



CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS
DEL INSTITUTO POLITÉCNICO NACIONAL

UNIDAD ZACATENCO

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

SECCIÓN COMPUTACIÓN

*Interfaces para un sistema de cómputo
que resuelve la ecuación de
Ornstein - Zernike*

Tesis que presenta

Luz Virginia Morales Morón

para obtener el grado de

Maestría en Ciencias

en la especialidad de

Ingeniería Eléctrica

con opción de

Computación

Director de la tesis

Guillermo Benito Morales Luna

México, D.F.

Junio 2004

Índice general

Resumen	VII
Abstract	IX
Agradecimientos	XI
Dedicatoria	XIII
1. Introducción	1
2. La ecuación de Ornstein-Zernike	5
2.1. Desarrollos alternativos actuales	5
2.2. Descripción de la ecuación	8
2.3. Diseño computacional	9
3. Tratamiento matemático y numérico	13
3.1. Planteamiento del problema	13
3.2. Coordenadas esféricas prolatas	16
3.2.1. Cambio de variable	16
3.2.2. Transformación de mallados	20
3.3. Procedimiento de integración numérica	20
3.3.1. Presentación del problema	20
3.4. Método de Newton-Raphson y convergencia global para el problema de Ornstein-Zernike	22
3.5. Principales parámetros involucrados en la resolución de la ecuación de Ornstein-Zernike	24
3.5.1. Proceso de especificación de mallas planares	24
3.6. Cálculo de funciones núcleo	32
3.6.1. Cálculo de raíces de operadores en términos de las funciones que calculan los núcleos de la ecuación de Ornstein-Zernike	35
3.7. Cálculo de las funciones que involucran a los procedimientos núcleo en términos de la ecuación de Ornstein-Zernike	39
3.8. Despliege de soluciones	42

4. Alternativa de diseño para el sistema de cómputo que resuelve la ecuación OZ	47
4.1. Antecedentes	47
4.2. Diseño del sistema de acceso al servidor <i>Web</i>	47
4.3. Base de datos	48
4.4. Diseño del sistema en lenguaje C	48
4.5. Esquema general del sistema que resuelve la ecuación de Ornstein-Zernike .	50
5. Diseño e implementación del sistema	53
5.1. Implementación del procesamiento numérico	53
5.2. Entradas y salidas	71
6. Conclusiones y trabajo futuro	73
6.1. Conclusiones	73
6.2. Trabajo futuro	74
A. Manual de usuario del sistema	75
A.1. Usuario autorizado	75
A.2. Usuario sin privilegios	83
B. Manual de administrador del sistema	91
B.1. Requerimientos de software	91
B.2. Carpetas que deben ser creadas	91
B.3. Manipulación de la base de datos	92
B.4. Interfaz del usuario administrador	92
Bibliografía	102

Índice de figuras

2.1. Volumen constante del calor específico C_v para SCOZA comparado con resultados exactos. Se nota que el calor específico para SCOZA no diverge hasta el máximo, muestran un semblanza de modelos de tamaño finito . . .	6
2.2. Residuales entre el SCOZA y resultados exactos. Para resultados óptimos de tamaño finito, las derivaciones son mas pequeñas que $\infty \times \infty$ de resultados exactos	7
2.3. Modelo de esfera dura de Yakuwa	7
2.4. La polarización MSA del parámetro $lambda$ para un ion bipolar mezclado como una función de concentración molar	8
3.1. Complejo α formado por dos partículas β y γ . Donde α esta localizada en el centro de masa de β y γ entonces $m_\alpha = m_\beta + m_\gamma$	14
3.2. Gráficas de la función M	18
3.3. Gráficas de la <i>primera</i> componente de Φ	18
3.4. Gráficas de la <i>segunda</i> componente de Φ	18
3.5. Gráficas de la <i>tercera</i> componente de Φ	18
3.6. Algunos efectos de la transformación Φ	21
3.7. Algunos otros efectos de la transformación Φ	21
3.8. Malla determinada por $P \times (Q_1 \cup Q_2)$	30
3.9. Malla $\Phi(P \times (Q_1 \cup Q_2))$	31
3.10. Malla $(P \times Q_1) \cup R$	31
3.11. Malla $\Phi((P \times Q_1) \cup R)$	31
3.12. Gráfica determinada por abc y wei , arreglos generados en $4arg$	39
3.13. Gráfica determinada por $\eta_1 = 2$ y $\xi_1 = 0.5$, nucleo1 positivo y negativo	42
3.14. Gráfica determinada por $\eta_1 = 1$ y $\xi_1 = 1$, nucleo1 positivo y negativo	42
3.15. Gráfica determinada por $\eta_1 = 21.1$ y $\xi_1 = 0.1$, nucleo1 positivo y negativo	43
3.16. Gráfica determinada por $\eta_1 = 15$ y $\xi_1 = 0.1$, nucleo2 positivo y negativo	43
3.17. Gráfica determinada por $\eta_1 = 1.1$ y $\xi_1 = 0.5$, nucleo2 positivo y negativo	44
3.18. Gráfica determinada por $\eta_1 = 1.1$ y $\xi_1 = 0.1$, nucleo3 positivo y negativo	44
3.19. Gráfica determinada por $\eta_1 = 3$ y $\xi_1 = 0.1$, nucleo3 positivo y negativo	44
3.20. Gráfica determinada por $\eta_1 = 15$ y $\xi_1 = 0.1$, nucleo3 positivo y negativo	44
4.1. Esquema general del sistema que resuelve la ecuación OZ	52
5.1. Elemento finito triangular de 7 nodos.	57

A.1. Pantalla de acceso.	76
A.2. Opciones de trabajo.	76
A.3. Entrada de datos.	77
A.4. Entrada de datos y coordenadas.	79
A.5. Generación de los datos de iniciales del procedimiento de Newton-Raphson.	80
A.6. Ejecución del sistema que resuelve la ecuación.	81
A.7. Mandar a ejecutar el proceso sin un nombre definido.	81
A.8. Proceso concluido.	82
A.9. Pantalla Consultar proceso.	82
A.10. Pantalla Borrar procesos.	83
A.11. Pantalla Consultar procesos por pares.	84
A.12. Pantalla Borrar procesos.	85
A.13. Datos de entrada a pruebas.	85
A.14. Datos de entrada y coordenadas a pruebas.	86
A.15. Mandar a hacer pruebas del proceso sin un nombre definido.	86
A.16. Tipos de pruebas.	87
A.17. Ejecución de las pruebas.	87
A.18. Datos de entrada a pruebas de usuario no autorizado.	88
A.19. Datos de entrada y coordenadas a pruebas.	89
A.20. Sin nombre de proceso de prueba.	89
A.21. Consultar proceso de prueba.	90
B.1. Estudio computacional de la ecuación de Ornstein-Zernike.	92
B.2. Opciones de administración del usuario Administrador.	93
B.3. Opción Alta.	94
B.4. Opción Alta con problema.	94
B.5. Opción Baja.	95
B.6. Opción Baja con problema.	96
B.7. Opción Modificar.	96
B.8. Opción Modificar con problema de llenado.	97
B.9. Opción Modificar con problema de usuario inexistente.	97
B.10. Opción Consultar.	98
B.11. Consultar procesos por pares.	99

Resumen

El presente estudio tiene el propósito de crear interfaces para un sistema que resuelve la ecuación de Ornstein - Zernike¹, en aras de calcular eficientemente las soluciones de esta ecuación para líquidos no-homogéneos. Se ha de proveer al usuario de un ambiente amigable de trabajo.

La importancia de esta tesis estriba en el aspecto técnico y se refiere a la eficiente administración del tiempo y de los recursos humanos. La importancia técnica estriba en el potencial explicativo que se obtiene para los usuarios del sistema que resuelve la ecuación OZ ya que permite graficar los pasos intermedios del proceso de solución, especificar en cual región del plano cartesiano se va a desarrollar la ejecución y, finalmente, establecer los parámetros físicos que intervienen en el procedimiento. Esto implica que cualquier usuario con conocimientos básicos de la Física del problema podrá verificar el desarrollo del procedimiento de la ecuación OZ; por ende, los resultados obtenidos al utilizar el sistema serán más precisos dado que provee una fácil manipulación de los datos de entrada.

También representa una importante aportación en lo que respecta a la administración del tiempo y los recursos humanos porque de aquí en adelante es posible mandar a ejecutar el sistema desde cualquier estación de trabajo, evitando al usuario la exigencia de desplazarse al servidor o de restringir el uso de su computadora únicamente a la ejecución del sistema.

Finalmente, dotamos al sistema de una interfaz gráfica que permita a un usuario modificar los valores sin que sea necesario modificar el programa ni recompilarlo. Este desarrollo está hecho en PHP, que permite mostrar el sistema en cualquier navegador *WEB*, haciendo que sea posible ejecutarlo desde diferentes sistemas operativos. Para lograr esto fue necesario analizar un sistema que resuelve la ecuación, el cual esta hecho en el lenguaje de programación FORTRAN, rediseñar las rutinas necesarias para la manipulación de los resultados y crear rutinas que permitieran una óptima manipulación de los datos de entrada en el lenguaje de programación C.

Términos más usados: Líquidos no-homogéneos, Ec. OZ, Raíces de operadores, Método de Newton-Raphson.

¹De aquí en adelante OZ

Abstract

The present study aims to establish sound interfaces to a computational system to solve the Ornstein - Zernike equation ², which is used to calculate efficiently the solutions of the OZE for non-homogenous liquids. Our work will provide to any user a friendly work environment.

The importance of this thesis is rather technical and is related to the efficient administration of both time and human resources. Also the users of the system may have at hand detailed explanations at any computing moment and will be able to display graphs of any intermediate step. It is possible to specify any region in the Cartesian plane in which the transformed equations will be treated. Finally, it is possible to introduce the physical parameters determining the whole procedure. This is with the purpose that any user with basic knowledge of the Physics of the problem will be able to check the OZ equation solving procedure.

We estimate our work as an important contribution regarding the system exploitation since it is possible to command the system execution from any workstation, avoiding the requirement to move to a server terminal or occupy exclusively a computer to the system execution.

Finally, we endowed the system with a graphical interface allowing the user to modify the input parameters without any recompilation. Some of our programs have been done in PHP, hence the system can be displayed at any web browser, independently of the operating system used.

Some redesign tasks were accomplished for the original Fortran system and several procedures were re-programmed in the programming language C.

Frequently used terms: Non-homogeneous liquids, Ec. OZ, Newton - Raphson's Method, Roots of operators.

²OZE from now on

Agradecimientos

Agradezco a el *CONACYT* por el apoyo económico brindado durante la realización de este proyecto.

Agradezco al *CINVESTAV*, esta institución de enorme calidad, que me brindó todo el apoyo durante mi estancia.

Quiero agradecerle a mi asesor de tesis, el *Dr. Guillermo Benito Morales Luna*, sus conocimientos invaluablees que me brindo para llevar a cabo esta investigación, y sobre todo su gran paciencia para esperar a que este trabajo pudiera llegar a su fin.

Agradezco a los miembros del jurado, el *Dr. José Oscar Olmedo Aguirre* y al *Dr. Luis Gerardo de la Fraga*, por las valiosas contribuciones que hicieron al trabajo final y por el tiempo que dedicaron para revisarlo, aún a pesar de tantas actividades que los ocupan.

Agradezco a aquellas grandes personas que hacen posible el conocimiento en las aulas de la sección, los exelentes profesores del programa de maestría. A mis compañeros de la generación, por todos los buenos y malos momentos que viví con ellos. A todos los que alguna vez han compartido sus conocimientos para enriquecernos todos.

Agradezco a Sofia Reza, por toda su comprensión, nobleza y dedicación que en todo momento mostró hacia mis compañeros y a mí.

Agradezco a Jorge Enrique Morfín Galván por todas sus enseñanzas, sugerencias y apoyo a los largo de nuestra valiosa amistad.

Agradezco a Luis Cruz Romo por todo su apoyo cuando más abatida me sentía, inyectandome animo para continuar lo que había empezado.

Agradezco a mi abuela por su cariño, paciencia, comprensión y apoyo desde siempre, inculcandome que todo lo que sueña es posible realizarlo.

Agradezco a mis hermanas quienes me han inundado en la alegría de compartir todos nuestros sentimientos y proyectos viendo los grandes logros y tropiezos de una forma amena.

A mis padres quienes me han heredado el tesoro más valioso que puede dasele a una

hija, amor. Quienes sin escatimar esfuerzo alguno han sacrificado gran parte de su vida, que me han formado y educado. A quienes la ilusión de su existencia ha sido verme convertida en persona de provecho. A quienes nunca podré pagar con las riquezas mas grandes de mundo . A ellos los seres universalmente más queridos sinceramente Gracias.

Dedicatoria

Dedico esta tesis a mi madre por enseñarme a ser la persona que soy y a quién le admiro su fortaleza de carácter a pesar de todos los problemas a los que se ha enfrentado en su vida, siempre saliendo alerosa.

Dedico está tesis a mi padre por permitirme llevar a cabo todos mis sueños e impusarme para lograrlos.

Capítulo 1

Introducción

La naturaleza de las teorías del estado líquido basadas en ecuaciones integrales, son un grupo de teorías que utilizan relaciones exactas ó aproximadas de la Mecánica Estadística, para obtener propiedades estructurales y termodinámicas de fluidos. En estas teorías se encuentran algunas ventajas como son buscar resolver una ecuación para $g(r)$, así el utilizar estas teorías requiere menos tiempo de cálculo que realizar una simulación, también no presentan problemas de tamaño finito y son más precisas que teorías más simples (campo medio), pero en realidad son solo teorías aproximadas, su bondad depende del sistema estudiado, tienen falta de consistencia termodinámica y una zona de no-solución.

Este es el motivo por el cual Ornstein y Zernike la propusieron la ecuación analizada en la presente tesis, en 1914, buscando la solución a la necesidad de encontrar una función de correlación que tuviera el mismo alcance que el potencial de interacción y (por esta razón) no divergiera en el punto crítico. Esta es exacta e implica la definición de la función de correlación directa, la ecuación de Ornstein-Zernike se conoce como una teoría de funciones de correlación de muchos cuerpos.

La función de correlación total se encuentra representada por $h(r) = g(r) - 1$, así, $h(r)$ es igual a la correlación directa más la correlación indirecta, lo que es la ecuación de Ornstein-Zernike expresada a continuación

$$h(r) = c(r) + \int c(|r - r'|)h(r')dr' \quad (1.1)$$

Donde $c(r)$ representa la correlación directa debida unicamente a las fuerzas de interacción ejercidas entre dos partículas y la integral representa las fuerzas ejercidas a través de las correlaciones directas con partículas intermedias. Al tener una incognita como resultado de una variable en la operación sale y entra la función de distribución la salida representa la difución de luz.

La ecuación OZ relaciona a $g(r)$ y $c(r)$, por lo que necesitamos una relación adicional entre ambas funciones a lo que se le llama cerradura de la ecuación OZ.

El objetivo de este estudio es la construcción de una alternativa de resolución a la ecuación de Ornstein-Zernike¹ para fluidos no-homogéneos. En nuestro trabajo, hemos partido del sistema desarrollado en el Programa de Ingeniería Molecular del Instituto Mexicano del Petróleo. Actualmente la ejecución de dicho sistema resulta difícil para el

¹De aquí en adelante *ecuación OZ*

usuario final en el sentido de que no le facilita el control de las variables involucradas dada la ausencia de interfaces gráficas así como por una cierta inconsistencia en la estructura del almacenamiento de datos al grado de que él puede verse en una situación de pérdida parcial o total de los datos supuestamente almacenados. Otro elemento que dificulta el uso del sistema en su forma original es su necesaria recompilación cuando se introducen nuevos datos. Por ejemplo, un usuario define en el sistema unas matrices de tamaño determinado, si otro usuario quiere utilizar el sistema con matrices de diferente tamaño, entonces tendrá que entrar al código del programa y recompilarlo.

La alternativa propuesta aquí soluciona esta situación para el usuario final proveyéndole en primer término de una interfaz gráfica, la cual además muestra las graficas generadas a lo largo de la ejecución, en segundo término de una base de datos que almacena la información total de ejecuciones anteriores evitando que el sistema se ejecute varias veces con los mismos datos y, en tercer término, de dinamismo en la definición de las matrices de manera que se evite la recompilación del sistema cada vez que se le utiliza.

Para realizar esta alternativa programamos una interfaz gráfica en PHP; construimos una base de datos en MySQL, reprogramamos algunas rutinas de el lenguaje FORTRAN en el lenguaje C y creamos rutinas para que los datos solicitados entre C y FORTRAN sean equivalentes.

Para esto ha sido necesario trabajar con un lenguaje de *scripts* de muy alto nivel[1] para servidores *Web* como lo es PHP[2]; utilizando aplicaciones visuales, a saber *navegadores WEB*, siempre bajo la suposición de que la posibilidad de visualizar la información en forma de imágenes estructuradas brinda una mejor vía de acceso al conocimiento y a la comprensión del mismo.

De esta forma, el segundo capítulo es introductorio, en tanto ofrece un panorama global sobre las distintas formas en que se ha abordado la ecuación OZ, así como una breve revisión de la misma destacando el porqué de su relevancia; finalmente, revisamos brevemente los métodos por los cuales se llega a su resolución. Por igual proveemos la definición de otros términos básicos necesarios para la comprensión total del trabajo.

El tercer capítulo contiene la revisión del tratamiento matemático y numérico que involucra la solución de la ecuación OZ. Para esto es importante revisar en primer término el planteamiento del problema en forma matemática, presentando las características específicas de las coordenadas *esféricas prolatas* y el paso del plano referido mediante coordenadas cartesianas al plano en coordenadas *esféricas prolatas*. Bien que la ecuación OZ modela un fenómeno en tres dimensiones, se tiene que éste posee una simetría cilíndrica, por lo cual basta describirlo en dos dimensiones puesto que una tercera, a saber, el ángulo respecto a un plano de referencia, puede ser omitida debido a la simetría. Revisamos aquí cómo se realiza la integración numérica. Presentamos el método de Newton-Rapshon² y el cálculo de soluciones de la ecuación OZ. Incluimos breves presentaciones del proceso de especificación de mallas planares, del proceso de intercambio entre representaciones (analizando un pequeño ejemplo de este intercambio) y la forma en que realizamos los cálculos de las funciones núcleo. Finalmente, explicamos en cuáles partes del sistema se utiliza el MNR y analizamos algunos despliegues gráficos.

En el cuarto capítulo, especificamos la alternativa de diseño del sistema de cómputo, objetivo prioritario de la presente tesis. Incluimos como parte del contenido la interfaz

²De aquí en adelante MNR

gráfica (vista del usuario), la base de datos implementada y la transcripción al lenguaje de programación C del sistema que resuelve la ecuación OZ en FORTRAN.

En el quinto capítulo explicamos la ingeniería del software, analizando la manera en la que programamos los métodos numéricos, las entradas que se reciben y las salidas generadas por el sistema.

En el sexto capítulo, hacemos hincapié en las conclusiones obtenidas y el trabajo que podría realizarse a futuro para optimizar el sistema.

El apéndice A incluye un manual de uso elaborado para el administrador del sistema.

El apéndice B ofrece un tercer manual, dirigido a usuarios finales, para que éstos puedan controlar la manipulación de la información sin tener como condición previa una formación teórica de índole matemática, física o computacional sobre el problema. Equivaldría al manual con el que cuenta cualquier usuario para cualquier aplicación, al margen de su conocimiento sobre el cómo tales aplicaciones fueron programadas.

Al final de la tesis se podrá encontrar un disco compacto que contiene la tesis en su totalidad, con un agregado fundamental, a saber, cuatro apéndices: 1) una versión ejecutable del programa original hecho en FORTRAN que resuelve el sistema; 2) los códigos de los programas rediseñados en lenguaje C; 3) los códigos de los programas hechos en PHP; 4) la estructura de la base de datos; y 5) el documento de la presente tesis. La presentación de esta información obedece a las siguientes razones fundamentales: si bien es cierto que la información contenida solamente en el disco compacto es una parte fundamental del trabajo de tesis, y aún más, es uno de sus principales productos, no se ha impreso incluyéndole ya sea como apéndice o como parte del contenido de algún capítulo pues, en principio, su contenido es amplio. Además se trata de un material que no es relevante para la lectura y la comprensión de la presente tesis y, por último, porque es esencial para el análisis y la ejecución del sistema de facto, condición esta última que por sí misma obliga su inclusión.

Capítulo 2

La ecuación de Ornstein-Zernike

Un elemento fundamental en el análisis de fluidos es la noción de *densidad* considerada como la razón de partículas entre unidades de volumen. El tratamiento matemático de una función de densidad está basado en la idea de energía libre de fluidos no-homogéneos, caracterizados éstos en términos de la densidad. El formalismo que utilizamos fue desarrollado principalmente para fluidos no-homogéneos durante la década de los años 60 del siglo XX y ha sido consecutivamente aplicado a fluidos homogéneos. Aquí la función de correlación total $h : r \mapsto h(r)$ y la función de correlación directa $c : r \mapsto c(r)$, puestas ambas en términos de un vector de posición r , están relacionadas por medio de la ecuación integral de Ornstein-Zernike. Para resolver la ecuación OZ debe darse una relación entre c y h a la que se le llama *cerradura* de la ecuación OZ. Dependiendo de la cerradura, diferentes teorías de ecuaciones integrales pueden ser desarrolladas. Las más conocidas de éstas son *Percus-Yevick* (PY), *cadena entrelazada* (*hypernetted chain*) (HNC) y la *aproximación esférica media* (*mean spherical*) (MSA).

