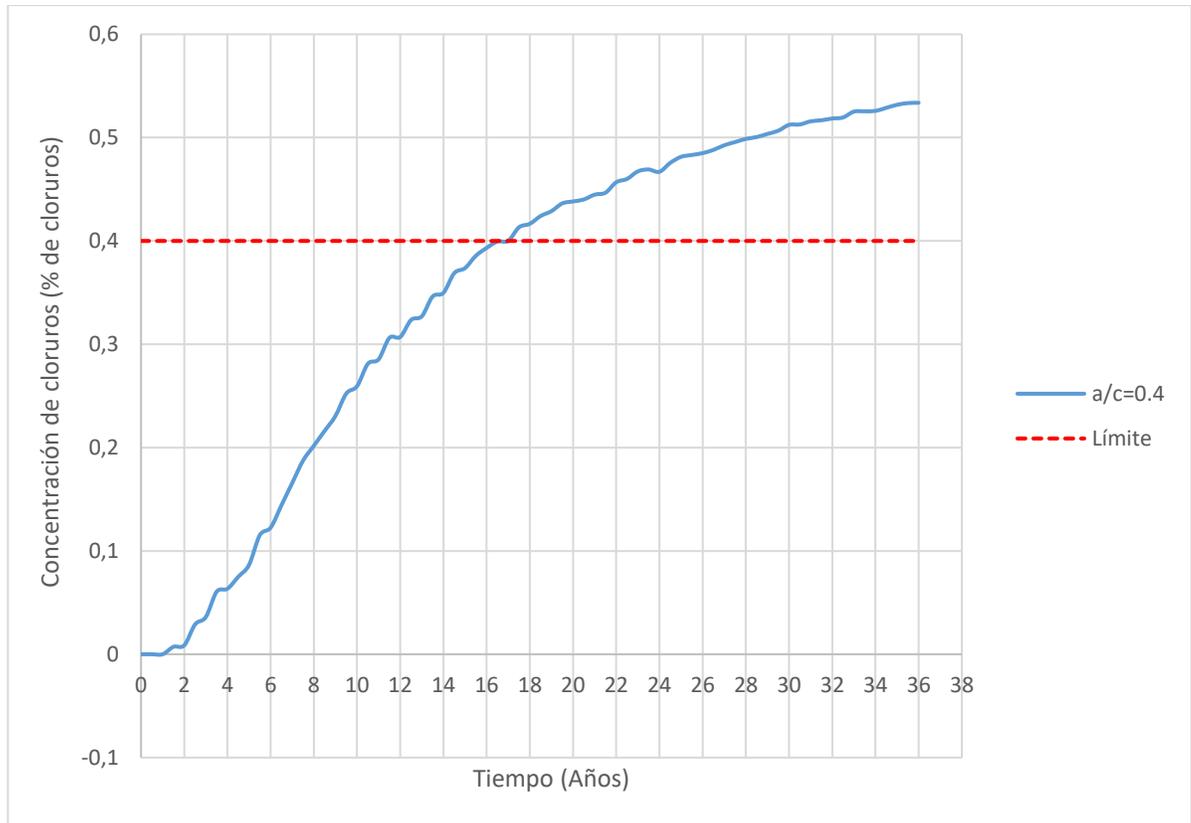


Tabla 3. Valores de entrada al algoritmo de simulación.

Variables de Entrada	Valor
Relación a/c	0.4
Relaciona g/c	6
Proporción molar de óxido de calcio en forma de CSH en concreto.	50%
Factor de reducción de la capacidad de unión de iones debido a la carbonatación	0.88
Coeficiente de transferencia de calor BT	7.75 W/(m ² K)
Coeficiente de transferencia de Cloruro superficial Bc	1 m/s
Coeficiente de transferencia de humedad superficial Bh	3*10 ⁻⁷ m/s
Coeficiente de difusión de referencia del cloruro D _{fc,ref}	5*10 ⁻¹² m ² /s
Coeficiente de difusión de referencia de la humedad D _{h,ref}	3*10 ⁻¹⁰ m ² /s
Densidad del agua	997 Kg/m ³
Densidad de los agregados	2600 Kg/m ³
Densidad del cemento	3150 Kg/m ³
Función de la temperatura	Cosenoidal Tmax= 303.15 K y Tmin=263.15 K
Función de la humedad	Cosenoidal hmax= 0.8 y hmin=0.7
Concentración Cl ⁻ en el medio	30 Kg/m ³
Concentración CO ₂ en el aire	0.03%

Figura 18. Concentración de cloruros para $a/c=0.4$



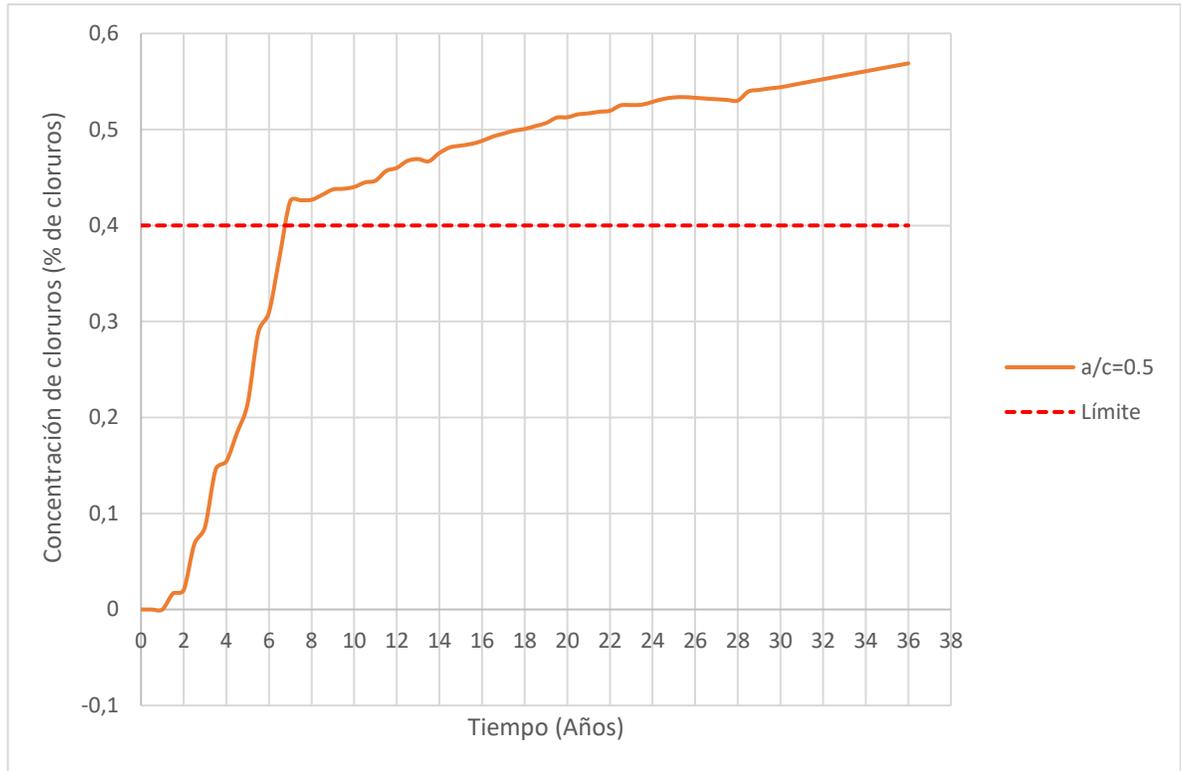
Como se puede apreciar en la gráfica para esta configuración del concreto se obtuvo un tiempo de despasivación de las estructuras del concreto cercano a los 16.5 años.

Se ejecutó una segunda simulación del algoritmo utilizando los valores mostrados en la tabla 4, obteniendo la gráfica de concentraciones mostrada en la figura 19. Como se puede apreciar se obtuvo tiempos de vida útil de la estructura de concreto cercana a los 7 años.

Tabla 4. Valores de entrada segunda simulación.