2.1. Desarrollos alternativos actuales

La literatura reciente indica que existen diversas implementaciones de métodos para resolver la ecuación OZ, probadas eficientemente de manera experimental. Tales trabajos se han elaborado en diversos centros del mundo, entre otros, el Departamento de Física de la Universidad de Salamanca, España [3]; el Departamento de Química de la Universidad de la Columbia Británica, Canadá [4], en el Departamento de Física del Cinvestav [5] y en el Programa de Ingeniería Molecular del Instituto Mexicano del Petróleo [6], en México. Es posible observar que cada implementación elige su propia cerradura y consecuentemente su propio método de resolución.

Actualmente diversas instituciones utilizan el método de Newton-Raphson para localizar soluciones a la ecuación OZ, tratando de acelerar la convergencia a partir de ciertos puntos dados. Reseñaremos en lo que sigue algunos enfoques de entre los que consideramos más importantes:

En [7] los autores presentan el *Método SCOZA: Self-consistent Ornstein-Zernike approach for monolayer films*. Ahí se refieren a un método de aproximación para obtener factores de estructura de modelos hamiltonianos en tres o más dimensiones, así como sus características termodinámicas, y a otros métodos que proporcionan resultados muy ex-

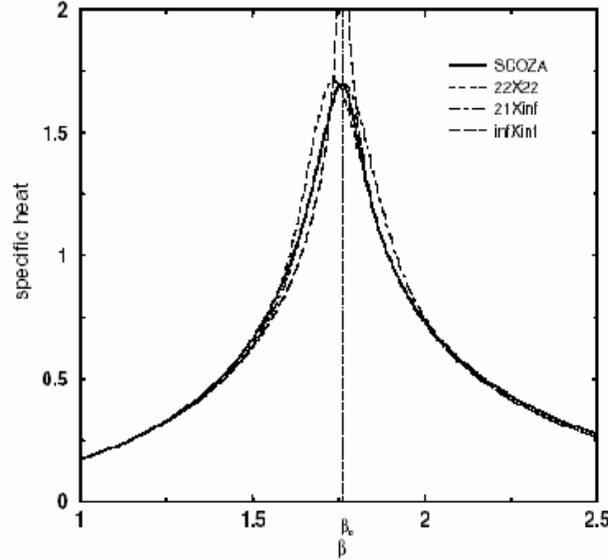


Figura 2.1: Volumen constante del calor específico C_v para SCOZA comparado con resultados exactos. Se nota que el calor específico para SCOZA no diverge hasta el máximo, muestran un semblanza de modelos de tamaño finito

actos en retículas bidimensionales, lo que ha hecho propicio el tratamiento de problemas de películas delgadas. El potencial entre partículas en el modelo SCOZA es de la forma

$$v(r_i - r_j) = \begin{cases} +\infty & \text{si } r_i = r_j \\ -\omega & \text{si } i, j \text{ son vecinos cercanos} \\ 0 & \text{en otro caso} \end{cases}$$

La energía interna por *spin* está relacionada con la energía interna por partícula mediante:

$$U = \rho u + \frac{1}{2}q\rho - \frac{1}{8}q \quad (2.1)$$

En vez de tener una pendiente infinita en la temperatura crítica exacta, la pendiente de SCOZA alcanza su máximo en una temperatura dentro de una fracción de un 1% del valor exacto (véase las figuras 2.1, 2.2). SCOZA es un sistema muy completo que trata casos especiales de la ecuación de Ornstein-Zernike.

El tratamiento que realizamos en el presente trabajo es más general y no involucra suposiciones *ad-hoc*. Sin embargo, esto lo hace más complejo y menos particularizado que el caso bidimensional presentado en [7].

Ahora bien, en [8] se presenta el *modelo de Yukawa para coloides de carga bidimensional estabilizada* (*Hard-core Yukawa model for two-dimensional charge stabilized colloids*) que utiliza las cerraduras HNC y PY para aproximar el diagrama de fase. Se calcula el factor estático de la estructura y la función de distribución variando los parámetros representativos. La función de correlación estática se utiliza para estimar el diagrama de fase líquido-sólido, también para una gama amplia de parámetros. Ahí se ilustra el uso de las condiciones HNC y PY que son utilizadas también en nuestro tratamiento (véase la

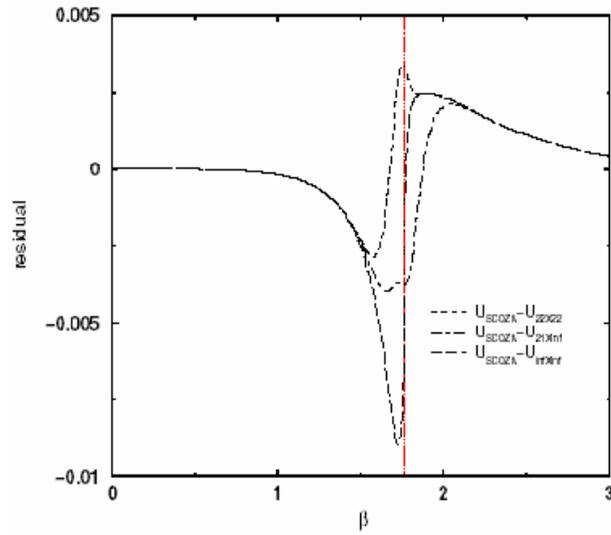


Figura 2.2: Residuales entre el SCOZA y resultados exactos. Para resultados óptimos de tamaño finito, las derivaciones son mas pequeñas que $\infty \times \infty$ de resultados exactos

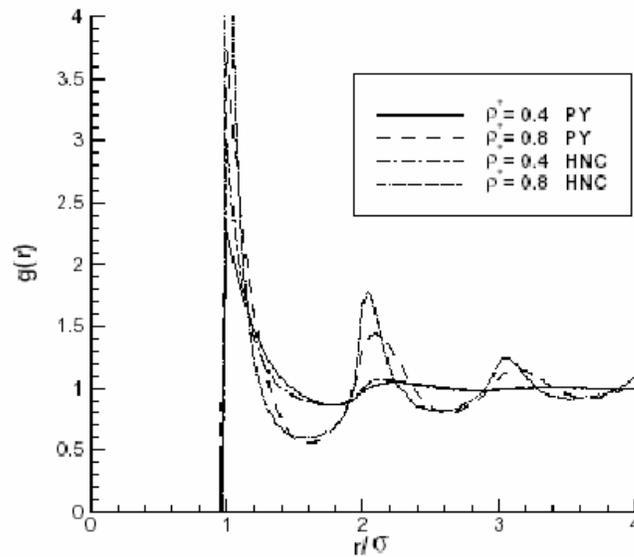


Figura 2.3: Modelo de esfera dura de Yakuwa

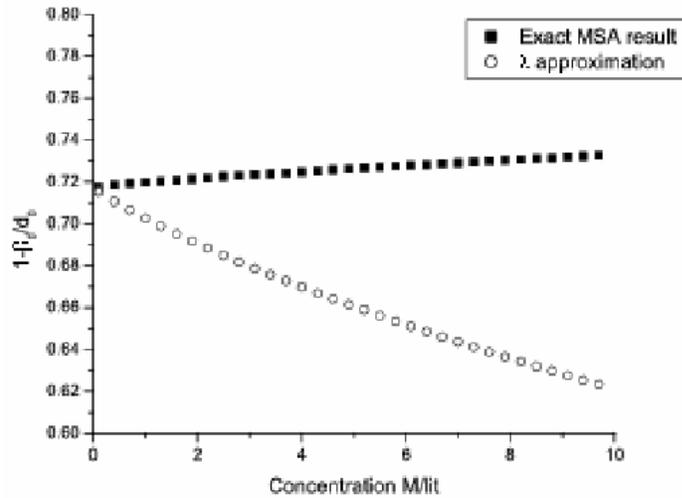


Figura 2.4: La polarización *MSA* del parámetro *lambda* para un ion bipolar mezclado como una función de concentración molar

figura 2.3). Se presenta así mismo un problema de coloides esencial en la teoría de líquidos no-homogéneos, consistente en que la interacción entre diversos pares de partículas al nivel microscópico que llega a determinar el comportamiento del líquido en un nivel macro. El modelo presentado muestra la importancia de la modelación numérica en este tipo de problemas relativos a líquidos.

Finalmente, en [9] se utiliza la MSA para escalar mezclas de iones duros con dipolos. Ahí, los autores obtienen expresiones simples de las propiedades termodinámicas por medio de la MSA. Utilizándola se muestra qué funciones termodinámicas se pueden formular en términos de un conjunto reducido de matrices de cambio de escala en los ejes de coordenadas. Se presenta un problema de iones duros y bipolares y la forma en la que deben ser medidas sus características, para obtener un modelo ideal del comportamiento de las partículas a un nivel molecular. Se utilizan modelos equivalentes para los cambios en las cargas eléctricas de las partículas

En suma, el modelo presentado muestra la importancia de la modelación numérica en diversos tipos de problemas en sólidos y líquidos.

2.2. Descripción de la ecuación

La ecuación OZ es una ecuación integral que se resuelve con técnicas de elemento finito, mediante una técnica iterativa utilizando el MNR, para localizar raíces de operadores en espacios de grandes dimensiones. La ecuación describe funciones de correlación entre pares de partículas, y su planteamiento involucra tan solo dos incógnitas, por lo que es necesario introducir suposiciones suplementarias para resolverla, las cuales, como ya hemos visto, pueden ser las condiciones HNC y MSA las cuales se consideran componentes de la ecuación OZ. Para poder combinarlos se requiere un tercer tratamiento llamado Extensión

a Tres Puntos en el cual se considera una tercera partícula, esto último obedece a que las ecuaciones HNC y MSA no tienen variables en común.

Siempre que la ecuación OZ reúna las características antes mencionadas se obtendrá una ecuación integral cuya función incógnita será una transformación definida del plano real al conjunto de números reales. Para determinar una solución es necesario en primer término resolver un problema de localización de un punto fijo, planteado en un espacio de funciones, si se considera la ecuación integral, o un espacio de sucesiones, si se transforma la ecuación original en una ecuación discretizada mediante técnicas de elemento finito.

La ecuación integral puede verse como un punto fijo de un operador Φ , digamos $h = \Phi(h)$, que actúa sobre el espacio de funciones continuas e integrables del plano en los reales. Cuando se discretiza por medio de elemento finito, se le formula como una ecuación $h = \Phi_D(h)$ donde ahora Φ_D es un operador en un espacio de sucesiones reales definidas sobre una malla en el plano. En ambos casos la función punto fijo se denomina: función solución. En el caso del espacio de sucesiones, el MNR aparece como una alternativa efectiva [10].

Al realizar el procedimiento MNR, distinguimos varias etapas relevantes:

- Establecimiento del funcional (cálculo de sus parámetros)
- Evaluación del funcional
- Iteración de MNR
- Estimación de la aproximación
- Despliegue de resultados

Actualmente, si se considera una malla de orden $m \times n$, donde $m \cdot n$ es el número de puntos en el plano para aproximar a la función solución y k es el número de partículas consideradas, entonces la iteración de MNR tendrá una complejidad del orden $O((knm)^2)$, tanto en tiempo como en espacio.

Hasta hoy este procedimiento ha exigido amplios tiempos de ejecución, por ello diferentes disciplinas, entre ellas la Física, la Matemática y la Química han buscado formas de optimizarlos.

2.3. Diseño computacional

La propuesta actual parte de que el sistema que efectivamente resuelve la ecuación OZ adolece de una debilidad en sus interfaces con los usuarios en tanto que cada usuario final carece de un control visual sobre las entradas a suministrarle.

El sistema en su versión original, requiere la especificación de, entre otros parámetros, de la malla sobre la que efectuará cálculos de elemento finito. Esta se especifica mediante un conjunto de *tercetas* numéricas cuyos valores determinan la malla de acuerdo con un procedimiento muy poco convencional. El uso de esta representación dificulta la comprensión y es, acaso, muy *sui-generis* de la Física del problema. Así, todo usuario que desconozca ciertas precisiones y complejidades de esa Física se enfrenta a la necesidad

de utilizar variables que no puede controlar lo que propicia condiciones de *overflow* y *underflow*, lo cual, a su vez imposibilitaría la resolución del problema planteado.

El diseño aquí propuesto responde a esta dificultad enriqueciendo el sistema con una interface, más amigable. Se ha modificado la entrada originalmente propuesta para especificar ahora, en cada eje de coordenadas, los valores extremos, el número de puntos en ellos, e inclusive intervalos de distintas densidades, es decir, de cada razón de los números de puntos entre las longitudes de esos intervalos. La introducción de estas últimas dos variables permite al usuario ejercer un mayor control sobre la extensión y la densidad de la malla sin preocuparse de la Física del problema, y da una mayor intuición respecto a los parámetros iniciales, con lo cual se evitan condiciones de *overflow* o *underflow*. Otro aspecto nuevo con el que se dotó al sistema es la posibilidad de mostrar como gráficas en GNUPLOT[11] el comportamiento de la solución y de los estados previos en diversas iteraciones del MNR. Por otro lado, el sistema como fue diseñado originalmente arroja los resultados en forma tabular, lo cual es poco amigable para la mayoría de los usuarios. El presente diseño, además, favorece la presentación gráfica de resultados en un navegador *WEB*. Con todas estas características en la interfaz, el sistema se hace más comprensible y controlable para el usuario.

A continuación mostramos dos figuras en las que mostramos diferencias entre la propuesta actual y la original, en cuanto a los datos que solicita el programa. Como se puede apreciar la propuesta actual solicita más valores que la original esto con el propósito de generar una malla de diferentes densidades, y no utilizar como entrada las tercetas, que poco aportan al entendimiento natural de lo que se esta llevando a cabo en esta parte del programa. En la propuesta actual se puede apreciar que muchos de los datos se solicitan son similares a los de la propuesta anterior, esto debido a que son necesarios para el procesamiento correcto del sistema, en su parte medular.

PROPUESTA ACTUAL	PROPUESTA ORIGINAL
1 Especificación de la temperatura con la que se trabaja;	1 Especificación de la temperatura con la que se trabaja;
2 Especificación de la contante dieléctrica del líquido a evaluar;	2 Especificación de la constante dieléctrica del liquido a evaluar;
3 Especificación del diámetro de las esferas, una en el eje positivo y otra en el negativo;	3 Especificación del diámetro de las esferas, una en el eje positivo y otra en el negativo;
4 Especificación de la distancia entre los centros de las esferas;	4 Especificación de la distancia entre los centros de las esferas;
5 Especificación de la distancia del centro de la esfera al eje que las separa;	5 Especificación de la distancia del centro de la esfera al eje que las separa;
6 Especificación de las cargas que se utilizan;	6 Especificación de las cargas que se utilizan;

PROPUESTA ACTUAL	PROPUESTA ORIGINAL
<p>7 Especificación del número de ocasiones necesarias para que el sistema se ejecute en aras de lograr una mayor aproximación del resultado esperado de acuerdo a experimentos realizados con anterioridad;</p>	<p>7 Especificación del número de ocasiones necesarias para que el sistema se ejecute en aras de lograr una mayor aproximación del resultado esperado de acuerdo a experimentos realizados con anterioridad;</p>
<p>8 Especificación de tres valores que indican que densidad tendrá el plano de trabajo;</p>	<p>8 Conjunto de <i>tercetas</i> que llevan implícito el tamaño de la malla y su densidad.</p>
<p>9 Especificación del tamaño del plano en el que se va a trabajar, tomando tres valores, el de la variable denominada <i>eta</i>, que puede tener la extensión que se requiera y, el valor de las variables <i>ksi1</i> y <i>ksi2</i>, que se encuentran en el eje de las abscisas, bajo la condición previamente establecida respecto a que la suma de estas últimas dos variables no debe ser mayor que 1;</p>	
<p>10 Especificación de un conjunto de coordenadas límite (relacionados con la determinación de las variables <i>eta</i>, $ksi1+ksi2 < 1$), así como de los índices en los que se encuentra cada coordenada.</p>	

Como veremos más adelante, la ecuación OZ al ser discretizada tiene la forma

$$\omega_{j,\iota_1} = \exp \left[M_j(\eta_{\iota_1}, \xi_{\iota_1}) + \sum_{m=1}^k \rho_m \sum_{\iota_2 \in I_0} \omega_{m,\iota_2} C_{\iota_2}(\eta_{\iota_1}, \xi_{\iota_1}) \right] - 1.$$

A la izquierda de la ecuación se encuentra la incógnita y a la derecha el operador, en términos de la incógnita, cuyo punto fijo da la solución del problema. Al localizarla, la sucesión se grafica como una función en el plano en los reales. El MNR parte de una incognita ω_0 inicial y va construyendo una nueva en cada iteración hasta converger en la solución buscada. Pues bien, nuestro sistema facilita el despliegue gráfico de esas sucesiones previas, incluida la final. Ahora, por razones técnicas, se introduce un cambio de coordenadas de manera que el dominio original de la solución, de contorno circular, se transforma en uno de malla que es rectangular en una región de ella y con los ejes verticales inclinados a la derecha en una segunda región. El sistema permite graficar las sucesiones de la variable tanto en el dominio original como en el dominio transformado mediante el cambio de coordenadas.

Al resolver la ecuación OZ se han de realizar procesos de aproximación numérica, por ejemplo en el cálculo de las funciones núcleos, y de resolución de sistemas lineales de ecuaciones. En cada iteración es necesario almacenar una gran cantidad de valores calculados. Todos los datos que son generados al ejecutar el sistema, así como aquellos que fueron introducidos por el usuario son almacenados en una base de datos. Este almacenamiento tiene el propósito de que si posteriormente algún usuario deseará consultar la información introducida y generada por el programa, éste disponible la información.

Capítulo 3

Tratamiento matemático y numérico

3.1. Planteamiento del problema

La ecuación OZ que se evalúa es de la forma

$$g_{\alpha j}(\mathbf{r}_{\alpha j}) = \exp \left[c_{\alpha j}(\mathbf{r}_{\alpha j}) + \int \left[\sum_{m=1}^k \rho_m h_{\alpha m}(\mathbf{r}_3 - \mathbf{r}_{\alpha}) c_{mj}(\mathbf{r}_3 - \mathbf{r}_j) \right] d\mathbf{r}_3 \right] \quad (3.1)$$

donde

- h es una función, llamada de *correlación directa*, y es, de hecho, la función incógnita en la ecuación OZ (3.1), $g = h + 1$, y
- las funciones c son también funciones de *correlación indirecta*; son parámetros en la ecuación OZ.

En la *teoría de los tres cuerpos* se tiene que el índice α se refiere a un complejo formado por dos partículas, que llamamos β y γ , separadas una distancia τ (véase la figura 3.1).

Planteamos la ecuación (3.1) utilizando las así llamadas *coordenadas esféricas prolatas* (las que trataremos con más detalle en la sección siguiente).

Sea $B_1 = [1, +\infty[\times [-1, 1] \times] - \pi, \pi]$ la región en el espacio \mathbb{R}^3 que es el producto cartesiano de tres intervalos: en el primero, $[1, +\infty[$, toma valores una variable, η , que hace el papel de radio, en el segundo $[-1, 1]$ una variable, ξ , y en el tercero, $] - \pi, \pi]$, una variable θ , que hace el papel de ángulo, digamos de *longitud*, es decir, de giro respecto al eje z . Sea $\Phi : B_1 \rightarrow \mathbb{R}^3$ la función de cambio de coordenadas:

$$\begin{aligned} \Phi & : (\eta, \xi, \theta) \mapsto \frac{\tau}{2} [M(\eta, \xi) \cos(\theta), \quad M(\eta, \xi) \sin(\theta), \quad \eta \cdot \xi] \quad \text{donde} \\ M & : (\eta, \xi) \mapsto \sqrt{(\eta^2 - 1)(1 - \xi^2)} \end{aligned}$$

En valor absoluto, el jacobiano de esa transformación es $|J\Phi(\eta, \xi, \theta)| = \frac{\tau^3}{8}(\eta^2 - \xi^2)$. La integral $Int(\mathbf{r}_{\alpha}, \mathbf{r}_j)$ que aparece a la derecha de la ecuación (3.1) puede expresarse equivalentemente como:

$$Int(\mathbf{r}_{\alpha}, \mathbf{r}_j) = \frac{\tau^3}{8} \int_{B_1} \left[\sum_{m=1}^k \mu_m \right] (\eta_3^2 - \xi_3^2) d\eta_3 d\xi_3 d\theta_3 \quad (3.2)$$

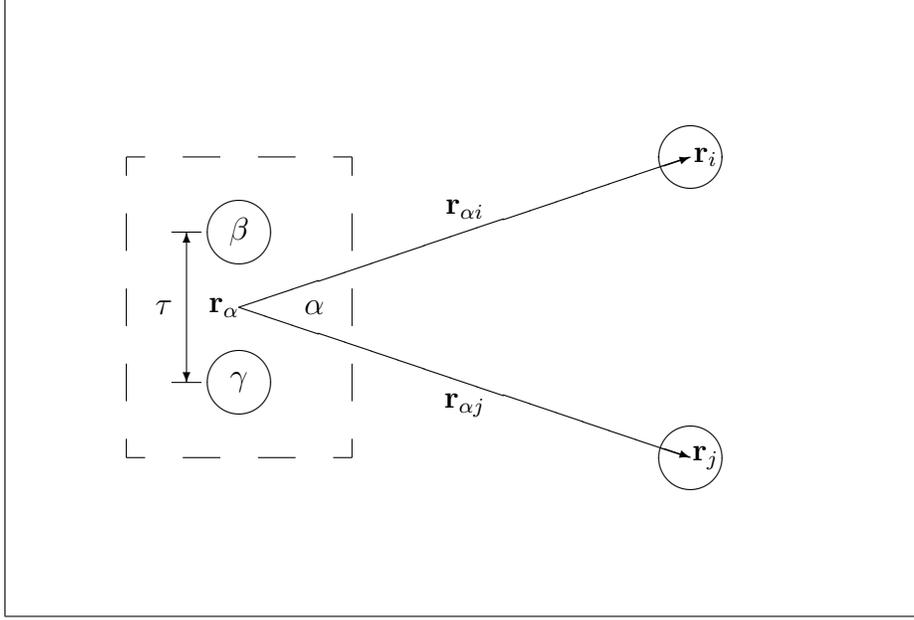


Figura 3.1: Complejo α formado por dos partículas β y γ . Donde α esta localizada en el centro de masa de β y γ entonces $m_{\alpha} = m_{\beta} + m_{\gamma}$

donde $\mu_m = \rho_m h_{\alpha m}(\Phi(\eta_3, \xi_3, \theta_3) - \mathbf{r}_{\alpha}) c_{mj}(\Phi(\eta_3, \xi_3, \theta_3) - \mathbf{r}_j)$

Bajo condiciones de simetría esférica, suponemos que cada función $(\eta_3, \xi_3, \theta_3) \mapsto h_{\alpha m}(\Phi(\eta_3, \xi_3, \theta_3) - \mathbf{r}_{\alpha})$ es constante respecto a su tercer argumento, θ_3 . Introducimos el sistema de coordenadas cartesiano con origen en el vector de posición \mathbf{r}_{α} y con eje z a lo largo de la dirección $\overrightarrow{\gamma\beta}$. Escribamos $h_{1;\alpha m}(\eta_3, \xi_3) = h_{\alpha m}(\Phi(\eta_3, \xi_3, \theta_3) - \mathbf{r}_{\alpha})$.

También supondremos que las funciones $[\eta_3, \xi_3, \theta_3] \mapsto c_{mj}(\Phi(\eta_3, \xi_3, \theta_3) - \mathbf{r}_j)$ coinciden para las diferentes m 's y que su valor común se descompone como la suma de tres funciones dependientes únicamente de la magnitud del argumento:

$$c_{mj}(\mathbf{r}) = c_{j;1}(\|\mathbf{r}\|) + c_{j;2}(\|\mathbf{r}\|) + c_{j;3}(\|\mathbf{r}\|) \quad (3.3)$$

Para dos puntos $\mathbf{r}_1, \mathbf{r}_3 \in \mathbb{R}^3$ representados en *coordenadas esféricas prolatas* como

$$\mathbf{r}_1 = \Phi(\eta_1, \xi_1, \theta_1) \quad , \quad \mathbf{r}_3 = \Phi(\eta_3, \xi_3, \theta_3) \quad (3.4)$$

la distancia euclidiana entre ellos, es decir, la magnitud de su diferencia (\mathbf{r}_{13}), se expresa como:

$$\|\mathbf{r}_{13}\| = \frac{\tau}{2} \sqrt{d_{1;13} + d_{2;13}} \quad (3.5)$$

donde

$$\begin{aligned} d_{1;13} &= -2 + \xi_1^2 + \xi_3^2 + \eta_1^2 - 2\xi_1\xi_3\eta_1\eta_3 + \eta_3^2 \\ &= \nu + (\xi_1\eta_1 - \xi_3\eta_3)^2 \\ d_{2;13} &= -2\sqrt{\nu} [\cos(\theta_1)\cos(\theta_3) - \text{sen}(\theta_1)\text{sen}(\theta_3)] \\ &= -2\sqrt{\nu} \cos(\theta_1 + \theta_3) \end{aligned}$$

siendo $\nu = (1 - \xi_1^2)(-1 + \eta_1^2) + (1 - \xi_3^2)(-1 + \eta_3^2)$.

Sustituyendo la ecuación (3.3) en la ecuación (3.2) obtenemos

$$\begin{aligned} \text{Int}(\mathbf{r}_\alpha, \mathbf{r}_j) &= \int_D d\eta_3 d\xi_3 \int_{-\pi}^{\pi} \frac{\tau^3}{8} \left(\sum_{\ell=1}^3 H_k(\eta_3, \xi_3) c_{j;\ell}(\|\mathbf{r}_{3j}\|) \right) (\eta_3^2 - \xi_3^2) d\theta_3 \\ &= \sum_{\ell=1}^3 \int_D H_k(\eta_3, \xi_3) K_\ell(\eta_j, \xi_j; \eta_3, \xi_3) d\eta_3 d\xi_3 \end{aligned} \quad (3.6)$$

donde

$$\begin{aligned} D &= [1, +\infty[\times [-1, 1] \\ H_k(\eta_3, \xi_3) &= \sum_{m=1}^k \rho_m h_{1;\alpha m}(\eta_3, \xi_3) \\ K_\ell(\eta_j, \xi_j; \eta_3, \xi_3) &= \frac{\tau^3}{8} (\eta_3^2 - \xi_3^2) \int_{-\pi}^{\pi} c_{j;\ell}(\|\mathbf{r}_{3j}\|) d\theta_3 \end{aligned} \quad (3.7)$$

Habiendo hecho estas suposiciones, de la definición de h al inicio de esta sección, y de la ecuación (3.1) resulta la ecuación

$$h_{\alpha j}(\eta_j, \xi_j) = \exp \left[\lambda + \sum_{\ell=1}^3 \int_D H_k(\eta_3, \xi_3) K_\ell(\eta_j, \xi_j; \eta_3, \xi_3) d\eta_3 d\xi_3 \right] - 1 \quad (3.8)$$

donde

$$\lambda = c_{\alpha j} [\Phi(\eta_j, \xi_j)] \quad (3.9)$$

la cual ha de resolverse para h (aquí, la pareja (η_j, ξ_j) denota a la posición genérica de la partícula j , con $j \leq k$). Puesto que existe un $x_0 > 1$ tal que

$$(\forall (\eta_3, \xi_3) \in D : \eta_3 > x_0 \Rightarrow \forall m \leq k : h_{\alpha m}(\eta_3, \xi_3) = 0),$$

basta considerar el dominio $D_0 = \{(\eta, \xi) \in D \mid \eta \leq x_0\}$.

Sea $\mathcal{T} = (T_\iota)_{\iota \in I_1}$ una triangulación de D_0 y sea $((\eta_\iota, \xi_\iota))_{\iota \in I_0}$ la colección de sus nodos. Por ejemplo, si I_0 es el retículo formado por $n_T \cdot n_1$ puntos, obtenidos como una imagen isomorfa del producto cartesiano de una partición $(\eta_{0\iota})_{\iota \leq n_T}$ de $[1, x_0]$ y de una partición $(\xi_{0\iota})_{\iota \leq n_1}$ de $[-1, 1]$, entonces cada cuadrángulo es isomorfo a un producto $[\eta_{0\iota}, \eta_{0,\iota+1}] \times [\xi_{0i}, \xi_{0,i+1}]$ y da origen a dos triángulos, por lo que I_1 ha de tener $2n_T \cdot n_1$ elementos. Sea $\Psi = (\psi_\iota)_{\iota \in I_0}$ una familia de funciones básicas, tales que:

1. $\forall \iota_1, \iota_2 \in I_0 : \psi_{\iota_1}(\eta_{\iota_2}, \xi_{\iota_2}) = \delta_{\iota_1 \iota_2}$.
2. $\forall \iota \in I_0 \exists I_\iota : (\eta, \xi) \notin \bigcup_{\iota_1 \in I_\iota} T_{\iota_1} \Rightarrow \psi_j(\eta, \xi) = 0$

En otras palabras, cada función básica se anula fuera de un número muy limitado de triángulos: fuera de aquellos cuyos índices estén en I_ι . Usualmente, cada I_ι tendrá del orden de 10^2 elementos.

Para cada $m \leq k$, escribimos $h_{\alpha m}(\eta, \xi) = \sum_{\iota_1 \in I_0} \omega_{m\iota_1} \psi_{\iota_1}(\eta, \xi)$. Al evaluar esta expresión en puntos de la forma $(\eta_{\iota_2}, \xi_{\iota_2})$, se tiene

$$\forall m \leq k, \iota_2 \in I_0 : \omega_{m\iota_2} = h_{\alpha m}(\eta_{\iota_2}, \xi_{\iota_2}). \quad (3.10)$$

Así tenemos, por un lado

$$\forall m \leq k : h_{\alpha m} = \sum_{\iota_2 \in I_0} h_{\alpha m}(\eta_{\iota_2}, \xi_{\iota_2}) \psi_{\iota_2} \quad (3.11)$$

y por otro, al sustituir (3.11) en (3.8) obtenemos $\forall j \leq k, (\eta, \xi) \in D_0$:

$$\begin{aligned} h_{\alpha j}(\eta, \xi) &= \exp \left[M_j(\eta, \xi) + \sum_{m=1}^k \sum_{\ell=1}^3 \int_D \rho_m h_{\alpha m}(\eta_3, \xi_3) v \right] - 1 \\ &= \exp \left[M_j(\eta, \xi) + \sum_{m=1}^k \sum_{\ell=1}^3 \int_D \rho_m \sum_{\iota_2 \in I_0} h_{\alpha m}(\eta_{\iota_2}, \xi_{\iota_2}) \psi_{\iota_2}(\eta_3, \xi_3) v \right] - 1 \\ &= \exp \left[M_j(\eta, \xi) + \sum_{m=1}^k \rho_m \sum_{\iota_2 \in I_0} h_{\alpha m}(\eta_{\iota_2}, \xi_{\iota_2}) C_{\iota_2}(\eta, \xi) \right] - 1 \end{aligned} \quad (3.12)$$

donde $\forall j \leq k, (\eta, \xi) \in D_0$:

$$M_j(\eta, \xi) = c_{\alpha j}(\Phi(\eta_j, \xi_j)) \quad (3.13)$$

y $\forall \iota_2 \in I_0, (\eta, \xi) \in D_0$:

$$C_{\iota_2}(\eta, \xi) = \sum_{\ell=1}^3 \int_D \psi_{\iota_2}(\eta_3, \xi_3) K_{\ell}(\eta, \xi; \eta_3, \xi_3) d\eta_3 d\xi_3 \quad (3.14)$$

Sustituyendo la ecuación (3.10) en (3.12) obtenemos el sistema de ecuaciones:

$$\omega_{j,\iota_1} = \exp \left[M_j(\eta_{\iota_1}, \xi_{\iota_1}) + \sum_{m=1}^k \rho_m \sum_{\iota_2 \in I_0} \omega_{m,\iota_2} C_{\iota_2}(\eta_{\iota_1}, \xi_{\iota_1}) \right] - 1 \quad (3.15)$$

con $j \leq k, \iota_1 \in I_0$. Así pues se tiene un sistema de $k \cdot n_t \cdot n$ ecuaciones con $k \cdot n_t \cdot n$ incógnitas.

3.2. Coordenadas esféricas prolatas

3.2.1. Cambio de variable

Consideremos primeramente la función $M : \mathbb{R}^2 \mapsto \mathbb{R}$

$$(\eta, \xi) \mapsto \sqrt{(-1 + \eta^2)(1 - \xi^2)} \quad (3.16)$$

la cual está definida en $B = B_- \cup B_+$ donde $B_- =]-\infty, -1] \times [-1, 1]$ y $B_+ = [1, +\infty[\times [-1, 1]$. Evidentemente M es par: $\forall (\eta, \xi) \in B : M(\eta, \xi) = M(-\eta, \xi)$.

En la figura 3.2 se muestra la gráfica de M en un subconjunto del dominio. La gráfica (a) muestra su forma en los puntos $(\eta, \xi) \in [-100, -1] \times [0, 1]$. En la gráfica (b) se muestra en los puntos $(\eta, \xi) \in [1, 100] \times [0, 1]$.

En el conjunto $B \times \mathbb{R} \subset \mathbb{R}^3$ definamos

$$\Phi : (\eta, \xi, \theta) \mapsto [M(\eta, \xi) \cos(\theta), \quad M(\eta, \xi) \operatorname{sen}(\theta), \quad \eta \cdot \xi] \quad (3.17)$$

En la figura 3.3 se muestra la gráfica de la *primera* componente de Φ dejando constante al parámetro $\xi = \frac{1}{2}$. En la gráfica (a) se le muestra en los puntos $(\eta, \theta) \in [-100, -1] \times [-\frac{1}{2}\pi, \frac{3}{2}\pi]$. En la gráfica (b) se le muestra en los puntos $(\eta, \theta) \in [1, 100] \times [-\frac{1}{2}\pi, -\frac{3}{2}\pi]$.

En la figura 3.4 se muestra la gráfica de la *segunda* componente de Φ dejando constante al parámetro $\xi = \frac{1}{2}$. En la gráfica (a) se le muestra en los puntos $(\eta, \theta) \in [-100, -1] \times [-\frac{1}{2}\pi, \frac{3}{2}\pi]$. En la gráfica (b) se le muestra en los puntos $(\eta, \xi) \in [1, 100] \times [-\frac{1}{2}\pi, -\frac{3}{2}\pi]$.

Finalmente en la figura 3.5 se muestra la gráfica de la *tercera* componente de Φ , de manera independiente respecto al parámetro θ . En la gráfica (a) se muestra en los puntos $(\eta, \xi) \in [-100, -1] \times [-1, 1]$. En la gráfica (b) se le muestra en los puntos $(\eta, \xi) \in [1, 100] \times [-1, 1]$.

Es claro que

- $\forall \eta \in]-\infty, -1] \cup [1, +\infty[, \xi \in [-1, 1] : \Phi_{(\eta, \xi, \cdot)} : \mathbb{R} \rightarrow \mathbb{R}, \theta \mapsto \Phi(\eta, \xi, \theta)$ tiene a 2π como período.
- $\forall \eta \in]-\infty, -1] \cup [1, +\infty[, \theta \in \mathbb{R} : \Phi_{(\eta, \cdot, \theta)} : [-1, 1] \rightarrow \mathbb{R}, \xi \mapsto \Phi(\eta, \xi, \theta)$ es par.
- $\forall \xi \in [-1, 1], \theta \in \mathbb{R} : \Phi_{(\cdot, \xi, \theta)} :]-\infty, -1] \cup [1, +\infty[\rightarrow \mathbb{R}, \eta \mapsto \Phi(\eta, \xi, \theta)$ es par.

Por estas propiedades, se puede considerar a θ variando solamente en un intervalo de longitud 2π , digamos $C =]-\frac{1}{2}\pi, \frac{3}{2}\pi]$, y a ξ en el intervalo $I = [0, 1]$. A η se le deja variar en todo el conjunto $D =]-\infty, -1] \cup [1, +\infty[$. Consideremos sólo $B_0 = D \times I \times C$. La restricción $\Phi_0 = \Phi|_{B_0}$ es una biyección $B_0 \rightarrow \mathbb{R}^3$.

La inversa de Φ_0 es $\Phi_0^{-1} : \mathbb{R}^3 \rightarrow B_0, (x, y, z) \mapsto (\eta, \xi, \theta)$ donde

$$\xi = \sqrt{\frac{1}{2}(\kappa - \sqrt{\lambda})} \quad (3.18)$$

$$\text{con } \kappa = x^2 + y^2 + z^2 + 1, \quad \lambda = (\kappa - 2z)(\kappa + 2z)$$

$$\eta = \frac{1}{\xi} \quad (3.19)$$

$$\theta = \begin{cases} \arctan\left(\frac{y}{x}\right) & \text{si } x > 0, \\ \frac{\pi}{2} & \text{si } x = 0 \quad \& \quad y \geq 0, \\ -\frac{\pi}{2} & \text{si } x = 0 \quad \& \quad y < 0, \\ \arctan\left(\frac{y}{x}\right) + \pi & \text{si } x < 0, \end{cases} \quad (3.20)$$

(los radicandos involucrados en la definición de ξ siempre son positivos.)

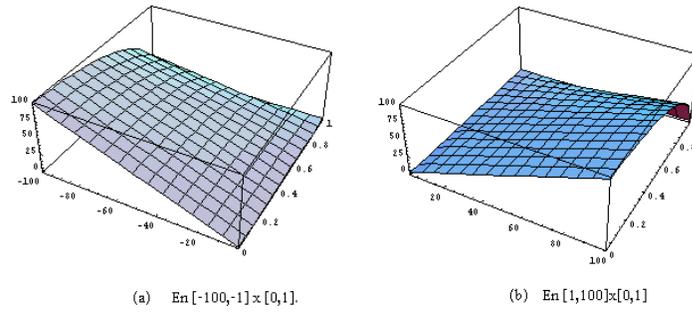


Figura 3.2: Gráficas de la función M

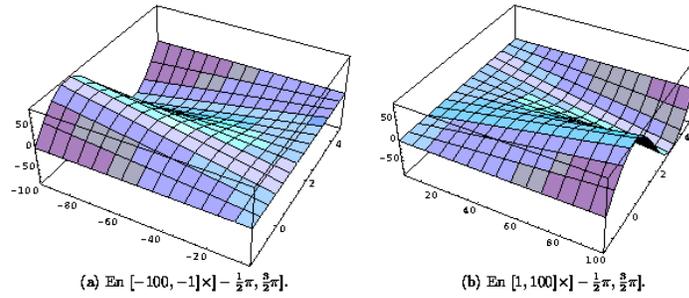


Figura 3.3: Gráficas de la *primera* componente de Φ

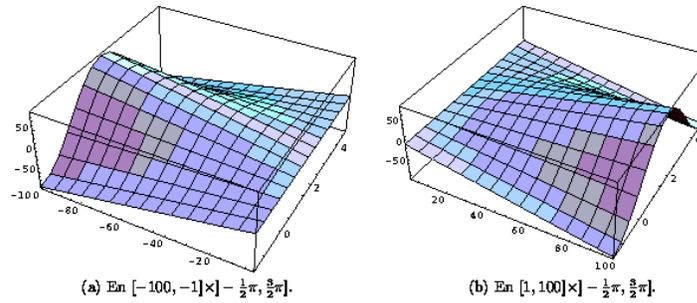


Figura 3.4: Gráficas de la *segunda* componente de Φ

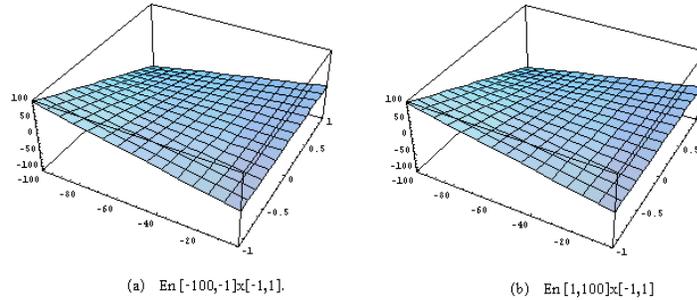


Figura 3.5: Gráficas de la *tercera* componente de Φ

La matriz jacobiana $J\Phi(\eta, \xi, \theta)$ del *cambio de variables* Φ es entonces

$$\begin{bmatrix} \frac{(1-\xi^2)\eta \cos(\theta)}{\sqrt{(1-\xi^2)(-1+\eta^2)}} & -\frac{\xi(-1+\eta^2) \cos(\theta)}{\sqrt{(1-\xi^2)(-1+\eta^2)}} & -\sqrt{(1-\xi^2)(-1+\eta^2)} \sin(\theta) \\ \frac{(1-\xi^2)\eta \sin(\theta)}{\sqrt{(1-\xi^2)(-1+\eta^2)}} & -\frac{\xi(-1+\eta^2) \sin(\theta)}{\sqrt{(1-\xi^2)(-1+\eta^2)}} & \sqrt{(1-\xi^2)(-1+\eta^2)} \cos(\theta) \\ \xi & \eta & 0 \end{bmatrix}$$

y su determinante es, en consecuencia

$$|J\Phi(\eta, \xi, \theta)| = \xi^2 - \eta^2 \quad (3.21)$$

el cual nunca es positivo en todo el dominio de definición $B \times \mathbb{R}$.