Variables de Entrada	Valor
Relación a/c	0.5
Relaciona g/c	6
Proporción molar de óxido de calcio en forma de CSH en concreto.	50%
Factor de reducción de la capacidad de unión de iones debido a la carbonatación	0.88
Coeficiente de transferencia de calor BT	7.75 W/(m ² K)
Coeficiente de transferencia de Cloruro superficial Bc	1 m/s
Coeficiente de transferencia de humedad superficial Bh	3*10 ⁻⁷ m/s
Coeficiente de difusión de referencia del cloruro D _{fc,ref}	5*10 ⁻¹² m ² /s
Coeficiente de difusión de referencia de la humedad D _{h,ref}	3*10 ⁻¹⁰ m ² /s
Densidad del agua	997 Kg/m ³
Densidad de los agregados	2600 Kg/m ³
Densidad del cemento	3150 Kg/m ³
Función de la temperatura	Cosenoidal Tmax= 303.15 K y Tmin=263.15 K
Función de la humedad	Cosenoidal hmax= 0.8 y hmin=0.7
Concentración Cl ⁻ en el medio	30 Kg/m ³
Concentración CO ₂ en el aire	0.03%

Figura 19. Concentración de cloruros para $a/c=0.5$



Como se puede apreciar en la figura 19, se presenta una disminución en el tiempo de vida útil de la estructura de concreto, esto principalmente al aumento de la relación agua/cemento la cual hace que la estructura sea más porosa y por ende su tortuosidad sea menor, lo cual facilita el ingreso de los iones que favorecen el fenómeno de la corrosión.

Y una tercera ejecución utilizando los valores mostrados en la tabla 5, obteniendo un tiempo de inicio de la despasivación de las estructuras de refuerzo cercana a los 5 años como se muestra en la figura 20.

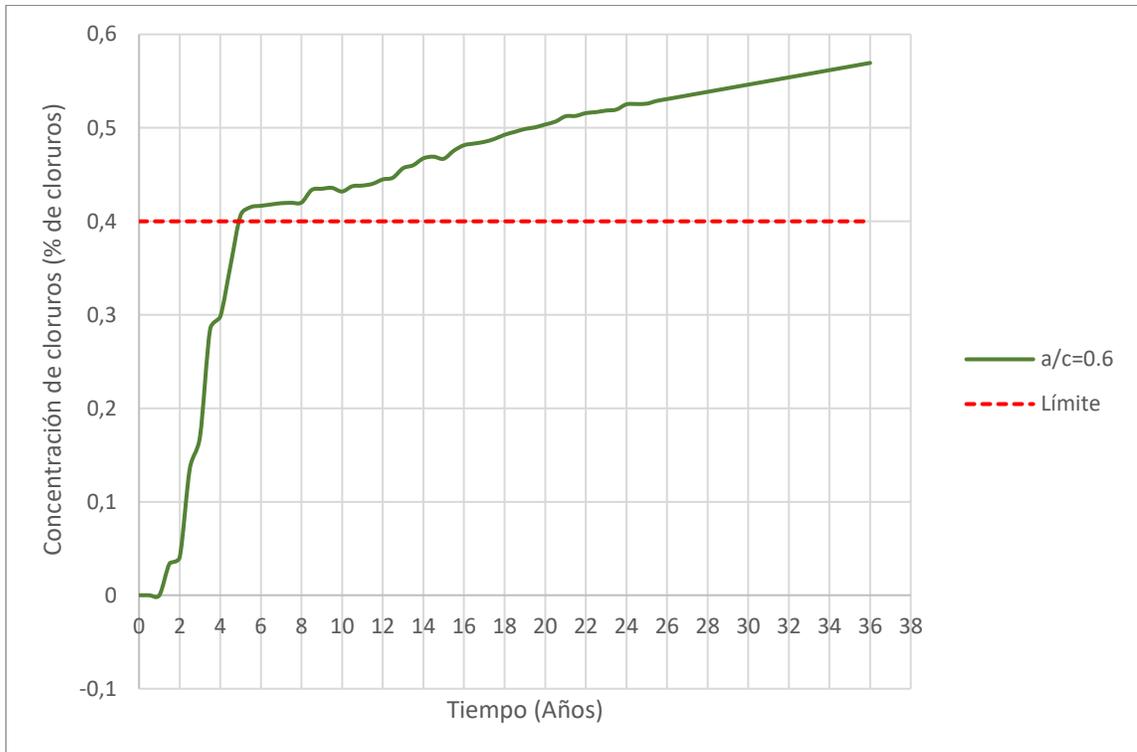
Tabla 5. Valores de entrada para la tercera simulación.