Así, si $F : \mathbb{R}^3 \mapsto \mathbb{R}$ es integrable, para cualquier conjunto medible $R \subset \mathbb{R}^3$ se tiene la fórmula de cambio de variable

$$\int_R F(x, y, z) dx dy dz = \int_{\Phi_0^{-1}(R)} F \circ \Phi(\eta, \xi, \theta) (\eta^2 - \xi^2) d\eta d\xi d\theta \quad (3.22)$$

en particular

$$\begin{aligned} \int_{-\infty}^{+\infty} dx \int_{-\infty}^{+\infty} dy \int_{-\infty}^{+\infty} F(x, y, z) dz &= \int_{-\infty}^{-1} d\eta \int_0^1 d\xi \int_{-\frac{\pi}{2}}^{\frac{3\pi}{2}} G(\eta, \xi, \theta) d\theta \\ &+ \int_1^{+\infty} d\eta \int_0^1 d\xi \int_{-\frac{\pi}{2}}^{\frac{3\pi}{2}} G(\eta, \xi, \theta) d\theta \end{aligned} \quad (3.23)$$

donde $G(\eta, \xi, \theta) = F \circ \Phi(\eta, \xi, \theta) (\eta^2 - \xi^2)$

Si $\Phi(\eta, \xi, \theta) = (x, y, z)$ se dice que η, ξ, θ son las *coordenadas esféricas prolatas* del punto (x, y, z) . Por la manera en la que hasta aquí hemos definido al dominio B_0 , tenemos que éste es disconexo. Consideramos ahora el dominio

$$B_1 = [1, +\infty[\times]-1, 1] \times]-\pi, \pi] \quad (3.24)$$

ciertamente conexo, y la restricción $\Phi_1 = \Phi|_{B_1}$. Φ_1 es a su vez una biyección $B_1 \rightarrow \mathbb{R}^3$. La inversa de Φ_1 es $\Phi_1^{-1} : \mathbb{R}^3 \rightarrow B_1, (x, y, z) \mapsto (\eta_1, \xi_1, \theta_1)$ donde, siempre que $(\eta, \xi, \theta) = \Phi_0^{-1}(x, y, z)$ se ha de tener

$$\begin{aligned} \eta_1 &= |\eta| \\ \xi_1 &= \text{signo}(\eta) \cdot \xi \\ \theta_1 &= \begin{cases} \theta & \text{si } -\frac{1}{2}\pi < \theta \leq \pi, \\ \theta - 2\pi & \text{en otro caso.} \end{cases} \end{aligned} \quad (3.25)$$

Para una función $F : \mathbb{R}^3 \mapsto \mathbb{R}$ integrable, para cualquier conjunto medible $R \subset \mathbb{R}^3$ se ha de tener una fórmula de cambio de variable para Φ_1 similar a la fórmula (3.2.1). En particular

$$\int_{-\infty}^{+\infty} dx \int_{-\infty}^{+\infty} dy \int_{-\infty}^{+\infty} F(x, y, z) dz = \int_1^{+\infty} d\eta \int_{-1}^1 d\xi \int_{-\pi}^{\pi} G(\eta, \xi, \theta) d\theta \quad (3.26)$$

3.2.2. Transformación de mallados

El cambio de coordenadas esféricas prolatas puede simplificar las ecuaciones de algunos lugares geométricos. Por ejemplo, la ecuación cartesiana de una esfera con centro (x_0, y_0, z_0) y radio a es $(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = a^2$. Al cambiar a coordenadas prolatas y simplificando tenemos:

$$a^2 = b + c + d \tag{3.27}$$

donde

$$\begin{aligned} b &= \xi^2 \eta^2 + x_0^2 + y_0^2 - 2\xi \eta z_0^2 + z_0^2 \\ c &= -2\sqrt{(1 - \xi^2)(-1 + \eta^2)}(x_0 \cos(\theta) + y_0 \sin(\theta)) \\ d &= (-1 + \xi^2 + \eta^2 - \xi^2 \eta^2) \end{aligned}$$

en particular, para $(x_0, y_0, z_0) = (0, 0, 1)$ la ecuación (3.27) se convierte en $(\xi - \eta)^2 = a^2$.

En la figura 3.6 (a) mostramos las curvas en las que se transforman los segmentos de la forma $\eta = 1 + \frac{k}{7}$, con $-1 \leq \xi \leq 1$, para $k = 0, 1, \dots, 7$. En el diagrama (b) las curvas en las que se transforman los segmentos de la forma $\xi = \frac{7k}{100}$, con $1 \leq \eta \leq 2$, para $k = [-8, 8]$. En el diagrama (c) se muestra el mallado en el plano $\eta - \xi$ consistente de los segmentos arriba descritos y en el (d) el mallado en el que se transforma, que es en realidad la superposición de las gráficas (a) y (b). Tenemos que para toda recta horizontal $\xi = s$ ésta se transforma en la curva $\eta \mapsto (\sqrt{(-1 + \eta^2)(1 - s^2)}, \eta s)$ y toda recta vertical $\eta = r$ en la curva $\xi \mapsto (\sqrt{(-1 + r^2)(1 - \xi^2)}, r\xi)$.

En la figura 3.7 (a) mostramos una malla consistente de segmentos en las rectas diagonales $\eta - \xi = \frac{7+k\frac{3}{7}}{10}$ y en las rectas horizontales $\xi = \frac{7+k\frac{3}{7}}{10}$, con $k = 0, \dots, 7$. En el diagrama (b) mostramos este mallado transformado. Todo segmento en la recta diagonal $\eta - \xi = a$ se transforma en un arco en la circunferencia $x^2 + (z - 1)^2 = a^2$. Para un punto $\delta \in [0, 1]$ y dos particiones, una $\eta_0 = \{1 = \eta_{00} < \eta_{01} < \dots < \eta_{0n_\eta}\}$ del intervalo $[1, \eta_{0n_\eta}]$ y la otra $\xi_0 = \{0 = \xi_{00} < \xi_{01} < \dots < \xi_{0n_\xi} = 1\}$ del intervalo $[0, 1]$, con $\delta = \eta_{0i_0} \in \xi$. Sea $R(\eta_0, \xi_0 : \delta)$ el mallado en el dominio $B_1^+ = B_1 \cap \{(\eta, \xi) | 0 \leq \xi\}$ definido por los segmentos de los siguientes cuatro tipos:

$$\begin{aligned} H_{0\delta} &: (\eta, \xi) = (\eta, \xi_{0i}), & \text{con } \eta \geq 1, i = 0, \dots, i_0. \\ H_{1\delta} &: (\eta, \xi) = (\eta, \xi_{0i}), & \text{con } \eta \geq 1 - \delta + \xi_{0i}, i_0 + 1, \dots, \eta_\xi. \\ V_{0\delta} &: (\eta, \xi) = (\eta_j, \xi), & \text{con } 0 \leq \xi \leq \delta, j = 0, \dots, \eta_\eta. \\ V_{1\delta} &: (\eta, \xi) = (\eta_j - \delta + \xi, \xi), & \text{con } \delta \leq \xi \leq 1, j = 0, \dots, \eta_\eta. \end{aligned}$$

En las figuras 3.6 y 3.7 se ven estos segmentos. En sus figuras (b) se ve la partición de \mathbb{R}^2 definida por $S(\eta_0, \xi_0; \delta) = \Phi(R(\eta_0, \xi_0; \delta))$.

3.3. Procedimiento de integración numérica

3.3.1. Presentación del problema

Sean $\boldsymbol{\eta}_0 = \{\eta_{01}, \eta_{02}, \dots, \eta_{0m}\}$ y $\boldsymbol{\xi}_0 = \{\xi_{01}, \xi_{02}, \dots, \xi_{0n}\}$ sendas particiones de los intervalos $[1, \eta_{0m}]$ y $[0, 1]$. La partición $\boldsymbol{\eta}_0 \times \boldsymbol{\xi}_0$ del rectángulo $B = [1, \eta_{0m}] \times [0, 1]$ naturalmente puede

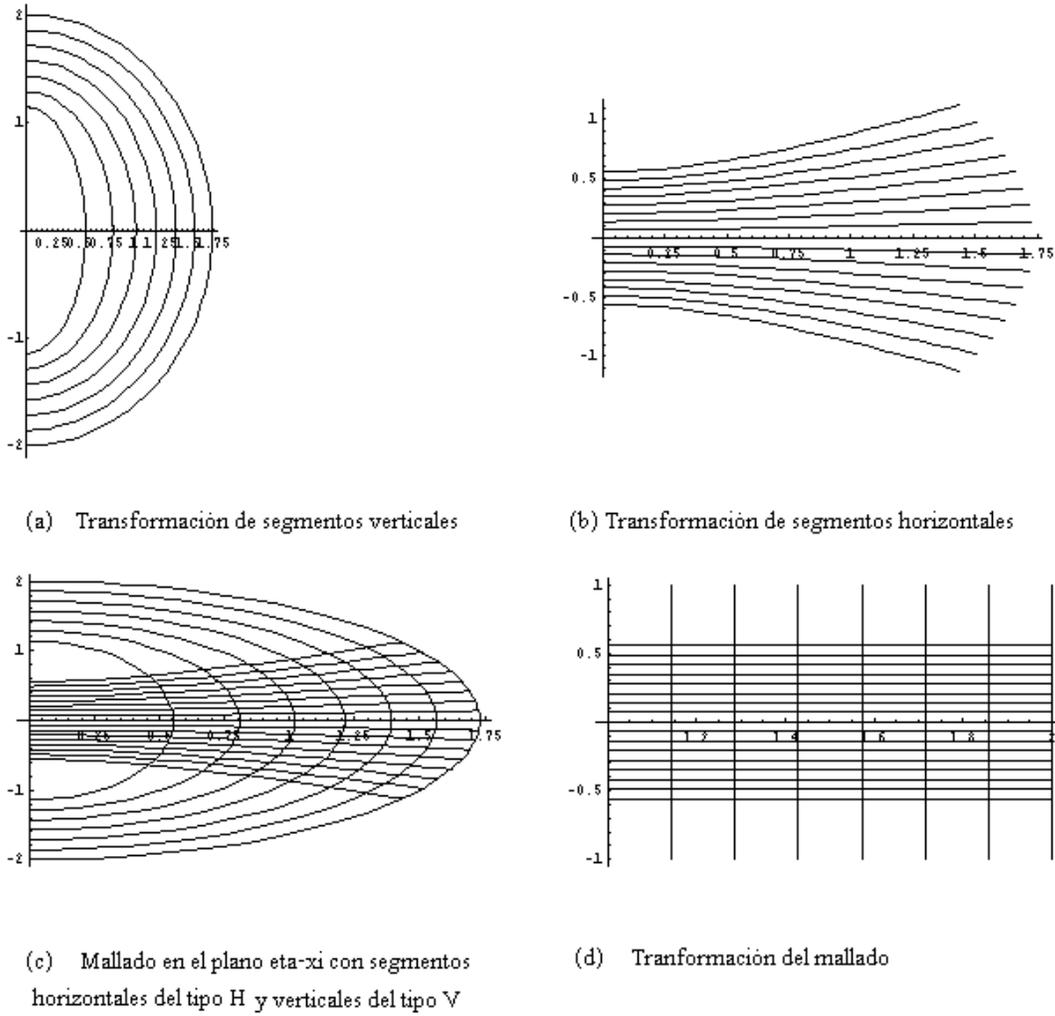


Figura 3.6: Algunos efectos de la transformación Φ

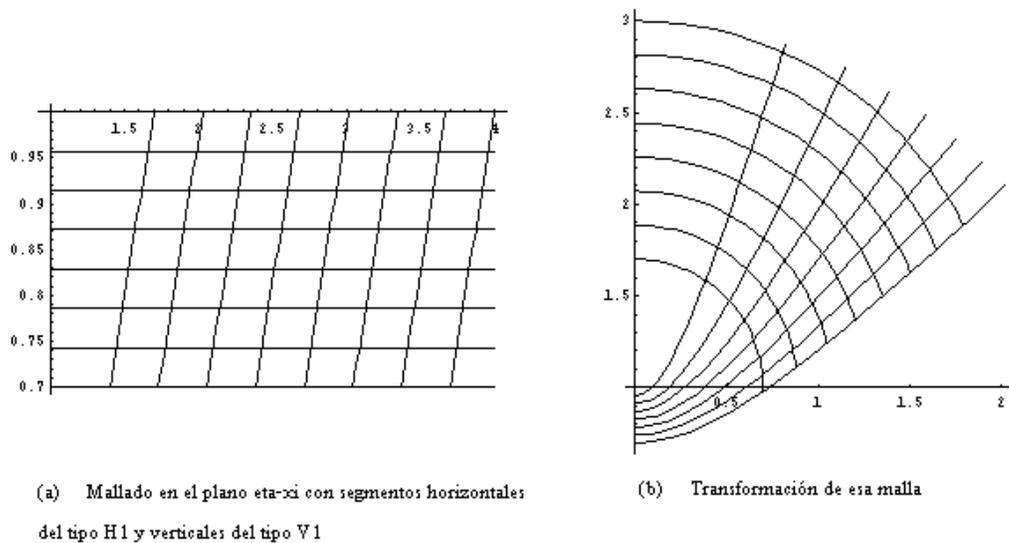


Figura 3.7: Algunos otros efectos de la transformación Φ

identificarse con el producto de intervalos (de números enteros) $I_{mn} = \llbracket 1, m \rrbracket \times \llbracket 1, n \rrbracket$ mediante la correspondencia $(i, j) \mapsto (\eta_j, \xi_i)$. Una función $g : B \rightarrow \mathbb{R}$ puede así representarse, aunque de manera aproximada, mediante la matriz $\mathbf{g} = (g(\eta_j, \xi_i))_{i=1, \dots, m}^{j=1, \dots, n}$ consistente de sus valores en I_{mn} . Recíprocamente, dada una matriz $\mathbf{g} = (g(\eta_j, \xi_i))_{i=1, \dots, m}^{j=1, \dots, n}$, sea $g_{\mathbf{g}}$ la función lineal por trozos que en cada triángulo $\Delta[(\eta_j, \xi_i), (\eta_{j+1}, \xi_i), (\eta_j, \xi_{i+1})]$ interpola linealmente los valores $g_{ij}, g_{i,j+1}, g_{i+1,j}$ respectivamente, y en cada triángulo $\Delta[(\eta_j, \xi_i), (\eta_j, \xi_{i-1}), (\eta_{j-1}, \xi_i)]$ los valores $g_{ij}, g_{i-1,j}, g_{i,j+1}$ respectivamente. Es claro que ambas funciones g y $g_{\mathbf{g}}$ corresponden a la matriz \mathbf{g} según lo anterior.

3.4. Método de Newton-Raphson y convergencia global para el problema de Ornstein-Zernike

Denotemos por \mathbb{R}^B al conjunto de todas las funciones $B \rightarrow \mathbb{R}$ y por $\mathbb{R}^{m \times n}$ al conjunto de matrices reales de orden $m \times n$. Todo operador de funciones $F : \mathbb{R}^B \rightarrow \mathbb{R}^B$ determina un operador de matrices $F_M : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$: Dada una $\mathbf{g} = (g(\eta_j, \xi_i))_{i=1, \dots, m}^{j=1, \dots, n} \in \mathbb{R}^{m \times n}$, sea $h = F(g_{\mathbf{g}})$ la imagen bajo F de la función lineal por trozos determinada por \mathbf{g} y sea \mathbf{h} la matriz correspondiente a h , tomamos entonces $F_M(\mathbf{g}) = \mathbf{h}$.

La búsqueda de un punto fijo del operador F puede verse como la búsqueda de un punto fijo del operador F_M .

Sea $k = m \cdot n$. Enumerando a las matrices por renglones, $\mathbb{R}^{m \times n}$ se identifica con \mathbb{R}^k , el cual tiene naturalmente su propia estructura de espacio normado. De hecho al conjunto de índices I_{mn} lo identificaremos con el conjunto $\llbracket 1, k \rrbracket$ mediante la biyección $(i, j) \mapsto (i-1)n + j$.

Para cada $(i, j) \in I_{mn}$ consideremos un operador $A_{ij} : \mathbb{R}^B \rightarrow \mathbb{R}$. Sean $\mathbf{u} \in \mathbb{R}^k$ y $F : \mathbb{R} \rightarrow \mathbb{R}$ una función real continua. Para cada $(i, j) \in I_{mn}$ sea $h_{ij} : \mathbb{R}^B \rightarrow \mathbb{R}$ el operador

$$h_{ij} : g \mapsto h_{ij}(g) = F(u_{ij} + A_{ij}(g)) \quad (3.28)$$

En base a tales operadores, definimos el operador $H : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$ haciendo

$$\forall \mathbf{g} \in \mathbb{R}^k : \quad H(\mathbf{g}) : (i, j) \mapsto h_{ij}(g_{\mathbf{g}}) \quad (3.29)$$

Tanto F como las funciones A_{ij} son de clase C^2 en sus respectivos espacios de definición, H lo es también. Observamos que la ecuación de punto fijo de H , a saber, $\mathbf{g} = H(\mathbf{g})$ equivale a la de una raíz $G(\mathbf{g}) = 0$ del operador $G : \mathbf{g} \mapsto G(\mathbf{g}) = \mathbf{g} - H(\mathbf{g})$. Procedamos según el método de Newton-Raphson para localizar tal raíz:

$$\begin{aligned} &\text{Elijamos} \quad \mathbf{g}_0 \in \mathbb{R}^k \\ &\text{y sucesivamente} \quad \forall r \geq 0 : \mathbf{g}_{r+1} = \mathbf{g}_r - (DG)^{-1}|_{\mathbf{g}_r} \circ G(\mathbf{g}_r) \end{aligned} \quad (3.30)$$

donde DG es la diferencial del G en el punto \mathbf{g}_r , es decir, $DG|_{\mathbf{g}_r}$ es la transformación lineal representada, respecto a la base canónica de \mathbb{R}^k , por la matriz *jacobiana*

$$JG|_{\mathbf{g}} = [\partial_{k_2} (g_r - h_{k_1}(g_r))]_{k_1=1, \dots, k}^{k_2=1, \dots, k} \quad (3.31)$$

De esta manera se ha de tener un orden de convergencia (cuando el punto inicial \mathbf{g}_0 se elija de manera que en efecto converja la sucesión) al menos cuadrático [12]. Escribamos

$k_l = (i_l, j_l)$ para denotar que se cumple la relación $k_l = (i_l - 1)n + j_l$. Escribamos de manera indistinta k_l sola o la pareja de índices (i_l, j_l) para referirnos a esa posición en vectores de \mathbb{R}^k .

En cuanto a la convergencia global para el problema de Ornstein-Zernike, primeramente es necesario caracterizar los puntos iniciales \mathbf{g} para los cuales el método de Newton-Raphson produce una sucesión convergente.

Suponemos, considerando ahora un caso particular, que para cualesquiera i_1, j_1 la función $A_{i_1 j_1} : \mathbb{R}^k \rightarrow \mathbb{R}$ es lineal. Entonces, respecto a la base canónica, se le puede representar mediante una matriz de orden $1 \times k$: $A_{i_1 j_1} = \left(a_{(i_1, j_1)}^{(i_2, j_2)} \right)^{(i_2, j_2)}$. Así pues, la colección de todos estos operadores puede representarse por la matriz $A = \left(a_{(i_1, j_1)}^{(i_2, j_2)} \right)^{(i_2, j_2)}$ de orden $k \times k$. Evidentemente al calcular las derivadas parciales de los operadores A_{k_1} obtenemos

$$\forall \mathbf{g} : \partial_{k_2} A_{k_1}(\mathbf{g}) = a_{(i_2, j_2)}^{(i_1, j_1)}. \quad (3.32)$$

es decir, cada tal derivada parcial es constante. Ahora, también supongamos que F es la función exponencial, $F : t \mapsto \exp(t)$. Entonces $\frac{dF}{dt}(t) = F(t)$, $\forall t \in \mathbb{R}$. De la regla de la cadena hemos de tener:

$$\forall k_1, k_2, \mathbf{g} : \partial_{k_2} H_{k_1}(\mathbf{g}) = F'(u_{k_1} + A_{k_1}(\mathbf{g})) \partial_{k_2} A_{k_2}(\mathbf{g}) = a_{k_1}^{k_2} H_{k_1}(\mathbf{g}). \quad (3.33)$$

y en consecuencia

$$\forall k_1, k_2, \mathbf{g} : \partial_{k_2} G_{k_1}(\mathbf{g}) = \delta_{k_1, k_2} - a_{k_1}^{k_2} H_{k_1}(\mathbf{g}). \quad (3.34)$$

donde δ_{k_1, k_2} es una delta de Kroenecker.

La matriz jacobiana es pues de la forma $JG(\mathbf{g}) = (\delta_{k_1, k_2} - a_{k_1}^{k_2} H_{k_1}(\mathbf{g}))_{k_1}^{k_2}$. En la realización del procedimiento 3.30, en vez de calcular la inversa del jacobiano $JG(\mathbf{g}_r)$, se puede proceder a calcular su descomposición LU .

En efecto, el paso iterativo del método de Newton Raphson puede presentarse equivalentemente como

$$(DG)|_{\mathbf{g}_r}(\mathbf{g}_{r+1} - \mathbf{g}_r) = G(\mathbf{g}_r) \quad (3.35)$$

así pues, en nuestro caso actual, al resolver el sistema de ecuaciones $JG(\mathbf{g}_r) \cdot \delta_r = G(\mathbf{g}_r)$ se podría hacer $\mathbf{g}_{r+1} = \mathbf{g}_r + \delta_r$. Ahora bien, si $JG(\mathbf{g}_r) = L_r \cdot U_r$, donde L_r es una matriz triangular inferior y U_r es triangular superior entonces δ_r es solución de $U_r \delta_r = \epsilon_r$ donde ϵ_r es a su vez solución de $L_r \epsilon_r = G(\mathbf{g}_r)$. La resolución de ambos sistemas de ecuaciones es sencilla dado que las matrices de coeficientes son triangulares.