Variables de Entrada	Valor
Relación a/c	0.6
Relaciona g/c	6
Proporción molar de óxido de calcio en forma de CSH en concreto.	50%
Factor de reducción de la capacidad de unión de iones debido a la carbonatación	0.88
Coeficiente de transferencia de calor BT	7.75 W/(m ² K)
Coeficiente de transferencia de Cloruro superficial Bc	1 m/s
Coeficiente de transferencia de humedad superficial Bh	3*10 ⁻⁷ m/s
Coeficiente de difusión de referencia del cloruro D _{fc,ref}	5*10 ⁻¹² m ² /s
Coeficiente de difusión de referencia de la humedad D _{h,ref}	3*10 ⁻¹⁰ m ² /s
Densidad del agua	997 Kg/m ³
Densidad de los agregados	2600 Kg/m ³
Densidad del cemento	3150 Kg/m ³
Función de la temperatura	Cosenoidal Tmax= 303.15 K y Tmin=263.15 K
Función de la humedad	Cosenoidal hmax= 0.8 y hmin=0.7
Concentración Cl ⁻ en el medio	30 Kg/m ³
Concentración CO ₂ en el aire	0.03%

Como se puede apreciar en la figura 20, al aumentar la relación agua/cemento se ve afectada la estructura de concreto reforzado ya que disminuye su tiempo de vida

útil, esto se debe al aumento de la porosidad de la estructura y por ende la facilidad de ingreso de los iones que favorecen el fenómeno de la corrosión.

Figura 20. Concentración de cloruros para $a/c=0.6$



7. Conclusiones

La aplicación de técnicas de computación de alto rendimiento sobre arquitecturas basadas en múltiples GPU, permite obtener resultados de la estimación del avance del ion cloruro dentro de una estructura de concreto de una forma más rápida y más exacta ya que permite manejar muchos más factores que la afectan, además de un mallado más denso el cual permite obtener resultados mucho más exactos.

La aplicación de computación de alto rendimiento se hace indispensable en los centros de investigación ya que permite validar de forma más rápida diferentes modelos complejos sin el inconveniente de la espera de los altos tiempos de ejecución de los mismos.

La aplicación del modelo planteado en este proyecto es una herramienta muy práctica que permite evaluar el avance del ión Cl^- a través de las estructuras de concreto y así determinar si se ha alcanzado la concentración necesaria para el inicio de la despasivación del acero de refuerzo o el tiempo aproximado para que esto suceda.

Para las estructuras simuladas se puede decir que la vida útil del refuerzo disminuye cuando aumenta la relación a/c debido a que la difusión de iones cloruro se hace más rápida a causa de la mayor porosidad y menos tortuosidad del concreto.

BIBLIOGRAFÍA

A. DELNAVAZ A. Ramezaniapour The assessment of carbonation effect on chloride diffusion in concrete based on artificial neural network model [Sección de libro]. - 2012.

A. MORANDEAU M. THIERY, Dangla Investigation of the carbonation mechanism of CH and C–S–H in terms of kinetics, microstructure changes and moisture properties [Publicación periódica] // Cem. Concr. Res. 56. - 2014. - págs. 153–170.

A. SAETTA R. SCOTTA, R. Vitaliani Mechanical behavior of concrete under physical–chemical attacks [Publicación periódica] // J. Eng. Mech. ASCE 124 (10) . - 1998. - págs. 1100–1109.

A. SAETTA R. Vitaliani Experimental investigation and numerical modeling of carbonation process in reinforced concrete structures. Part I. Theoretical Formulation [Sección de libro]. - 2004.

A. SAETTA R. Vitaliani Experimental investigation and numerical modeling of carbonation process in reinforced concrete structures: Part II. Practical applications. [Sección de libro]. - 2005.

A. STRAUSS R. WENDNER, K. Bergmeister, C. Costa Numerically and experimentally based reliability assessment of a concrete bridge subjected to chloride-induced deterioration, J. Infrastruct [Sección de libro]. - 2013.

Alejo. K. Modelación Numérica de la penetración de Iones Cloruro en el Hormigón. La Habana, Cuba, 2005.

ARMSTRONG M. P. AND P. J. DENSHAM Domain decomposition for parallel processing of spatial problems. Computers, Environment and Urban Systems [Sección de libro]. - 1992.

B. MARTIN, PEREZ S. Pantazopoulou, M. Thomas [Sección de libro] // Numerical solution of mass transport equations in concrete structures. - 2001.

B. SAASSOUH Z. Lounis Probabilistic modeling of chloride-induced corrosion in concrete structures using first- and second-order reliability methods [Sección de libro]. - 2012.

BAZANT Z.P. Physical model for steel corrosion in concrete sea structures theory [Sección de libro] // , J. Struct. Div. ASCE 105 . - 1979.