Veamos cómo obtener la descomposición LU de una matriz F , $F = L \cdot U$, donde L es una matriz triangular inferior y U es triangular superior. Introduciendo una partición consistente en las matrices involucradas, escribimos

$$\begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ L_{21} & F_{22} \end{bmatrix} \cdot \begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix} = \begin{bmatrix} L_{11}U_{11} & L_{11}U_{12} \\ L_{21}U_{11} & L_{21}U_{11} + L_{11}U_{22} \end{bmatrix}$$

consecuentemente

$$\begin{aligned} L_{11}U_{11} &= F_{11} \\ L_{12} &= L_{11}^{-1}F_{12} \\ L_{21} &= F_{21}U_{11}^{-1} \\ L_{22}U_{22} &= F_{22} - L_{21}U_{12} \end{aligned} \quad (3.36)$$

Es convencional tomar las primeras matrices F_{11} , L_{11} y U_{11} de orden 1×1 , lo cual da un algoritmo de descomposición LU con una complejidad de tiempo k^2 , donde el orden de la matriz F es $k \times k$.

3.5. Principales parámetros involucrados en la resolución de la ecuación de Ornstein-Zernike

3.5.1. Proceso de especificación de mallas planares

En el tratamiento numérico de la ecuación OZ el primer procedimiento consiste en introducir una discretización de ella, partiendo de una discretización de su dominio, y las funciones incógnitas han de ser aproximadas ahí. Dado que el dominio es una región en el plano, los puntos de aproximación quedarán definidos en los vértices de una malla en el plano. Las ecuaciones diferenciales involucradas son aproximadas mediante un conjunto de ecuaciones algebraicas sobre tal colección de puntos, el cual al ser resuelto produce un conjunto discreto de valores que aproximan a la solución del sistema diferencial sobre la malla introducida. La discretización de la ecuación diferencial requiere pues de cierta organización para ser eficiente; esto es, debe ser posible identificar rápidamente los puntos o nodos donde está definida.

Por las particularidades físicas de la ecuación, la malla de aproximación no puede ser uniforme, es decir, no puede consistir de puntos igualmente separados. En diversas regiones de la malla, ha de haber separaciones distintas entre puntos contiguos. Llamemos *diámetro de la malla* a la máxima separación entre dos puntos contiguos en la malla. En la presente sección nos abocaremos al problema de determinar una malla de acuerdo con diversos diámetros establecidos previamente.

Representaciones de mallas

Consideremos un rectángulo en \mathbb{R}^2 de la forma $R = [a, b] \times [c, d]$. Introduzcamos una partición en el intervalo $[a, b]$, de la forma

$$a = X_0 < X_1 < \cdots < X_{m-1} < X_m = b$$

consistente de una sucesión creciente de $m + 1$ puntos, incluidos los extremos, y otra partición similar en el intervalo $[c, d]$, de la forma

$$c = Y_0 < Y_1 < \cdots < Y_{n-1} < Y_n = d$$

consistente de una sucesión creciente de $n + 1$ puntos, incluidos los extremos. La partición $\mathbf{X}_b = (X_j)_{j=0}^m$ divide al intervalo $[a, b]$ en m segmentos $[X_{j-1}, X_j]$, $j \leq m$, y la partición $\mathbf{Y}_d = (Y_i)_{i=0}^n$ divide al intervalo $[c, d]$ en n segmentos $[Y_{i-1}, Y_i]$, $i \leq n$. En cada segmento $[X_{j-1}, X_j]$ consideraremos una separación constante u_j y $m_j + 1$ puntos incluidos los extremos. Así pues tendremos una partición en el intervalo $[X_{j-1}, X_j]$, de la forma

$$X_{j-1} = x_{j,0} < x_{j,1} < \cdots < x_{j,m_j-1} < x_{j,m_j} = X_j$$

donde $x_{j,k} = X_{j-1} + ku_j$, con $k = 0, 1, \dots, m_j$, y $u_j = \frac{X_j - X_{j-1}}{m_j}$. De manera similar, para cada $i \leq n$, en el segmento $[Y_{i-1}, Y_i]$ consideraremos una separación constante v_i y $n_i + 1$ puntos incluidos los extremos. Así pues tendremos una partición en el intervalo $[Y_{i-1}, Y_i]$, de la forma

$$Y_{i-1} = y_{i,0} < y_{i,1} < \dots < y_{i,n_i-1} < y_{i,n_i} = Y_i$$

donde $x_{i,k} = Y_{i-1} + kv_i$, con $k = 0, 1, \dots, n_i$, y $v_i = \frac{Y_i - Y_{i-1}}{n_i}$.

Naturalmente, para cada $j = 1, \dots, m-1$ se tiene $x_{j-1,m_{j-1}} = x_{j,0}$, es decir, el último punto de la partición en el segmento $[X_{j-2}, X_{j-1}]$ coincide con el primero del segmento $[X_{j-1}, X_j]$, y ambos coinciden con X_{j-1} . Así pues, la sucesión de puntos $\mathbf{X}_t = (x_{j,k} | 0 \leq k \leq m_j, j = 0, \dots, m)$ consiste de $M_t = 1 + \sum_{j=1}^m m_j$ puntos. Similarmente, la sucesión de puntos $\mathbf{Y}_t = (y_{i,k} | 0 \leq k \leq n_i, i = 0, \dots, n)$ consiste de $N_t = 1 + \sum_{i=1}^n n_i$ puntos.

Observamos que la partición \mathbf{X}_t queda determinada completamente por medio de cualquiera de las siguientes especificaciones:

Extremo-separación. $\{(X_j, h_j)\}_{j=0}^m$, $h_0 = 0$.

Extremo-número de puntos. $\{(X_j, m_j)\}_{j=0}^m$, con $m_0 = 0$.

Ahora bien, la sucesión \mathbf{X}_t consta de M_t puntos. Para fines de almacenamiento en un programa de computadora y de referencia a sus puntos en cualquier proceso que involucre a sus componentes, se puede colocar esa sucesión en un arreglo I_X consistente de M_t localidades adyacentes, de tipo real. Evidentemente, para cada $\kappa \leq M_t$ existe un índice $j \leq m$ tal que $1 + \sum_{\ell=1}^{j-1} m_\ell < \kappa \leq 1 + \sum_{\ell=1}^j m_\ell$. Sea $k = \kappa - (1 + \sum_{\ell=1}^{j-1} m_\ell)$. Entonces se ha de tener

$$I_X[\kappa] = x_{jk}.$$

De esta manera, los índices de la forma $M_j = 1 + \sum_{\ell=1}^j m_\ell$ corresponderán a los puntos X_j en la sucesión de extremos \mathbf{X}_b . Es claro que $M_j - M_{j-1} = m_j$. Esto permite entonces representar a la partición \mathbf{X}_t , junto con I_X , también de la forma:

Extremo-índice. $\{(X_j, M_j)\}_{j=0}^m$, con $M_0 = 1$.

Como una última forma de representar una malla, observemos ahora que para cada $j \leq m$, $X_j = X_{j-1} + u_j m_j$. Sea $\epsilon > 0$ una constante positiva y sean r_j, s_j dos enteros primos relativos tales que $\frac{r_j}{s_j} \approx \frac{u_j}{\epsilon}$. Entonces

$$m_j = \frac{X_j - X_{j-1}}{u_j} = (X_j - X_{j-1}) \frac{s_j}{\epsilon r_j}$$

y en consecuencia

$$M_j = M_{j-1} + (X_j - X_{j-1}) \frac{s_j}{\epsilon r_j}.$$

Así pues, se puede tener una nueva representación de la partición \mathbf{X}_t :

Tercetas. $\{(X_j, s_j, r_j)\}_{j=0}^m$, con $r_0 = 0$.

Esta última representación tiene una motivación geométrica acaso menos intuitiva pero es igualmente útil. Todas las formas vistas son equivalentes y el cambio entre cualesquiera dos representaciones es inmediato.

Habiendo representado una partición en un intervalo $[a, b]$ y otra en un intervalo $[c, d]$, en el producto cartesiano $[a, b] \times [c, d]$ quedará entonces determinada una malla como el producto cartesiano de ambas particiones.

Procedimiento de intercambio entre representaciones

En diversas aplicaciones se ha utilizado la que aquí hemos llamado de *tercetas* como representación de mallas. La razón $\frac{\text{diametro}}{TE}$ es un mero factor de escala donde TE es el cociente de $\frac{RR}{TAU}$, TAU es la distancia entre las dos partículas en evaluación y RR es la distancia entre una partícula y el eje de las abcisas. Veremos ahora cómo hacemos el intercambio entre representaciones.

Conversión de tercetas a separación-número de puntos y viceversa

Originalmente los datos que se introducen en el sistema son tercetas de la forma (W_j, W_l, W_z) , y los valores del diámetro y TE que descritos anteriormente se encuentran involucrados en estos valores.

De esta manera obtenemos la representación separación-número de puntos descrita anteriormente. Como mera ilustración, consideremos la representación por tercetas:

0.25	80	1
0.75	40	1
1.75	20	1
2.75	10	1
4.75	5	1
14.75	2	1
24.75	1	1

A continuación se presenta el pseudocódigo del algoritmo utilizado para pasar de la representación por tercetas a separación-número de puntos:

```

Para i = 0 hasta i<N'umerodetercetas
  Extremo = (TerceraColumna[i]/SegundaColumna[i])
    (diametro/tau);
  Extremo = Extremo + ExtremoInferior (
    (PrimeraColumna[i]-auxiliar)*
    SegundaColumna[i]/TerceraColumna[i]);
  auxiliar=PrimeraColumna[i]

```

Utilizando el algoritmo anterior obtenemos para el primer bloque 20 subbloques de longitud 0.006250, como se muestra:

0.000000	0
0.006250	20
0.012500	20
0.025000	20
0.050000	10
0.100000	10
0.250000	20
0.500000	10

Para pasar de la representación separación-número de puntos a tercetas, utilizamos el siguiente algoritmo:

Conversión

```
NuevoRefinamiento = Cálculo * tau / diametro;
TerceraColumna = EnteroInferior(
    NuevoRefinamiento * 10^5);
Temporal = MáximoComúnDivisor(TerceraColumna,
    10^5)
SegundaColumna = 10^5 / Temporal;
```

RepresentaciónAnterior

```
Para j = 1 hasta j < NúmerodeRefinamientos
    Cálculo = (Refinamiento[j] -
        Refinamiento[j-1]) / (Extremo[j] -
        Extremo[j-1]);
    SegundaColumna, TerceraColumna =
        Conversión(RefinamientoActual,
            diámetro, tau );
    PrimeraColumna = auxiliar + (
        Extremo[ j ] - Extremo[j-1]) *
        TerceraColumna / SegundaColumna;
    auxiliar = PrimeraColumna;
```

Conversión de tercetas a extremo-índice y viceversa

Se tiene que $\frac{W_l}{W_z} * \frac{diametro}{TE}$ coincide con el refinamiento que se utiliza en la malla, y para obtener el índice hasta donde cada refinamiento se utiliza se toma la parte entera del número real $X_j - X_{j-1} * \frac{W_l}{W_z}$.

Como mera ilustración, consideremos la representación por tercetas presentada en la sección anterior. A continuación presentamos el pseudocódigo del algoritmo utilizado para pasar de la representación tercetas a extremo-índice:

IncrementoIndice

```
Para i=0 hasta i < Numero de tercetas
    Extremo = ( TerceraColumna[i] / SegundaColumna[i] ) * diametro / tau
    Extremo = Extremo + EnteroInferior(( PrimeraColumna - auxiliar ) *
        SegundaColumna / TerceraColumna )
    Indice = Indice + 1;
    auxiliar = PrimeraColumna;
```

NuevaRepresentación

```
ExtremoAuxiliar = IncrementoIndice
IndiceAuxiliar = IncrementoIndice
Para i =1 hasta i<Numero de tercetas
    auxiliar1=ExtremoAuxiliar
    Indice= IndiceAuxiliar
    auxiliar2 = Indice - auxiliar3
    Extremo = Extremo + auxiliar1 * auxiliar2
    auxiliar3=Indice
```

Utilizando el algoritmo anterior obtenemos para el primer bloque que nos encontramos en la coordenada de extremo de segmento el extremo 0.125000 y en el índice 21, como se muestra a continuación.

0.000000	1
0.125000	21
0.375000	41
0.875000	61
1.375000	71
2.375000	81
7.375000	101
12.375000	111

Ahora nos abocaremos a mostrar el procedimiento a partir de una interfaz de entrada que permita introducir como datos los relativos a representaciones de mallas planares de la forma: coordenadas de extremos de segmentos, y los índices que ocuparán en los arreglos numéricos definidos (I_X e I_Y). A partir de esta especificación podemos obtener la representación equivalente por tercetas. Es posible involucrar en su definición datos como un diámetro de partícula (con valor numérico del orden de 10^{-8}) y una separación τ entre partículas.

Aquí mostramos el algoritmo para pasar de la representación extremo-índice a tercetas:

Division1

```
auxiliar = DivisionRestoExtremoIndice * tau / diametro;
TerceraColumnaAuxiliar = EnteroInferior ( auxiliar * 10^5);
SegundaColumnaAuxiliar = 10^5
auxiliar2 = MaximoComunDivisor( TerceracolumnaAuxiliar,
    SegundaColumnaAuxiliar );
auxiliar3 = SegundaColumnaAuxiliar / auxiliar2
```

Division2

```
auxiliar = DivisionRestoExtremoIndice * ( tau / diametro );
TerceraColumnaAuxiliar = EnteroInferior ( auxiliar * 10^5);
SegundaColumnaAuxiliar = 10^5
auxiliar2 = MaximoComunDivisor( TerceraColumnaAuxiliar,
    SegundaColumnaAuxiliar );
auxiliar4 = TerceraColumnaAuxiliar / auxiliar2
```

Transformacion

```
Para j=0 hasta j<NumerodeDatos
    DivisionRestoExtremoIndice = ( Extremo[j] - Extremo[j-1] )
        / ( Indice[j] - Indice[j-1] )
    SegundaColumna = Division1(DivisionRestoExtremoIndice,
        NumerodeDatos)
    TerceraColumna = Division2(DivisionRestoExtremoIndice,
        NumerodeDatos)
    PrimeraColumna = Auxiliar5+( Indice[j] - Indice[j-1])*
        TerceraColumna/SegundaColumna
```

Conversión de tercetas a extremo-número de puntos y viceversa

La representación extremo-número de puntos puede ser la más intuitiva desde el punto de vista geométrico. La representación por tercetas es la utilizada por mera convención. Para pasar de la representación por tercetas a la representación por extremo-número de puntos, se utiliza el algoritmo que se muestra a continuación:

```

ExtremoAuxiliar = IncrementoIndice
ExtremoAuxiliar2 = IncrementoIndice
Extremo=ExtremoAuxiliar
auxiliar4= ExtremoAuxiliar2
Para i = 0 hasta i < Numerodetercetas
    auxiliar1 = ExtremoAuxiliar
    auxiliar2[i] = ExtremoAuxiliar2
    auxiliar3 = auxiliar2 - auxiliar4
    Extremo = Extremo + auxiliar1 * auxiliar3
    Numerodepuntos=auxiliar2[i] - auxiliar2[i-1]

```

Para ejemplificar el procedimiento utilizamos la representación por tercetas mostrada anteriormente y obtenemos la representación separación-índice:

```

0.000000 1
0.006250 21
0.012500 41
0.025000 61
0.050000 71
0.100000 81
0.250000 101
0.500000 111

```

Para cambiar de la representación separación-índice a la representación por tercetas utilizamos el siguiente algoritmo:

```

Para j = 1 hasta j < NúmerodeRefinamientos
    Cálculo = (Refinamiento[j] -
        Refinamiento[j-1]) / (Extremo[j] -
        Extremo[j-1]);
    SegundaColumna, TerceraColumna =
        Conversión(RefinamientoActual,
        diámetro, tau );
    PrimeraColumna = auxiliar + (
        Extremo[ j ] - Extremo[j-1]) *
        TerceraColumna / SegundaColumna;
    auxiliar = PrimeraColumna;

Para j=0 hasta j<NumerodeDatos
    DivisionRestoExtremoIndice = ( Extremo[j] - Extremo[j-1] )
        / ( Indice[j] - Indice[j-1] )
    SegundaColumna = Division1(DivisionRestoExtremoIndice,
        NumerodeDatos)
    TerceraColumna = Division2(DivisionRestoExtremoIndice,
        NumerodeDatos)
    PrimeraColumna = Auxiliar5+( Indice[j] - Indice[j-1])*
        TerceraColumna/SegundaColumna

```

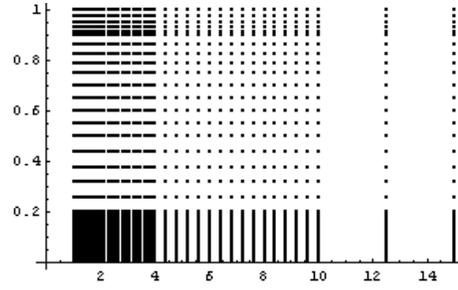


Figura 3.8: Malla determinada por $P \times (Q_1 \cup Q_2)$

Todas las representaciones creadas son equivalentes y de manera similar como ya se vio se pueden hacer conversiones entre ellas.

Ejemplo

En el tratamiento numérico de la ecuación OZ es necesario calcular un arreglo de valores sobre una malla definida en el plano cartesiano con coordenadas $\eta - \xi$.

Por ejemplo, en el intervalo $J_1 = [0., 0.5]$ consideremos la partición cuya representación extremo-número de puntos es

$$Q_1 = \begin{pmatrix} 0. & 0.05 & 0.1 & 0.2 & 0.5 \\ 0 & 20 & 15 & 10 & 5 \end{pmatrix}$$

y en el intervalo $J_2 = [0.5, 1.]$ consideremos la partición cuya representación extremo-número de puntos es

$$Q_2 = \begin{pmatrix} 0.5 & 0.75 & 0.9 & 0.95 & 1. \\ 0 & 5 & 4 & 3 & 2 \end{pmatrix}$$

Por otro lado, en el intervalo $I = [1., 15.]$ consideremos la partición cuya representación extremo-número de puntos es

$$P = \begin{pmatrix} 1. & 2. & 4. & 8. & 10. & 15. \\ 0 & 20 & 15 & 10 & 5 & 2 \end{pmatrix}$$

En la figura 3.20 se representa a la malla determinada por el producto cartesiano $P \times (Q_1 \cup Q_2)$.

En la figura 3.9 se representa a la imagen de esa malla al aplicar el cambio de variables dado por la transformación Φ descrita en la sección 3.2.1.

Ahora, a cada punto en la submalla de arriba $P \times Q_2$ apliquemos la transformación $(\eta, \xi) \mapsto (\eta + \xi - 0,5, \xi)$, la cual hará que los segmentos verticales se conviertan en líneas paralelas a la diagonal principal. Denotemos por R a esa nueva malla. En la figura 3.10 se representa a la malla $(P \times Q_1) \cup R$.

En la figura 3.11 se representa a la imagen de esa malla al aplicar el cambio de variables Φ .

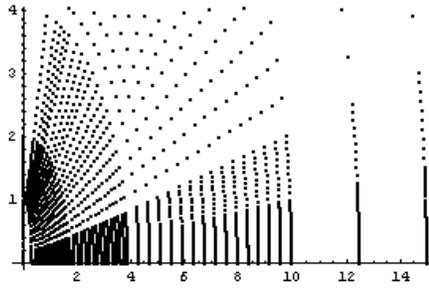


Figura 3.9: Malla $\Phi(P \times (Q_1 \cup Q_2))$

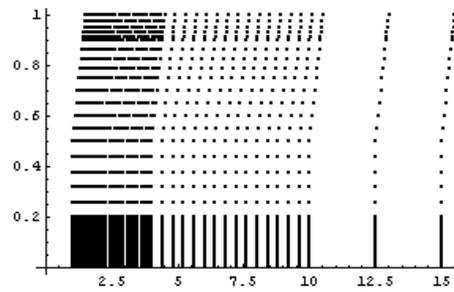


Figura 3.10: Malla $(P \times Q_1) \cup R$

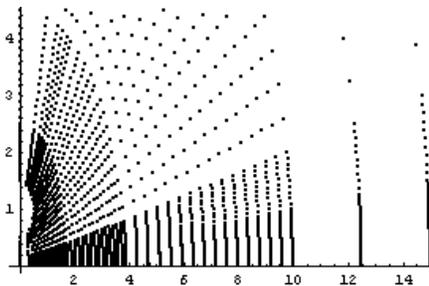


Figura 3.11: Malla $\Phi((P \times Q_1) \cup R)$

3.6. Cálculo de funciones núcleo

En la ecuación 3.14 encontramos la función $K_l(\eta, \xi, \eta_3, \xi_3)$, la cual representa el cálculo principal de la ecuación OZ, para dar solución a esta función se requiere analizarla positiva y negativamente, pues nuestro espacio de trabajo se encuentra en los ejes positivo y negativo del plano cartesiano. Así para dos puntos $(\eta_1, \xi_1), (\eta_3, \xi_3) \in B_1$ se calculan dos valores $XF^+(\eta_1, \xi_1; \eta_3, \xi_3)$ y $XF^-(\eta_1, \xi_1; \eta_3, \xi_3)$, de esta manera definimos.

$$g_{13}^+ = (n_1^2 - 1)(1 - \xi_1^2) + (\eta_3^2 - 1)(1 - \xi_3^2) + (\xi_1\eta_1 - \xi_3\eta_3)^2 \quad (3.37)$$

$$g_{13}^- = (n_1^2 - 1)(1 - \xi_1^2) + (\eta_3^2 - 1)(1 - \xi_3^2) + (\xi_1\eta_1 - \xi_3\eta_3)^2 \quad (3.38)$$

$$h_{13} = 2\sqrt{(1 - \xi_1^2)(1 - \xi_3^2)(\eta_1^2 - 1)(\eta_3^2 - 1)}$$

Para resolver la cerradura de la ecuación descrita en la sección 2.2 se tienen que usar técnicas de métodos numéricos debido a la imposibilidad para obtener soluciones analíticas. El método de elemento finito¹ ya ha sido usado anteriormente para resolver la ecuación HCA/MSA en varias geometrías y ha demostrado ser un método eficiente. Utilizando MEF podemos obtener los valores de los pesos involucrados en nuestro problema, realizamos el proceso de integración con base en triángulos, aquí los hemos definido como constantes.

$a_0 =$	1.38629436112	$b_0 =$	0.500000000000
$a_1 =$	0.09666344259	$b_1 =$	0.12498593597
$a_2 =$	0.03590092383	$b_2 =$	0.06880248576
$a_3 =$	0.03742563713	$b_3 =$	0.03328355346
$a_4 =$	0.01451196212	$b_4 =$	0.00441787012

hagamos

$$f_1(x) = \left(\sum_{i=0}^4 a_i \left(\frac{x - h_{13}}{x + h_{13}} \right)^i \right) - \left(\sum_{i=0}^4 a_i \left(\frac{x - h_{13}}{x + h_{13}} \right)^i \right) \log_{10} \left(\frac{x - h_{13}}{x + h_{13}} \right) \quad (3.39)$$

$$f_2(x) = \sqrt{x + h_{13}} \quad (3.40)$$

$$f_3(x) = \tau^2 (\eta_3^2 - \xi_3^2) \frac{f_1(x)}{f_2(x)} \quad (3.41)$$

entonces definimos las funciones núcleo positiva y negativa

$$XF^+(\eta_1, \xi_1; \eta_3, \xi_3) \quad XF^-(\eta_1, \xi_1; \eta_3, \xi_3)$$

Para resolver la ecuación descrita anteriormente se utiliza el procedimiento que calcula una función $R^2 \times R^2 \rightarrow R^2$.

Primero, realizan las asignaciones siguientes:

¹De aquí en adelante MEF

```
othzee=(pow(eta,2)-1)*(1-pow(xsi,2));
stothzeep=pow(((xsi1*eta1)+(xsi3*eta3)),2);
stothzeen=pow(((xsi1*eta1)-(xsi3*eta3)),2);
```

El siguiente procedimiento realiza el cálculo de las funciones 3.39, 3.40, 3.41, y es utilizado en todas las funciones núcleo descritas

Función

```
a0 = 1.38629436112
a1 = 0.09666344259
a2 = 0.03590092383
a3 = 0.03742563713
a4 = 0.01451196212
b0 = 0.50000000000
b1 = 0.12498593597
b2 = 0.06880248576
b3 = 0.03328355346
b4 = 0.00441787012

r1 = ( xa - xb0 )/( xa + xb0 )
r2 = pow(r1,2)
r3 = pow(r1,3)
r4 = pow(r1,4)

a = a0 + a1 * r1 + a2 * r2 + a3 * r3 + a4 * r4
b = ( b0 + b1 * r1 + b2 * r2 + b3 * r3 + b4 * r4 ) * log( r1 )
fst = a - b
yst = sqrt( xa + xb0 )

temp = (pow(tau,2) * ( pow(eta3,2) - pow(xsi3,2) )) * fst / yst
```

La función principal para la rutina que calcula los primeros núcleos de la ecuación OZ es:

```
z1 = ZXkernel(eta1,xsi1)
z3 = ZXkernel(eta3,xsi3)
xap = z1 + z3 + Z13Xkernel(eta1,xsi1,eta3,xsi3)
xb0 = 2 * sqrt( z1 * z3 )
xkplp =Función( tau, xap, xb0, eta3, xsi3)
```

El procedimiento principal para la rutina que hace el cálculo de los segundos núcleos de la ecuación OZ es:

```
npd2=0
te = tau/2
xap = z1 + z3+Z13Xkernel(eta1,xsi1,eta3,xsi3)
xb0 = 2 * sqrt( z1 * z3 )
rm = te * sqrt( xap + xb0 )
Si ( rn >=diam )
    xkp = 0
en caso contrario
```

```

ni = 2
np = 64
nxx = ni*np
xmax = acos( -pow(diam,2)+xap*pow(te,2))/(xb0*pow(te,2))
Si ( rm < diam )
    xmax = pi
xmin = 0
pesos( np, npd2 )
recibepesos( ni, np, npd2, xmax, xmin )
Para i1= 1 hasta i1<=nxx
    s13 = te * sqrt( xap - xb0 + cos( xp[ i1 ] ) )
    xa[ i1 ] = s13
    xb[ i1 ] = pow(s13,3)
    xc[ i1 ] = 1/s13
sum1 = 0
sum2 = 0
Para j = 1 hasta j <= nxx
    m =( xku + xkd * xa[ j ] + xkt * xb[ j ] )
    sum1 = sum1 - 2 * wt[ j ] * m
    n =( xc[ j ] + fun1 + fun2 * xa[ j ] )
    sum2 = sum2 + 2 * wt[ j ] * n
    xkp = sum1 * ( pow(eta3,2) - pow(xsi3,2 ) ) * pow(te,3)

```

El procedimiento principal para la rutina que hace el cálculo de los terceros núcleos de la ecuación OZ es:

```

eta31=eta3*(-1)
npd2=0
te = tau/2
xap = z1 + z3+Z13Xkernel(eta1,xsi1,eta31,xsi3)
xb0 = 2 * sqrt( z1 * z3 )
rm = te * sqrt( xap + xb0 )
Si ( rn >=diam )
    xkp = 0
en caso contrario
    ni = 2
    np = 64
    nxx = ni*np
    xmax = acos( -pow(diam,2)+xap*pow(te,2))/(xb0*pow(te,2))
    Si ( rm < diam )
        xmax = pi
    xmin = 0
    pesos( np, npd2 )
    recibepesos( ni, np, npd2, xmax, xmin )
    Para i1 = 1 hasta i1 <= nxx
        s13 = te * sqrt( xap - xb0 + cos( xp[ i1 ] ) )
        xa[ i1 ] = s13
        xb[ i1 ] = pow(s13,3)
        xc[ i1 ] = 1/s13
    sum1 = 0
    sum2 = 0
    Para j = 1 hasta j <= nxx
        m = ( xku + xkd * xa[ j ] + xkt * xb[ j ] )
        sum1 = sum1 - 2 * wt[ j ] * m
        n = ( xc[ j ] + fun1 + fun2 * xa[ j ] )

```

$$a_1 = 0,57735027 \quad w_1 = 1,00000000$$

Tabla 3.1: Pesos cuando $IPQ \in J_1$ en *Quadr*.

$$\begin{aligned} a_1 &= 0,33998104 & w_1 &= 0,65214515 \\ a_2 &= 0,86113631 & w_2 &= 0,34785484 \end{aligned}$$

Tabla 3.2: Pesos cuando $IPQ \in J_2$ en *Quadr*.

$$\begin{aligned} \text{sum2} &= \text{sum2} + 2 * \text{wt}[j] * n \\ \text{xkp} &= \text{sum1} * (\text{pow}(\text{eta31}, 2) - \text{pow}(\text{xsi3}, 2)) * \text{pow}(\text{te}, 3) \end{aligned}$$

Como se muestra la diferencia entre los procedimientos para el cálculo de los segundos y terceros núcleos de la ecuación OZ radica únicamente en el cambio de signo en el valor de eta3 , pero este implica que el procedimiento converja mas rápidamente. Para obtener las gráficas solo es necesario introducir los parámetros en las funciones y ejecutarlas.

3.6.1. Cálculo de raíces de operadores en términos de las funciones que calculan los núcleos de la ecuación de Ornstein-Zernike

En función de su argumento de entrada la rutina *Pesos* define dos listas de pesos y un valor de salida *NPD2*.

La rutina *Pesos* y la rutina *RecibePesos* comparten 4 arreglos: a, w, wt y xp . Los dos primeros son pesos y se determinan en *Pesos*. Los otros dos quedan determinados en *RecibePesos*.

Si $IPQ \leq 7$ se hace $NPD2 = \lfloor \frac{IPQ}{2} \rfloor$ y se define los pesos según las relaciones en los recuadros (5.4), (5.5) o (5.6) de acuerdo con que $NPD2 = 1, 2$ o 3 .

Si $IPQ \geq 8$ se hace $NDIR = \text{Min}\{\lfloor \frac{IPQ}{8} \rfloor, 5\}$, $IPQ = \text{Min}\{IPQ, 64\}$ y $NPD2 = \lfloor \frac{IPQ}{2} \rfloor$ y se define los pesos según las relaciones en los recuadros (5.7), (5.8), (5.9), (5.10) o (5.11) de acuerdo con el valor de *NPD2*. En otras palabras, al considerar los intervalos de enteros:

$$\begin{aligned} J_1 &= [2, 4[& J_2 &= [4, 6[& J_3 &= [6, 8[& J_4 &= [8, 16[\\ J_5 &= [16, 24[& J_6 &= [24, 32[& J_7 &= [32, 40[& J_8 &= [40, +\infty[\end{aligned}$$

entonces si el valor de IPQ cae en J_i se toma el conjunto de pesos en el i -ésimo recuadro. Observamos que los vectores de pesos, para cada caso, tienen longitudes

$$\ell_1 = 1, \ell_2 = 2, \ell_3 = 3, \ell_4 = 4, \ell_5 = 8, \ell_6 = 12, \ell_7 = 16, \ell_8 = 32.$$

Los vectores presentados en las tablas son los pesos definidos en el programa

Los vectores w son coeficientes de combinaciones convexas: Estas coordenadas son siempre positivas y estan normalizados a 1 por la 1-norma. Los vectores a son positivos y se incrementan en consecuencia de la tendendica a 1 del último elemento.

$a_1 = 0,23861919$	$w_1 = 0,46791393$
$a_2 = 0,66120939$	$w_2 = 0,36076157$
$a_3 = 0,93246951$	$w_3 = 0,17132449$

Tabla 3.3: Pesos cuando $IPQ \in J_3$ en Quadr.

$a_1 = 0,1834346424$	$w_1 = 0,3626837834$
$a_2 = 0,5255324099$	$w_2 = 0,3137066459$
$a_3 = 0,7966664774$	$w_3 = 0,2223810344$
$a_4 = 0,9602898565$	$w_4 = 0,1012285362$

Tabla 3.4: Pesos cuando $IPQ \in J_4$ en Quadr.

$a_1 = 9,5012509838 \times 10^{-02}$	$w_1 = 1,8945061046 \times 10^{-01}$
$a_2 = 2,8160355078 \times 10^{-01}$	$w_2 = 1,8260341504 \times 10^{-01}$
$a_3 = 4,5801677766 \times 10^{-01}$	$w_3 = 1,6915651940 \times 10^{-01}$
$a_4 = 6,1787624440 \times 10^{-01}$	$w_4 = 1,4959598882 \times 10^{-01}$
$a_5 = 7,5540440836 \times 10^{-01}$	$w_5 = 1,2462897126 \times 10^{-01}$
$a_6 = 8,6563120239 \times 10^{-01}$	$w_6 = 9,5158511682 \times 10^{-02}$
$a_7 = 9,4457502307 \times 10^{-01}$	$w_7 = 6,2253523939 \times 10^{-02}$
$a_8 = 9,8940093499 \times 10^{-01}$	$w_8 = 2,7152459412 \times 10^{-02}$

Tabla 3.5: Pesos cuando $IPQ \in J_5$ en Quadr.

$a_1 = 6,4056892863 \times 10^{-02}$	$w_1 = 1,2793819535 \times 10^{-01}$
$a_2 = 1,9111886747 \times 10^{-01}$	$w_2 = 1,2583745635 \times 10^{-01}$
$a_3 = 3,1504267970 \times 10^{-01}$	$w_3 = 1,2167047293 \times 10^{-01}$
$a_4 = 4,3379350763 \times 10^{-01}$	$w_4 = 1,1550566805 \times 10^{-01}$
$a_5 = 5,4542147139 \times 10^{-01}$	$w_5 = 1,0744427012 \times 10^{-01}$
$a_6 = 6,4809365194 \times 10^{-01}$	$w_6 = 9,7618652104 \times 10^{-02}$
$a_7 = 7,4012419158 \times 10^{-01}$	$w_7 = 8,6190161532 \times 10^{-02}$
$a_8 = 8,2000198597 \times 10^{-01}$	$w_8 = 7,3346481411 \times 10^{-02}$
$a_9 = 8,8641552700 \times 10^{-01}$	$w_9 = 5,9298584915 \times 10^{-02}$
$a_9 = 8,8641552700 \times 10^{-01}$	$w_{10} = 4,4277438817 \times 10^{-02}$
$a_{10} = 9,3827455200 \times 10^{-01}$	$w_{11} = 2,8531388629 \times 10^{-02}$
$a_{11} = 9,7472855597 \times 10^{-01}$	$w_{12} = 1,2341229800 \times 10^{-02}$
$a_{12} = 9,9518722000 \times 10^{-01}$	

Tabla 3.6: Pesos cuando $IPQ \in J_6$ en Quadr.

$a_1 = 4,8307665688 \times 10^{-02}$	$w_1 = 9,6540088515 \times 10^{-02}$
$a_2 = 1,4447196158 \times 10^{-01}$	$w_2 = 9,5638720079 \times 10^{-02}$
$a_3 = 2,3928736225 \times 10^{-01}$	$w_3 = 9,3844399081 \times 10^{-02}$
$a_4 = 3,3186860228 \times 10^{-01}$	$w_4 = 9,1173878696 \times 10^{-02}$
$a_5 = 4,2135127613 \times 10^{-01}$	$w_5 = 8,7652093004 \times 10^{-02}$
$a_6 = 5,0689990893 \times 10^{-01}$	$w_6 = 8,3311924227 \times 10^{-02}$
$a_7 = 5,8771575724 \times 10^{-01}$	$w_7 = 7,8193895787 \times 10^{-02}$
$a_8 = 6,6304426693 \times 10^{-01}$	$w_8 = 7,2345794109 \times 10^{-02}$
$a_9 = 7,3218211874 \times 10^{-01}$	$w_9 = 6,5822222776 \times 10^{-02}$
$a_{10} = 7,9448379597 \times 10^{-01}$	$w_{10} = 5,8684093479 \times 10^{-02}$
$a_{11} = 8,4936761373 \times 10^{-01}$	$w_{11} = 5,0998059262 \times 10^{-02}$
$a_{12} = 8,9632115577 \times 10^{-01}$	$w_{12} = 4,2835898022 \times 10^{-02}$
$a_{13} = 9,3490607594 \times 10^{-01}$	$w_{13} = 3,4273862913 \times 10^{-02}$
$a_{14} = 9,6476225559 \times 10^{-01}$	$w_{14} = 2,5392065309 \times 10^{-02}$
$a_{15} = 9,8561151155 \times 10^{-01}$	$w_{15} = 1,6274394731 \times 10^{-02}$
$a_{16} = 9,9726386185 \times 10^{-01}$	$w_{16} = 7,0186100095 \times 10^{-03}$

Tabla 3.7: Pesos cuando $IPQ \in J_7$ en *Quadr*.

La rutina *RecibePesos*, recibe 4 argumentos de entrada: $n = \text{NSUB}$, $I = \text{IPQ}$, $\text{NPD2} = \frac{I}{2}$, $x_{Mx} = \text{XMAX}$ y $x_{Mn} = \text{XMIN}$. n es el número de subdivisiones del intervalo $[x_{Mn}, x_{Mx}]$ e I es el número de nodos en cada subdivisión.

Se hace $\delta = \frac{1}{n}(x_{Mx} - x_{Mn})$, $c = \frac{\delta}{2}$ y para cada $i \leq n$, $x_i = x_{Mn} + i\delta$.

Para cada $i \in \llbracket 1, n \rrbracket$ se procede como sigue:

Sean $t_i = \frac{1}{2}(x_i + x_{i-1})$ el punto medio del intervalo real $[x_{i-1}, x_i]$ y $K_i = I(i - \frac{1}{2})$ el punto medio del i -ésimo intervalo entero consistente de I enteros consecutivos. Entonces, para $j \leq \frac{I}{2}$ se define

$$\begin{aligned} wt_{K_i+j} &= t_i + ca_j & xp_{K_i+j} &= cr_j \\ wt_{K_i+1-j} &= t_i - ca_j & xp_{K_i+1-j} &= cr_j \end{aligned} \quad (3.42)$$

donde los arreglos \mathbf{a} y \mathbf{r} , en \mathbb{R}^I , están dados previamente. Así pues, los arreglos de salida \mathbf{wt} y \mathbf{xp} son de dimensión $n \cdot I$.

El algoritmo de la función *RecibePesos* es:

```

k = -npd
delta = ( xmax - xmin )/ (float) nsub
b = xmin
j = 1
Para i = 1 hasta i <= nsub
    a = b
    b = b + delta
    trans = ( b + a )/ 2.0
    coeff = ( b - a )/ 2.0
    k = k + ipq
    l = k + 1
Si ( npd - j >= 0 )

```

$a_1 = 2,4350292663 \times 10^{-02}$	$w_1 = 4,8690957009 \times 10^{-02}$
$a_2 = 7,2993121788 \times 10^{-02}$	$w_2 = 4,8575467442 \times 10^{-02}$
$a_4 = 1,6964442042 \times 10^{-01}$	$w_3 = 4,8344762235 \times 10^{-02}$
$a_3 = 1,2146281930 \times 10^{-01}$	$w_4 = 4,7999388596 \times 10^{-02}$
$a_5 = 2,1742364374 \times 10^{-01}$	$w_5 = 4,7540165715 \times 10^{-02}$
$a_6 = 2,6468716221 \times 10^{-01}$	$w_6 = 4,6968182816 \times 10^{-02}$
$a_7 = 3,1132287199 \times 10^{-01}$	$w_7 = 4,6284796581 \times 10^{-02}$
$a_8 = 3,5722015834 \times 10^{-01}$	$w_8 = 4,5491627927 \times 10^{-02}$
$a_9 = 4,0227015796 \times 10^{-01}$	$w_9 = 4,4590558164 \times 10^{-02}$
$a_{10} = 4,4636601725 \times 10^{-01}$	$w_{10} = 4,3583724529 \times 10^{-02}$
$a_{11} = 4,8940314571 \times 10^{-01}$	$w_{11} = 4,2473515124 \times 10^{-02}$
$a_{12} = 5,3127946402 \times 10^{-01}$	$w_{12} = 4,1262563243 \times 10^{-02}$
$a_{13} = 5,7189564620 \times 10^{-01}$	$w_{13} = 3,9953741133 \times 10^{-02}$
$a_{14} = 6,1115535517 \times 10^{-01}$	$w_{14} = 3,8550153179 \times 10^{-02}$
$a_{14} = 6,1115535517 \times 10^{-01}$	$w_{15} = 3,7055128540 \times 10^{-02}$
$a_{15} = 6,4896547125 \times 10^{-01}$	$w_{16} = 3,5472213257 \times 10^{-02}$
$a_{16} = 6,8523631305 \times 10^{-01}$	$w_{17} = 3,3805161837 \times 10^{-02}$
$a_{17} = 7,1988185017 \times 10^{-01}$	$w_{18} = 3,2057928355 \times 10^{-02}$
$a_{18} = 7,5281990726 \times 10^{-01}$	$w_{19} = 3,0234657072 \times 10^{-02}$
$a_{19} = 7,8397235894 \times 10^{-01}$	$w_{20} = 2,8339672614 \times 10^{-02}$
$a_{20} = 8,1326531512 \times 10^{-01}$	$w_{21} = 2,6377469715 \times 10^{-02}$
$a_{21} = 8,4062929625 \times 10^{-01}$	$w_{22} = 2,4352702569 \times 10^{-02}$
$a_{22} = 8,6599939815 \times 10^{-01}$	$w_{23} = 2,2270173808 \times 10^{-02}$
$a_{23} = 8,8931544600 \times 10^{-01}$	$w_{24} = 2,0134823154 \times 10^{-02}$
$a_{24} = 9,1052213708 \times 10^{-01}$	$w_{25} = 1,7951715776 \times 10^{-02}$
$a_{25} = 9,2956917213 \times 10^{-01}$	$w_{26} = 1,5726030476 \times 10^{-02}$
$a_{26} = 9,4641137486 \times 10^{-01}$	$w_{27} = 1,3463047897 \times 10^{-02}$
$a_{27} = 9,6100879965 \times 10^{-01}$	$w_{28} = 1,1168139460 \times 10^{-02}$
$a_{28} = 9,7332682779 \times 10^{-01}$	$w_{29} = 8,8467598264 \times 10^{-03}$
$a_{29} = 9,8333625388 \times 10^{-01}$	$w_{30} = 6,5044579690 \times 10^{-03}$
$a_{30} = 9,9101337148 \times 10^{-01}$	$w_{31} = 4,1470332606 \times 10^{-03}$
$a_{31} = 9,9634011677 \times 10^{-01}$	$w_{32} = 1,7832807217 \times 10^{-03}$
$a_{32} = 9,9930504174 \times 10^{-01}$	

Tabla 3.8: Pesos cuando $IPQ \in J_8$ en Quadr.