BÖHNI H. Corrosion in reinforced concrete structures [Sección de libro]. - Cambridge : Woodhead Publishing Ltd y CRC Press, 2005.

BREWER. E. Portable High Performance Supercomputing: High Level Platform Dependent Optimization [Sección de libro]. - [s.l.]: Massachusetts Institute of Technology, 1994.

C. NOGUEIRA E. Leonel Probabilistic models applied to safety assessment of reinforced concrete structures subjected to chloride ingress [Sección de libro]. - 2013.

C. YUAN D. NIU, D. Luo Effect of carbonation on chloride diffusion in fly ash concrete [Sección de libro]. - 2012.

C. YUAN D. NIU, N. Chen, F. Duan Influence of carbonation on the microstructure of concrete [Publicación periódica] // Bull. Chin. Ceram. Soc. 32 (4) . - 2013. - págs. 687–691.

CAIMS J. State of the art report on bond of corroded reinforcement [Informe]. - 1998.

CASTELLOTE M., Andrade, C. & Alonso, C. Comparison with natural diffusion tests [Sección de libro] // Measurement of the steady and nonsteady-state chloride diffusion coefficients in a migration test by means of monitoring the conductivity in the anolyte chamber. - [s.l.] : Cement and Concrete Research, 2001.

COMMITTEE 365 American Concrete Institute Service Life Prediction - State of the Art Report [Conferencia]. - 2010.

CRANK J. The mathematics of Diffusion [Libro]. - Oxford : Oxford University Press, 1976.

D. NIU L. CHEN, C. Zhang Computational model of gas diffusion coefficient in concrete [Publicación periódica] // J. Xian Univ. Arch. Technol. (Nat. Sci. Ed.) 39 (6). - 2007. - págs. 741–745.

D. VAL P. Trapper Probabilistic evaluation of initiation time of chlorideinduced corrosion [Publicación periódica] // Reliab. Eng. Syst. Saf. 93. - 2008. - págs. 364–372.

D.W.S. HO R.K. Lewis Carbonation of concrete and its prediction [Sección de libro]. - 1987.

DEL VALLE MORENO A. y Pérez López T. y Maerínez Madrid, M. El Fenómeno de la Corrosión en Estructuras de Concreto Reforzado. [En línea]. - 2001. - <http://www.imt.mx/archivos/Publicaciones/PublicacionTecnica/pt182.pdf> .

E. BASTIDAS, ARTEAGA A. Chateauneuf, M. Sanchez-Silva, P.H. Bressolette, F. Schoefs A comprehensive probabilistic model of chloride ingress in unsaturated concrete [Publicación periódica] // Eng. Struct. 33 (3) . - 2011. - págs. 720–730.

E. BASTIDAS, ARTEAGA F. Schoefs, M. Stewart, X. Wang Influence of global warming on durability of corroding RC structures: a probabilistic approach [Publicación periódica] // Eng. Struct. 51 . - 2013. - págs. 259–266.

E. TOLENTINO F. LANMEIRAS, A. GOMES, C. Silva, W. Vasconcelos Effects of high temperature on the residual performance of portland cement concretes [Publicación periódica] // Mater. Res. 5 (3) . - 2002. - págs. 301–307.

E. TOLENTINO F. LANMEIRAS, A. GOMES, C. Silva, W. Vasconcelos Structural evaluation and performance of portland cement concretes after exposure to high temperatures [Publicación periódica] // Mater. Res. 5 (1). - 2002. - págs. 27–36.

F. DUPRAT N. Vu A probabilistic threshold for the onset of carbonation-induced corrosion [Publicación periódica] // Eur. J. Environ. Civ. Eng. 17 . - 2013. - págs. 478–495.

F. DUPRAT N. Vu, A. Sellier Accelerated carbonation tests for the probabilistic prediction of the durability of concrete structures [Publicación periódica] // Constr. Build. Mater. 66 . - 2014. - págs. 597–605.

F. DUPRAT N. Vu, A. Sellier Accelerated carbonation tests for the probabilistic prediction of the durability of concrete structures [Publicación periódica] // Constr. Build. Mater. 66 . - 2014. - págs. 597–605.

F. LOLLINI E. REDAELLI, L. Bertolini Analysis of the parameters affecting probabilistic predictions of initiation time for carbonation-induced corrosion of reinforced concrete structures [Publicación periódica] // Mater. Corros. 63 (12) . - 2012. - págs. 1059–1068.