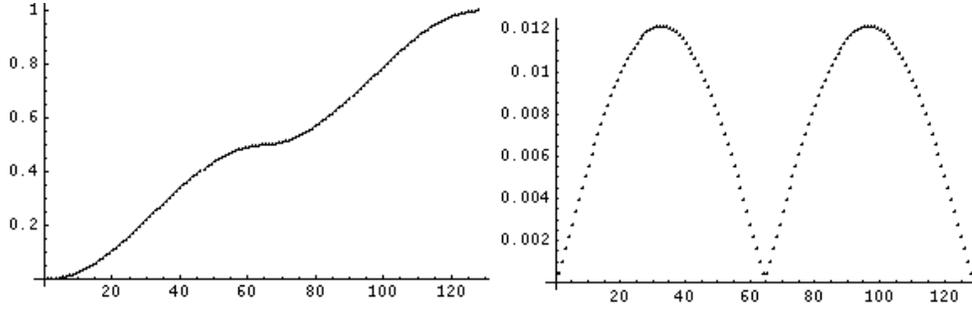


Figura 3.12: Gráfica determinada por abc y wei , arreglos generados en 4arg

```

m = k + j
n = l - j
part = coeff + ar[ j ]
xp[ m ] = trans + part
xp[ n ] = trans - part
part = coeff * w[ j ]
wt[ n ] = part
wt[ m ] = part
j = j + 1

```

Como ejemplo se puede mostrar que para $nrp = 2$, $isl = 8$, $xmx, xmn = 1., 0$, $abc, wei = Pesos[acf, wcf, nrp, isl, xmx, xmn]$ se obtienen las gráficas de la figura 3.12

3.7. Cálculo de las funciones que involucran a los procedimientos núcleo en términos de la ecuación de Ornstein-Zernike

Para cada punto $(\eta_1, \xi_1) \in \mathcal{M}$ se define aquí a 6 matrices

$$\begin{aligned}
 \mathbf{XF1}(\eta_1, \xi_1) &= (\mathbf{XF1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XF4}(\eta_1, \xi_1) = (\mathbf{XF4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} \\
 \mathbf{XYK1}(\eta_1, \xi_1) &= (\mathbf{XYK1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XYK4}(\eta_1, \xi_1) = (\mathbf{XYK4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} \\
 \mathbf{XYL1}(\eta_1, \xi_1) &= (\mathbf{XYL1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XYL4}(\eta_1, \xi_1) = (\mathbf{XYL4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}}
 \end{aligned}$$

Describiremos el proceso de cálculo considerando de manera genérica a una matriz

$$\mathbf{XF}(\eta_1, \xi_1) = (F(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}}$$

que puede ser cualquiera de las 6 calculadas aquí.

Consideremos la triangulación \mathcal{T} en la región B_1 cuyos nodos son los puntos en \mathcal{M} . \mathcal{T} consiste de los triángulos obtenidos al dividir mediante la diagonal (Norte-Oeste)-(Sur-Este) a cada cuadrángulo

$$C_{ij} = [\eta^{(j)}, \eta^{(j+1)}] \times [\xi^{(i)}, \xi^{(i+1)}], \quad i < n_2 + n_3, \quad j < n_1.$$

Cada punto $\mathbf{m}_{i_3 j_3} = (\eta^{(j_3)}, \xi^{(i_3)}) \in \mathcal{M}$ que no sea fronterizo es un vértice de 6 triángulos en \mathcal{T} :

$$\begin{aligned} T_1(\mathbf{m}_{i_3 j_3}) &= \triangle(\mathbf{m}_{i_3 j_3}, \mathbf{m}_{i_3, j_3+1}, \mathbf{m}_{i_3+1, j_3}) \quad ; \quad T_2(\mathbf{m}_{i_3 j_3}) = \triangle(\mathbf{m}_{i_3 j_3}, \mathbf{m}_{i_3+1, j_3}, \mathbf{m}_{i_3+1, j_3-1}) \\ T_3(\mathbf{m}_{i_3 j_3}) &= \triangle(\mathbf{m}_{i_3 j_3}, \mathbf{m}_{i_3+1, j_3-1}, \mathbf{m}_{i_3, j_3-1}) \quad ; \quad T_4(\mathbf{m}_{i_3 j_3}) = \triangle(\mathbf{m}_{i_3 j_3}, \mathbf{m}_{i_3, j_3-1}, \mathbf{m}_{i_3-1, j_3}) \\ T_5(\mathbf{m}_{i_3 j_3}) &= \triangle(\mathbf{m}_{i_3 j_3}, \mathbf{m}_{i_3-1, j_3}, \mathbf{m}_{i_3-1, j_3+1}) \quad ; \quad T_6(\mathbf{m}_{i_3 j_3}) = \triangle(\mathbf{m}_{i_3 j_3}, \mathbf{m}_{i_3-1, j_3+1}, \mathbf{m}_{i_3, j_3+1}) \end{aligned}$$

Sea $D(\mathbf{m}_{i_3 j_3}) = \bigcup_{\iota=1}^6 T_\iota(\mathbf{m}_{i_3 j_3})$ la unión de los seis triángulos que contienen al punto $\mathbf{m}_{i_3 j_3}$.

Se procede de manera similar a la de calcular integrales de la forma

$$F(\eta_1, \xi_1; \mathbf{m}_{i_3 j_3}) \approx \int_{D(\mathbf{m}_{i_3 j_3})} h(\eta_1, \xi_1; \eta, \xi) d\eta d\xi = \sum_{\iota=1}^6 \int_{T_\iota(\mathbf{m}_{i_3 j_3})} h(\eta_1, \xi_1; \eta, \xi) d\eta d\xi \quad (3.43)$$

en donde cada una de las integrales $\int_{T_\iota(\mathbf{m}_{i_3 j_3})} h(\eta_1, \xi_1; \eta, \xi) d\eta d\xi$ se calcula mediante el elemento finito cuyo conjunto $\phi_\iota(V_{(13)})$ de nodos consta de 13 puntos, y es una imagen isomorfa del conjunto $V_{(13)} = (\mathbf{v}_\ell)_{\ell=1}^{13}$ con pesos de integración

$$\begin{aligned} w_1^{(13)} &= -0,149570044467670 \\ w_2^{(13)} &= 0,175615257433204 \quad w_3^{(13)} = 0,175615257433204 \quad w_4^{(13)} = 0,175615257433204 \\ w_5^{(13)} &= 0,053347235608839 \quad w_6^{(13)} = 0,053347235608839 \quad w_7^{(13)} = 0,053347235608839 \\ w_8^{(13)} &= 0,077113760890257 \quad w_9^{(13)} = 0,077113760890257 \quad w_{10}^{(13)} = 0,077113760890257 \\ w_{11}^{(13)} &= 0,077113760890257 \quad w_{12}^{(13)} = 0,077113760890257 \quad w_{13}^{(13)} = 0,077113760890257 \end{aligned} \quad (3.44)$$

(obsérvese que estos pesos coinciden por renglones). Tendríamos pues

$$F(\eta_1, \xi_1; \mathbf{m}_{i_3 j_3}) = \sum_{\iota=1}^6 \text{área}(T_\iota) \sum_{\ell=1}^{13} w_\ell^{(13)} h(\eta_1, \xi_1; \phi_\iota(\mathbf{v}_\ell)). \quad (3.45)$$

Sin embargo, se procede de otra manera. Se hace un recorrido por grupos de 4 cuadrángulos de la forma C_{ij} . De hecho, se supone que n_1 y $n_T = n_2 + n_3$ son impares. Entonces en cada una de las coordenadas η - ξ hay un número par de intervalos. $\forall \mu \leq \frac{n_1-1}{2}$, $\nu \leq \frac{n_2+n_3-1}{2}$ sea

$$\begin{aligned} R_{\mu, \nu} &= C_{2\mu-1, 2\nu-1} \cup C_{2\mu-1, 2\nu} \cup C_{2\mu, 2\nu-1} \cup C_{2\mu, 2\nu} \\ &= [\eta^{(2\nu-1)}, \eta^{(2\nu)}] \times [\xi^{(2\mu-1)}, \xi^{(2\mu)}] \cup [\eta^{(2\nu)}, \eta^{(2\nu+1)}] \times [\xi^{(2\mu-1)}, \xi^{(2\mu)}] \cup \\ &= [\eta^{(2\nu-1)}, \eta^{(2\nu)}] \times [\xi^{(2\mu)}, \xi^{(2\mu+1)}] \cup [\eta^{(2\nu)}, \eta^{(2\nu+1)}] \times [\xi^{(2\mu)}, \xi^{(2\mu+1)}] \end{aligned}$$

el dominio obtenido como unión de 4 cuadrángulos. La triangulación arriba descrita induce en ese dominio una subtriangulación más y podemos separar los triángulos resultantes en dos subcolecciones. Las sumatorias en 5.6 pueden calcularse por separado en cada uno de los triángulos del dominio de la subtriangulación.

De acuerdo con la ec. (3.14), la función integrando h es el producto de una función básica, en el elemento finito, por un núcleo.

Dado un punto $(\eta, \xi) \in B_1$ existen $\mu \leq \frac{n_1-1}{2}$, $\nu \leq \frac{n_2+n_3-1}{2}$ tales que $(\eta, \xi) \in R_{\mu, \nu}$ y ahí se tiene que

T_{inf}		T_{sup}	
\mathbf{m}	$f_{\mathbf{m}}$	\mathbf{m}	$f_{\mathbf{m}}$
1	$f_1 = G_{21} = L_1(2L_1 - 1)$	7	$f_7 = G_{21} = L_1(2L_1 - 1)$
2	$f_2 = G_{24} = 4L_1L_2$	8	$f_8 = G_{26} = 4L_3L_1$
3	$f_3 = G_{22} = L_2(2L_2 - 1)$	9	$f_9 = G_{24} = 4L_1L_2$
4	$f_4 = G_{26} = 4L_3L_1$	10	$f_{10} = G_{23} = L_3(2L_3 - 1)$
5	$f_5 = G_{25} = 4L_2L_3$	11	$f_{11} = G_{25} = 4L_2L_3$
6	$f_6 = G_{23} = L_3(2L_3 - 1)$	12	$f_{12} = G_{22} = L_2(2L_2 - 1)$

Tabla 3.9: Funciones asociadas a vértices en triángulos.

- $(\eta, \xi) \in T_{inf} = \Delta(\mathbf{m}_{2\mu-1, 2\nu-1}, \mathbf{m}_{2\mu+1, 2\nu-1}, \mathbf{m}_{2\mu-1, 2\nu+1})$ o bien
- $(\eta, \xi) \in T_{sup} = \Delta(\mathbf{m}_{2\mu+1, 2\nu-1}, \mathbf{m}_{2\mu+1, 2\nu+1}, \mathbf{m}_{2\mu-1, 2\nu+1})$.

Los vértices del triángulo en el que está (η, ξ) son

$$(\mathbf{L}_I, \mathbf{L}_J, \mathbf{L}_K) = \begin{cases} (\mathbf{m}_{2\mu-1, 2\nu-1}, \mathbf{m}_{2\mu+1, 2\nu-1}, \mathbf{m}_{2\mu-1, 2\nu+1}) & \text{si } (\eta, \xi) \in T_{inf}, \\ (\mathbf{m}_{2\mu+1, 2\nu-1}, \mathbf{m}_{2\mu+1, 2\nu+1}, \mathbf{m}_{2\mu-1, 2\nu+1}) & \text{si } (\eta, \xi) \in T_{sup}. \end{cases}$$

Sean L_I, L_J, L_K las correspondientes funciones coordenadas de área. Estas son funciones afines tales que $L_M(\mathbf{L}_N) = \delta_{MN}$. Consideremos las seis funciones básicas, que dan una interpolación cuadrática. De acuerdo con su posición relativa en los triángulos T_{inf} o T_{sup} , a cada punto \mathbf{m} se le asocia una función $f_{\mathbf{m}}$ según se ve en el recuadro 5.13. La ec. (5.6) se escribe entonces como

$$F(\eta_1, \xi_1; \mathbf{m}_{i_3j_3}) = \sum_{\iota=1}^6 \text{área}(T_\iota) \sum_{\ell=1}^{13} w_\ell^{(13)} f_{\mathbf{m}}(\phi_\ell(\mathbf{v}_\ell)) g(\eta_1, \xi_1; \phi_\ell(\mathbf{v}_\ell)). \quad (3.46)$$

Las funciones g , para cada una de las seis matrices definidas aquí, están construídas como sigue:

XF1: $g_{\mathbf{XF1}}(\eta_1, \xi_1; \eta, \xi) = XF^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en Núcleo.

XYK1: $g_{\mathbf{XYK1}}(\eta_1, \xi_1; \eta, \xi) = XK^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en Núcleo1.

XYL1: $g_{\mathbf{XYL1}}(\eta_1, \xi_1; \eta, \xi) = XL^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en Núcleo1.

XF4: $g_{\mathbf{XF4}}(\eta_1, \xi_1; \eta, \xi) = XF^-(\eta_1, \xi_1; \eta, \xi)$, según se calcula en Núcleo.

XYK4: $g_{\mathbf{XYK4}}(\eta_1, \xi_1; \eta, \xi) = XK^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en Núcleo2.

XYL4: $g_{\mathbf{XYL4}}(\eta_1, \xi_1; \eta, \xi) = XL^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en Núcleo2.

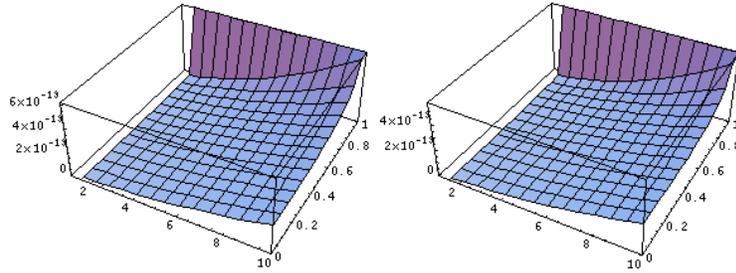


Figura 3.13: Gráfica determinada por $\eta_1 = 2$ y $\xi_1 = 0.5$, *nucleo1* positivo y negativo

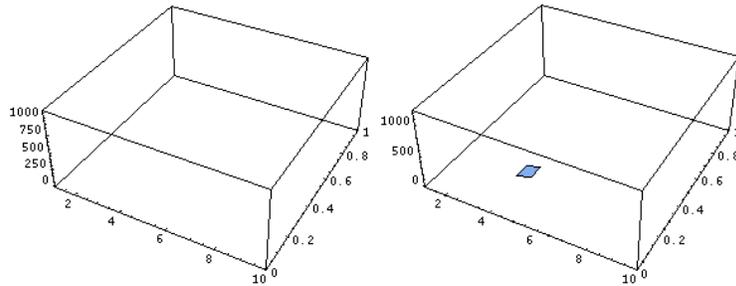


Figura 3.14: Gráfica determinada por $\eta_1 = 1$ y $\xi_1 = 1$, *nucleo1* positivo y negativo

3.8. Despliegue de soluciones

Las gráficas generadas por *nucleo1* nos permiten ver que siendo el punto inicial $\eta_1 = 2$, y $\xi_1 = 0.5$ hay una crecimiento en la grafica desde un inicio resultando una media cúpula en la parte positiva y el mismo crecimiento sucede al utilizar la funcion en su parte negativa. Pero siendo el punto inicial $\eta_1 = 1$, y $\xi_1 = 1$ todos los valores son infinitos en la parte positiva y en la parte negativa hay una pequeña region que contiene valores reales con aproximadamente la misma forma. Siendo el punto inicial $\eta_1 = 1$, y $\xi_1 = 0$ todos los valores son infinitos en la parte positiva y en la negativa. Cuando el valor de $\eta_1 = 1.1$, y $\xi_1 = 0.1$ hay una crecimiento en la gráfica similar a la primera gráfica, siendo $\eta_1 = 21.1$, y $\xi_1 = 0.1$ tenemos mas valores reales en nuestro plano de trabajo lo cual nos permite observar que la distribución de la función indica que cuando se esta más cerca de $\xi_1 = 0$ η tiende a infinito y si nos alejamos del punto inicial $(0, 0)$ tenemos valores cercanos a 0. Por lo que con un pequeño cambio en los valores de η y ξ la función *nucleo1* cae en funciones de *overflow* y *underflow*.

Las gráficas generadas por *nucleo2* nos permiten ver que siendo el punto inicial $\eta_1 = 1$, y $\xi_1 = 0$ hay una pronunciada caída en la gráfica y posteriormente tiene una forma uniforme. Pero siendo el punto inicial $\eta_1 = 1.1$, y $\xi_1 = 0.1$ el comportamiendo es suave en la caída de la gráfica en la presentación positiva mientras que en la negativa es bastante más drástica. Siendo el punto inicial $\eta_1 = 3$, y $\xi_1 = 0.1$ la curva ocupa menos extensión que en el primer caso. Siendo el punto inicial $\eta_1 = 3$, y $\xi_1 = 0.1$, la curva ocupa menos extensión, que en el primer caso. Cuando el valor de $\eta_1 = 3$, y $\xi_1 = 0.1$ existe una region pequeña donde

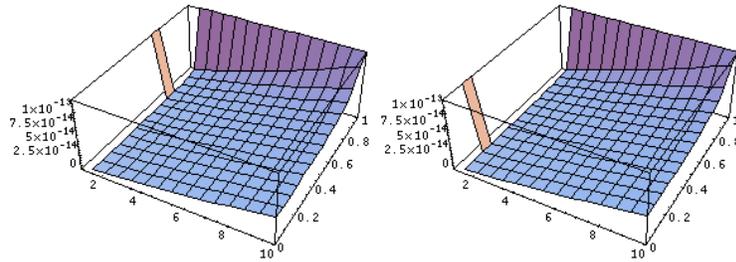


Figura 3.15: Gráfica determinada por $\eta_1 = 21.1$ y $\xi_1 = 0.1$, **nucleo1** positivo y negativo

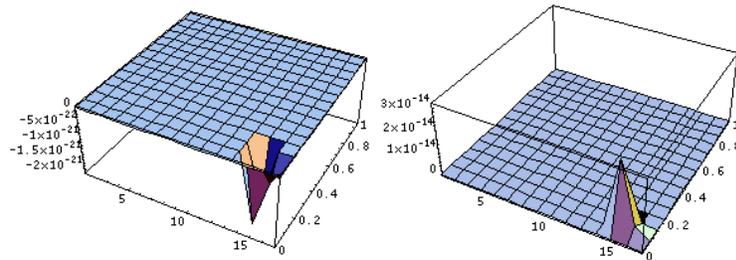


Figura 3.16: Gráfica determinada por $\eta_1 = 15$ y $\xi_1 = 0.1$, **nucleo2** positivo y negativo

hay cambio de valores sobre todo en la evaluación negativa. Siendo $\eta_1 = 15$, y $\xi_1 = 0.1$ aparentemente es simétrico su comportamiento, siendo $\eta_1 = 1.1$, y $\xi_1 = 0.5$ la curva es suave y extensa. Si el valor de $\eta_1 = 3$, y $\xi_1 = 0.5$ existe solo un área central en la cual se muestra un crecimiento y aparentemente es simétrico. Siendo $\eta_1 = 15$ y $\xi_1 = 0.5$ existe una pequeña región donde hay un cambio drástico de valores, y su comportamiento se muestra similar en la evaluación positiva y negativa. Si $\eta_1 = 1$ y $\xi_1 = 1$ hay una variación extensa, mostrándose simétrica en los dos polos, cuando $\eta_1 = 3$ y $\xi_1 = 1$ hay un cambio importante al final de la gráfica pues cambia casi al terminar ésta. Cuando tomamos $\eta_1 = 15$ y $\xi_1 = 1$ en la esquina última se presenta el cambio de valores en nuestro plano de trabajo lo cual nos permite concluir que con un pequeño cambio en los valores de η y ξ . La distribución de la función **nucleo2** nos permite ver que en general el resultado tiende a 0, con variaciones muy concretas dependiendo de los valores introducidos, cuando escojemos una coordenada cerca del punto $(0,0)$ la distribución esta dada en los valores cercanos y si movemos nuestros puntos a lo largo del plano en sus alrededores se da una distribución diferente de 0. La característica de esta función nos permite tomar puntos a lo largo del plano muestra sin muchos problemas de *underflow* y *overflow*.