FERREIRA R. M. Probability based durability analysis of concrete structures in marine environment [Sección de libro]. - Braga, Portugal : University of Minho, 2004.

FONTANA M. G. [Sección de libro]// Corrosion engineering. - Nueva York : Mc GrawHill, 1986.

FOSTER I. Designing and building parallel programs [Sección de libro]. - [s.l.] : Addison Wesley Publishing Company, 2003.

HANS B. Corrosion in Reinforced Concrete Structures [Libro]. - England : Woodhead Publishing Limited, 2005.

HPCwire OpenACC Group Reports Expanding Support for Accelerator Programming Standard [Libro]. - 2012.

HUANG T. The experimental research on the interaction between concrete carbonation and chloride ingress under loading [Informe]. - [s.l.] : Zhejiang University, 2013.

J. BACKUS D. Mcpolin, M. Basheer, A. Long, N. Holmes Exposure of mortars to cyclic chloride ingress and carbonation [Sección de libro]. - 2013.

J. KONG A. Ababneh, D. Frangopol, Y. Xi Reliability analysis of chloride penetration in saturated concrete [Sección de libro]. - 2002.

J. LIU Q. Qiu, F. Xing, D. Pan Permeation properties and pore structure of surface layer of fly ash concrete [Publicación periódica] // Materials 7 (6) . - 2014. - págs. 4282–4296.

J. MAI, NHU A. Sellier, F. Duprat, P. Rougeau, B. Caprad, N. Hyvert, P. Francisco Probabilistic approach for durable design of concrete cover: application to carbonation [Publicación periódica] // J. Environ. Civ. Eng. 16 (3–4) . - 2012. - págs. 264–272.

J. MARCHAND, E. SAMSON Predicting the service-life of concrete structures – Limitations of simplified models [Sección de libro].

JOHANNESSON B. F. [Sección de libro] // A theoretical model describing diffusion of a mixture of different types of ions in pore solution of concrete coupled to moisture transport. - [s.l.] : Cement and Concrete Research, 2003.

JÖHNK K. HUTTER y K Continuum Methods of Physical Modeling [Libro]. - New York : Springer-Verlag Berlin Heidelberg , 2004.

K. NAKARAI T. ISHIDA, K. Maekawa Multi-scale physicochemical modeling of soil-cementitious material interaction [Publicación periódica] // Soils Found. 46 (5). - 2006. - págs. 653–663.

KHITAB A Modélisation des transferts ioniques dans les milieux poreux saturés: application à la pénétration des chlorures à travers les matériaux cimentaires [Libro]. - Toulouse : l'institut national des science appliquees de Toulouse, 2005.

KITWARE VTK [En línea]. - Kitware, 2017. - <https://www.vtk.org/>.

Li L PAGE C, Wang Y. Modelling of chloride ingress into concret from saline environment. [Sección de libro]. - 2005.

Li YAU Shu Wong and Guangrui [Sección de libro] // Exact Finite Difference Schemes for Solving Helmholtz Equation at any Wavenumber. - [s.l.] : International Journal of Numerical Analysis and Modeling, 2011.

LIZARAZO MARRIAGA J. M. Transport properties and multi-species modelling of slag based concretes [Libro]. - [s.l.] : Coventry University, 2010.

M. LEE S. JUNG, B. Oh Effects of carbonation on chloride penetration in concrete [Sección de libro]. - 2013.

M. STEWART J. MULLARD. Spatial time-dependent reliability analysis of corrosion damage and the timing of first repair for RC structures [Sección de libro]. - 2007.

M. STEWART Q. SUO. Extent of spatially variable corrosion damage as an indicator of strength and time-dependent reliability of RC beams [Sección de libro]. - 2009.

M. THIERY P. FAURE, A. MORANDEAU, G. PLATRET, J. BOUTELOUP, P. BELIN
Effect of carbonation on the microstructure and moisture properties of cement-based materials [Publicación periódica] // 12th International Conference on Building Materials and Components. - 2011. - págs. 1–8.

M. THIERY V. BAROGHEL-BOUNY, A. MORANDEAU, P. DANGLA. Impact of Carbonation on the Microstructure and Transfer Properties of Cement-based Materials [Publicación periódica]. - 2012. - págs. 1–10.

M.D.A. THOMAS P.B. BAMFORTH, Modelling chloride diffusion in concrete: effect of fly ash and slag [Sección de libro]. - [s.l.] : Cem. Concr. Res. 29, 1999.

MEJÍA De GUTIÉRREZ R. Durabilidad y Corrosión en Materiales Cementicios [Libro]. - Cali, Colombia : Universidad del Valle, 1999.

MINDESS S. , YOUNG, J.F Concrete [Libro]. - Nueva Jersey : Prentice Hall, 1981.

NARSILIO G. A., Li, R., PIVONKA, P. & SMITH, D. W. [Sección de libro] // Comparative study of methods used to estimate ionic diffusion coefficients using migration tests. - [s.l.] : Cement and Concrete Research, 2007.

NEVILLE A. Properties of Concrete [Libro]. - London and New York : Longman and John Wiley, 1995.

NVIDIA ¿Qué es el GPU Computing acelerado? [En línea]. - NVIDIA, 2017. - <http://www.nvidia.es/object/gpu-computing-es.html>.

NVIDIA Accelerated Computing [En línea]. - 2016. - <https://devblogs.nvidia.com/parallelforall/developing-new-materials-gpu-accelerated-supercomputers/>.

NVIDIA Diseño basado en la realidad. [En línea]. - NVIDIA, 2017. - <http://www.nvidia.es/object/multi-gpu-for-design-manufacturing-es.html>.

NVIDIA GPU-Accelerated Applications [En línea]. - 2016. - <http://www.nvidia.es/docs/IO/123576/nv-applications-catalog-lowres.pdf>.

O. ISGOR, A. RAZAQPUR Finite element modeling of coupled heat transfer, moisture transport and carbonation processes in concrete structures [Sección de libro]. - 2004.

OFFICIALS AMERICAN ASSOCIATION of State Highway and Transportation Standard Method of Test for Electrical Indication of Concrete's Ability to Resist Chloride Ion Penetration [Libro]. - 2007.

OPENACC Librerías OpenACC [En línea]. - OpenACC, 2017. - <https://www.openacc.org/>.

P. CHINDAPRASIRT, C. JATURAPITAKKUL, T. Sinsiri Effect of fly ash fineness on microstructure of blended cement paste [Publicación periódica] // Constr. Build. Mater. 21 (7) . - 2013. - págs. 1534–1541.

P. CHINDAPRASIRT, S. RUKZON, V. SIRIVIVATNANON. Effect of carbon dioxide on chloride penetration and chloride ion diffusion coefficient of blended Portland cement mortar [Sección de libro]. - 2008.

P. RYAN, A. O'CONNOR. Probabilistic analysis of the time to chloride induced corrosion for different self-compacting concretes [Sección de libro]. - 2013.

PEREIRA Alexandre da Costa Estudio de métodos probabilísticos para la predicción de la vida útil de estructuras de hormigón. [Conferencia]. - Madrid - España : [s.n.], 2003.

PÉREZ B. M. A study of th effect of chloride binding on service life predictions. [Sección de libro] // Cement and Concrete Research 30. - 2000. - Vol. 30.

PGI OpenACC Getting Started Guide [En línea]. - PGI Compilers & Tools, 2017. - https://www.pgroup.com/resources/docs/17.10/pdf/openacc17_gs.pdf.

Q. SUO, M. STEWART. Corrosion cracking prediction updating of deteriorating RC structures using inspection information [Publicación periódica] // Reliab. Eng. Syst. Saf. 94 . - 2009. - págs. 1340–1348.

Q. ZENG K. Li, F. TEDDY, P. Dangla Pore structure characterization of cement pastes blended with high-volume fly-ash [Publicación periódica] // Cem. Concr. Res. 42 (1) . - 2012. - págs. 194– 204.

QIANG Y. Chloride binding of cement-based materials subjected to external chloride environment – A review [Publicación periódica] // Construction and Building Materials. - 2009. - 1 : Vol. 23. - págs. 1-13.

S. BRUNAUER J. SKALNY, E. BODOR. Adsorption on non-porous solids [Publicación periódica] // J. Colloid Interface Sci. 30 (4) . - 1969. - págs. 546–552.

SELOUMA Tarek Ibrahim Mahmoud. Evaluacion de la degradación de prefabricados de hormigón sometidos a ambientes marinos mediante técnicas no destructibles y análisis físico químico. [Publicación periódica]. - 2009. - págs. 3-5.

SERRANO Juan Francisco , CAMPOS Rodolfo Morphskel. Esqueleto programático basado en morfología matemática para el tratamiento digital de imágenes. [En línea]. - 2014. - <http://www.redalyc.org/pdf/4277/427739431012.pdf>.

SÍSMICA Asociación Colombiana de Ingeniería Reglamento Colombiano de Construcción Sismo Resistente NSR-10 [Libro]. - Colombia : [s.n.], 2012.

T. ISHIDA, P. IQBAL, H. ANH. Modeling of chloride diffusivity coupled with nonlinear binding capacity in sound and cracked concrete [Sección de libro] // Cem. Concr. Res. 39. - 2009.

T. ISHIDA P. IQBAL, H. ANH. Modeling of chloride diffusivity coupled with nonlinear binding capacity in sound and cracked concrete [Publicación periódica] // Cem. Concr. Res. 39 (10) . - 2009. - págs. 913–923.

T. LARRARD, E. BASTIDAS-ARTEAGA, F. DUPRAT, F. SCHOEFS. Effects of climate variations and global warming on the durability of RC structures subjected to carbonation [Publicación periódica] // Civ. Eng. Environ. Syst. 31 . - 2014. - págs. 153–164.

TRUC O. [Sección de libro] // Prediction of Chloride Penetration into Saturated Concrete – Multi-species Approach. - [s.l.] : Chalmers University, 2000.

TUUTTI K Corrosion of Steel in Concrete [Libro]. - Stockholm, Swedish : The Swedish Cement and Concrete Association, 1982.

UIS Grupo de Investigación de corrosión Desarrollo metodológico electroquímico de la corrosividad de estructuras de concreto sometidas a los ambientes marinos de las costas del pacífico colombiano [Sección de libro]. - Bucaramanga, Colombia : [s.n.], 2009.

V. NGALA C. Page Effects of carbonation on pore structure and diffusional properties of hydrated cement pastes [Sección de libro]. - 1997.

V. PAPADAKIS, C. VAYENAS, M. FARDIS. A reaction engineering approach to the problem of concrete carbonation [Publicación periódica] // J. Am. Inst. Chem. Eng. 35 (10) . - 1989. - págs. 1639– 1650.

V. PAPADAKIS C. VAYENAS, M. FARDIS. Fundamental modeling and experimental investigation of concrete carbonation [Publicación periódica] // ACI Mater. J. 88 (4). - 1991. - págs. 363–373.

V. PAPADAKIS C. VAYENAS, M. FARDIS. Physical and chemical characteristics affecting the durability of concrete [Publicación periódica] // ACI Mater. J. 88 (2) . - 1991. - págs. 186–196.

VALDERRAMA. Iván. Blog de Corrosión [En línea]. - <http://nuevolugardemiblog.blogspot.com.co/2017/02/blode-corrosion.html>.

VARGAS Humberto Alejandro Girón [En línea]. - <http://www.imcyc.com/revista/1998/oct/ataque.htm> .

VELÁZQUEZ GONZALEZ R. Electrochemical evaluation of the corrosion in grade in reinforce steel in the presence of admixtures. [Publicación periódica] // Portugaliae Electrochimica Acta 23 . - [s.l.] : Portugaliae Electrochimica , 2005. - págs. 179-194.

W. PUATATSANANON V. SAOUMA. Nonlinear coupling of carbonation and chloride diffusion in concrete [Sección de libro]. - 2005.

X. WAN, F. WITTMANN, T. ZHAO, H. FAN. Chloride content and pH value in the pore solution of concrete under carbonation [Sección de libro]. - [s.l.] : J. Zhejiang University, 2013.

XATAKA Las GPU como pasado, presente y futuro de la computación [En línea]. - 2014. - <https://www.xataka.com/componentes/las-gpu-como-pasado-presente-y-futuro-de-la-computacion>.

Y. KIM, K. LEE, J. BANG, S. KWON. Effect of w/c ratio on durability and porosity in cement mortar with constant cement amount [Publicación periódica] // Adv. Mater. Sci. Eng.. - 2014. - págs. 1– 11.

YOON I. Deterioration of concrete due to combined reaction of carbonation and chloride penetration: experimental study [Sección de libro]. - 2007.

YOON I. Simple approach to calculate chloride diffusivity of concrete considering carbonation [Sección de libro]. - 2009.