Las gráficas generadas por **nucleo3** nos permiten ver que siendo el punto inicial $\eta_1 = 1$, y $\xi_1 = 0$ hay una cambio importante en la gráfica en una extensión similar al comportamiento en este mismo punto en **nucleo2**. Pero siendo el punto inicial $\eta_1 = 1.1$, y $\xi_1 = 0.1$ la onda es un poco mas pequeña, tomando $\eta_1 = 3$ y $\xi_1 = 0.1$ la gráfica se comporta de manera simétrica aparentemente. Cuando hacemos $\eta_1 = 15$ y $\xi_1 = 0.1$ todos los valores en la gráfica son 0, siendo $\eta_1 = 1.1$ y $\xi_1 = 0.5$ la gráfica tiene un pequeño crecimiento

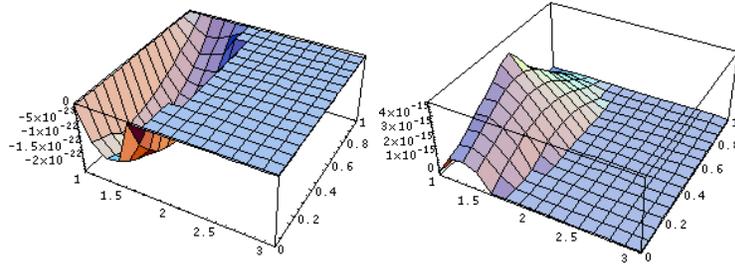


Figura 3.17: Gráfica determinada por $\eta_1 = 1.1$ y $\xi_1 = 0.5$, nucleo2 positivo y negativo

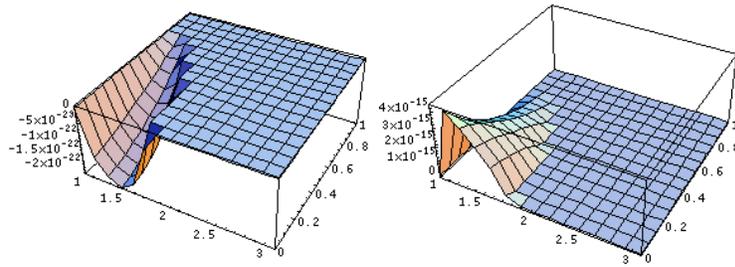


Figura 3.18: Gráfica determinada por $\eta_1 = 1.1$ y $\xi_1 = 0.1$, nucleo3 positivo y negativo

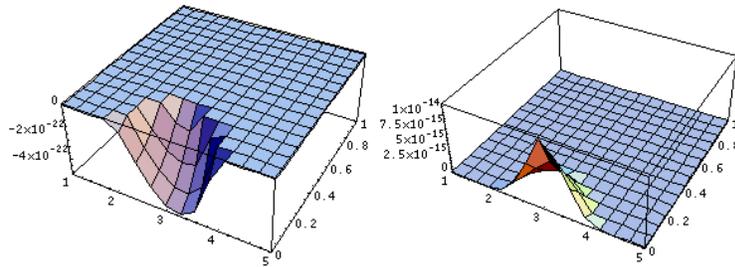


Figura 3.19: Gráfica determinada por $\eta_1 = 3$ y $\xi_1 = 0.1$, nucleo3 positivo y negativo

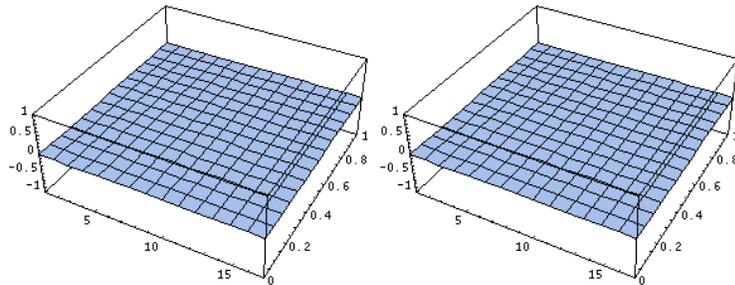


Figura 3.20: Gráfica determinada por $\eta_1 = 15$ y $\xi_1 = 0.1$, nucleo3 positivo y negativo

solo en su parte inicial, y los demás valores se conservan en 0. Si $\eta_1 = 3$ y $\xi_1 = 0.5$, todos los valores de la gráfica son 0, al igual que cuando $\eta_1 = 15$ y $\xi_1 = 0.5$, cuando $\eta_1 = 1$ y $\xi_1 = 1$, cuando $\eta_1 = 3$ y $\xi_1 = 1$, siendo $\eta_1 = 15$ y $\xi_1 = 1$, lo que nos permite observar que hay muchas más regiones en las que el resultado del procedimiento es cero.

Capítulo 4

Alternativa de diseño para el sistema de cómputo que resuelve la ecuación OZ

4.1. Antecedentes

En el Instituto Mexicano del Petróleo, en la ciudad de México, se ha construido un sistema en Fortran-90 que resuelve la ecuación OZ . Sin embargo, los principales parámetros están incorporados al código, por lo que en cada variación de ellos se debe compilar el sistema. Además requiere de muchos recursos en tiempo y en espacio para una ejecución eficiente. El problema de la presente tesis consistió en construir una interfaz en PHP[2] y MySQL[13], para manejar el sistema en un servidor y poder consultar los resultados actuales generados en el transcurso de su ejecución e incluso permite consultar los resultados de un proceso que ha sido finalizado en fechas anteriores, mediante un navegador compaginable con herramientas de despliegue gráfico, básicamente en GNUPLOT[11], para mostrar en forma gráfica y en archivos de texto la solución de la ecuación. Para esto se realizaron pruebas del comportamiento de la solución cuando se refina la malla del problema y se varían los parámetros físicos.

4.2. Diseño del sistema de acceso al servidor *Web*

El procedimiento del sistema inicia en la página *Web* donde a un usuario se le solicita una contraseña para acceder dependiendo del tipo de contraseña que tenga se le permite llevar a cabo diferentes procedimientos en el sistema. Existen 3 tipos de usuario, Administrador, Autorizado y Sin privilegios.

Si el usuario actual introdujo la contraseña de Sin privilegios, las partes del sistema que puede utilizar se limita a realizar pruebas sobre el sistema, es decir, le permite introducir datos que sirven únicamente para graficar funciones que no requieren mucho tiempo de cómputo al sistema, como son la graficación de la malla, la malla desplazada, los comportamientos de las funciones *nucleo1*, *nucleo2*, *nucleo3* positivas y negativas, y su conversión de todas estas gráficas a *coordenadas esféricas prolatas*.

En caso de que la contraseña introducida haya sido Autorizado, el usuario tiene la posibilidad de escoger entre iniciar el proceso del sistema completo, consultar el proceso actual o alguno que ya haya sido concluido ó solo hacer pruebas con algunos parámetros para gráficar funciones como son: la malla, la malla desplazada, los comportamientos de las funciones `nucleo1`, `nucleo2`, `nucleo3` positivas y negativas, y su conversión de todas estas gráficas a *coordenadas esféricas prolatas*, también es posible gráficar las funciones que necesitan de las rutinas `nucleo1`, `nucleo2`, `nucleo3` positivas y negativas, esta opción requiere, según los parámetros de entrada, mucho tiempo de ejecución. Este usuario también puede cancelar algún proceso que inicio el mismo y que aún no ha concluido, así como borrar alguno que no le sea de utilidad según su perspectiva.

El usuario que introduzca la contraseña de Administrador, tiene el privilegio de dar de alta, baja ó modificar a todos los usuarios, que pretendan acceder al sistema, así mismo puede consultar o borrar procesos de los usuarios, pero no iniciarlos ni hacer pruebas, para esto tendría que acceder con una cuenta de usuario Autorizado.

4.3. Base de datos

La base de datos la realizamos en *MySQL* con el nombre `Petroleo`. La base de datos contiene solamente una la cual se describe a continuación:

- Usuario

Y ésta conformada por la definición de los siguientes campos:

Usuario

Campo	Descripción	Tipo
TipoUsuario	Almacena el tipo de usuario los tipos, son Administrador, Autorizado, son No Autorizado	char(1)
Nombre	Guarda el nombre de un usuario	varchar(30)
login	Nombre corto dado a un usuario permiso para utilizar el sistema	varchar(10)
contraseña	Clave de acceso de un usuario	varchar(16)

La contraseña es encriptada con una función propia de *MySQL*, de manera que aún teniendo acceso a la base de datos la contraseña no es mostrada, si se llega a extraviar lo recomendable es generar una nueva contraseña, con el propósito de mantener seguridad con respecto a quién puede acceder al sistema y cuidar su integridad, esta nueva contraseña solo la puede generar el administrador desde su sesión.

4.4. Diseño del sistema en lenguaje C

El sistema en FORTRAN lee un arreglo de parámetros, que durante el desarrollo del sistema se convierte en la especificación de una malla, la desventaja es que los parámetros en realidad poco aportan a la comprensión natural de lo que generan.

CAPÍTULO 4. ALTERNATIVA DE DISEÑO PARA EL SISTEMA DE CÓMPUTO QUE RESUELVE LA ECUACIÓN OZ

Inicialmente el sistema recibe parámetros P_G que determinan la geometría de la malla que se utilizará para resolver el problema numérico y los parámetros correspondientes a las características físicas del problema modelado. Una primera labor consiste en calcular en específico los puntos en la malla. Luego, es necesario calcular las componentes de la ecuación OZ (véase la ecuación 3.6). Los términos M_{ij} involucran, en algunos casos dependiendo de los parámetros P_G , los cálculos de funciones núcleo (que involucran operaciones mediante técnicas de *aproximación numérica* e integración en una variable. Estas dependen de parejas de puntos en la malla). En ésta ecuación, los términos C_{ij} involucran procesos extensos de integración numérica, en el plano, de las funciones núcleo.

Hecho lo anterior, la solución buscada es un punto fijo de la ecuación OZ. Para localizarlo, se parte de un punto inicial, y luego se aplica el MNR. En cada paso del método es necesario resolver un sistema de ecuaciones lineales. Habiendo localizado el punto fijo se procede a desplegar los resultados.

En la implementación, propuesta como alternativa, se dan facilidades para visualizar las diversas etapas del procedimiento descrito arriba. Se puede desplegar, como una gráfica tridimensional con dominio en la malla, las funciones M_{ij} y las funciones núcleos cuando son utilizadas para calcular las funciones C_{ij} . Como los núcleos dependen de parejas de puntos en la malla, se considera al primer punto en una pareja como un parámetro y la gráfica se realiza haciendo variar al segundo punto en la pareja.

El motivo del rediseño de esta parte del sistema es hacerlo intuitivo geoméricamente, pues al solicitar los datos es necesario que se tenga una idea clara de lo que se requiere y así no desperdiciar los recursos con los que cuenta el sistema, obteniendo los resultados esperados.

No todas las rutinas hechas en FORTRAN se rediseñaron pues una de las ventajas de este lenguaje de programación es su cercanía al hardware de la máquina ya que permite en un sistema complejo que tiene múltiples funciones matemáticas, agilizar la ejecución con respecto a otros lenguajes de programación. Ciertamente el sistema completo se tardaba cantidades de tiempo considerables, sin embargo, de haber traducido al sistema totalmente a lenguaje C el tiempo de ejecución se hubiera visto incrementado aún más, de tal modo que para alcanzar la optimización del tiempo era recomendable solamente una modificación parcial.

En el sistema en el lenguaje C, la rutina denominada `RepresentacionActual` hace la conversión de la representación *Extremo-refinamiento* a *Extremo-número de puntos*, la rutina `RepresentacionAnterior` realiza la conversión de la representación *Extremo-refinamiento* a la representación por *Tercetas*, como hemos explicado en la sección 3.5.1.

En la rutina `SegmentosPuntos y Coordenadas`, se crea la malla del problema tomando como valores de entrada la representación *Extremo -refinamiento*. En el procedimiento `NuevaRepresentación` se hace el cambio de coordenadas cartesianas a prolatas, los archivos de texto generados por estas rutinas son leídos por el sistema original, supliendo algunas partes de los procedimientos como aquel que especifica los parámetros, la rutina que especifica la malla, el procedimiento que realiza el cálculo de los términos de grado cero en el exponente de la ecuación OZ y las funciones que hacen el cálculo de los núcleos de ésta, el rediseño de éstas rutinas permite la iteración amigable al usuario, pero no le quitan efectividad en tiempo de ejecución al problema.

Las rutinas encargadas de calcular los núcleos de la ecuación OZ, junto con las rutinas

que necesitan para su ejecución, fueron rediseñadas en el programa de C para que sea posible mostrar las gráficas esperadas desde el inicio de la ejecución, con el propósito de que un usuario pueda tener una idea del comportamiento del sistema y así escoger los datos de manera consistente.

También la rutina que calcula las seis funciones que son auxiliares a las rutinas que calculan los núcleos esta rediseñada para conocer con anterioridad a la ejecución completa del sistema los resultados generados por las matrices que en ella se definen. Se toman los datos de los arreglos lineales, generados en FORTRAN para construir arreglos planares que permiten una mejor visualización de sus resultados en una gráfica tridimensional.

Las gráficas, en general, las tomamos de los valores de los datos de cada una de las rutinas involucradas, las generamos con el resultado de las operaciones realizadas en las subrutinas en diferentes momentos de su ejecución.

La iteración entre los programas realizados en FORTRAN y los creados en C se da por medio de archivos de texto, que son generados por todos y cada uno de los programas en C, el sistema en FORTRAN tiene la capacidad de leer estos archivos formateados.

De aquí en adelante es posible comparar los procesos ejecutados, facilitando el análisis de los resultados que se obtienen al utilizar el sistema, ya que por medio de una interfaz *Web* es fácil ejecutar el sistema desde alguna estación de trabajo, e iniciar varios procesos simultáneamente, si que sea necesario dedicar una computadora a ejecutar un único proceso, pero si es necesario tener un servidor dedicado a la ejecución de todos los procesos.

4.5. Esquema general del sistema que resuelve la ecuación de Ornstein-Zernike

En la figura 4.1 mostramos un esquema general del sistema que resuelve la ecuación OZ, en un diagrama a bloques, en el cual se explica el procedimiento de ejecución que sigue. A continuación damos una breve explicación de lo que representan los bloques con número.

1. Programa en PHP que se comunica con la base de datos.
2. Programa en PHP que administra las carpetas del sistema.
3. Programa en PHP que escribe los datos introducidos.
4. Programa en PHP que ejecuta los Scripts, los programas en C, el programa en FORTRAN, genera los nuevos archivos en GNUPLOT y los ejecuta.
5. Programas en C que realizan las conversiones entre las diferentes representaciones a partir de los archivos generados por PHP y los escribe en archivos de texto.
6. Programas en C que realiza los cálculos de los procedimientos `nucleo1`, `nucleo2`, `nucleo3`, `XMTX` que realizan el procedimiento de MNR para resolver la ecuación OZ y el programa `ptosinic` que se encarga de la generación de los puntos iniciales del MNR. El conjunto de estos nos sirve para las pruebas de procesamiento, antes de ejecutar el sistema en su totalidad.

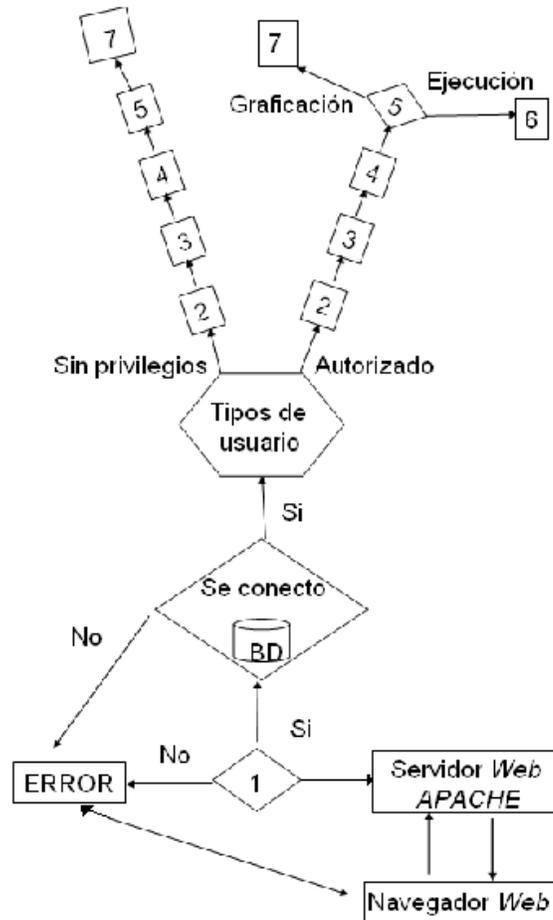


Figura 4.1: Esquema general del sistema que resuelve la ecuación OZ

7. Programa en FORTRAN que resuelve la ecuación OZ, para líquidos no homogéneos, a partir de los datos leídos de los archivos de texto generados por C y almacena sus resultados en archivos de texto para ser leídos por GNUPLOT y así graficar.

Como se ha dicho anteriormente contamos con 3 tipos de usuarios en el diagrama a bloques enfatizamos el procedimiento que se sigue para la solución de la ecuación, por esta razón no incluimos en el diagrama al Administrador del sistema. En general para el usuario Autorizado y Sin privilegios, se sigue el mismo procedimiento excepto al momento de ejecutar el programa, el usuario Autorizado tiene esta posibilidad, el usuario Sin privilegios solo puede hacer pruebas con los datos para graficar.

Capítulo 5

Diseño e implementación del sistema

El sistema que resuelve la denominada ecuación de Ornstein - Zernike, corre actualmente bajo el sistema operativo Unix en una computadora Sun, el cual, como ya ha sido mencionado padece de la debilidad de obligar al usuario a recompilarle al momento de utilizarlo, lo cual involucra una serie de desventajas anexas. Así, el tiempo que dura esta recompilación y que se relaciona directamente con el tamaño de las matrices que el sistema por sí mismo ha determinado (sin intervención del usuario), puede llegar a significar periodos de tiempo tan prolongados como podrían ser, por ejemplo, 60 días o más. Por otro lado la operación predeterminada del sistema provoca un desperdicio de espacio.

Las anteriores desventajas son importantes de por sí, pero, lo que resultaba de mayor trascendencia, dada la forma en que se ejecutan algunas partes del sistema, era resolver la posibilidad de que cada usuario cambiara los datos sin tener que compilar el sistema en todo momento, esto se alcanzo rediseñando algunas funciones en C e implementando la aceleración en el procedimiento MNR, creando una interfaz gráfica; programas de manejo de resultados y, construyendo una organización de arquitectura distribuida *cliente - servidor*. Ahora, por medio del cliente los datos se envían, y se visualizan los resultados, mientras que el servidor se encarga de la ejecución del sistema cada que hay nuevos datos, por medio de PHP, los datos quedan almacenados en una base de datos accesible a posibles consultas posteriores y se tiene también la posibilidad de realizar pruebas y análisis del comportamiento de la solución al refinar la malla.

5.1. Implementación del procesamiento numérico

Presentamos aquí algunos de los detalles de implementación.

En una situación como la bosquejada en la figura 3.1, sea τ la separación entre las partículas β y γ , y sea D el diámetro común de ellas. Al colocar el origen de coordenadas en el punto \mathbf{r}_α , la distancia de la frontera de la partícula β al origen es $\delta = \frac{\tau-D}{2}$.

Al pasar a coordenadas prolatas, el proceso de integración de una función se transforma en uno de integración en la región $D_0 = \{(\eta, \xi) \mid |\eta| \leq x_0, |\xi| \leq 1\}$, el cual puede realizarse mediante técnicas de elemento finito considerando un mallado \mathcal{M} consistente de segmentos horizontales $H_{0\delta} \cup H_{1\delta}$ y de segmentos verticales $V_{0\delta} \cup V_{1\delta}$ descritos en las secciones 3.2.1, 3.6.1. Sea n_1 el número común de segmentos verticales en $V_{0\delta}$ o $V_{1\delta}$. Sea n_2 el número de

<p><i>B</i> Constante de Boltzmann: $B = 1,38044 \times 10^{-16}$.</p> <p>$\epsilon_0$ Constante eléctrica: $\epsilon_0 = 4,80298 \times 10^{-10}$.</p> <p><i>D</i> Diámetro de partícula: $D = O(10^{-8})$.</p> <p>τ : $\tau = O(10^{-8})$. Separación entre ambas partículas que conforman el complejo α.</p> <p><i>R</i> : $R = O(10^{-8})$. Debería tenerse que $R = 1 - \frac{2\tau}{D}$ para que $r_{12} := \frac{1}{2}D(1 + R)$ coincida con τ.</p> <p>En tal caso $\frac{2r_{12}}{\tau} = \frac{1+R}{\tau}$.</p> <p>$x_0 = \frac{R+D}{\tau}$:</p> <p>$\beta = \frac{2}{B\tau}$:</p> <p>$R_0 = (r_1, r_2) \in \mathbb{R}^2$: $\ R_0\ = O(10^{20})$. $\rho = r_1 + r_2$.</p> <p>$E = \frac{1}{6}\pi\rho D^3$:</p>	<p>$\epsilon_1 = 78,5$: Una constante.</p> <p>$C = \frac{\beta\epsilon_0^2}{\epsilon_1}$:</p> <p>$c_1 = \left(\frac{1-2E}{(1-E)^2}\right)^2$:</p> <p>$c_2 = -\left(\frac{1-\frac{1}{2}E}{D(1-E)^2}\right)^2$:</p> <p>$\alpha = \frac{2\sqrt{\pi\beta}}{\epsilon_1}$:</p> <p>$x_K = \alpha\sqrt{\epsilon_0^2(r_1q_1^2 + r_2q_2^2)}$:</p> <p>$\gamma = \frac{\sqrt{1+2x_K D}-1}{2D}$:</p> <p>$XK = (c_1, 6Ec_2, -\frac{Ec_2}{2D^3})$:</p> <p>$Fun = \left(-2\frac{\gamma}{1+D\gamma}, \left(\frac{\gamma}{1+D\gamma}\right)^2\right)$:</p>
---	---

Tabla 5.1: Constantes definidas inicialmente

segmentos horizontales en $H_{0\delta}$ y n_3 el de $H_{1\delta}$. Hagamos $n_T = n_2 + n_3$.

Se ha de resolver el sistema de ecuaciones 3.15, con $I_0 = \mathcal{M}$ y el caso particular de $k = 2$.

En el archivo `datosgenerales.in` se dan los principales valores a los parámetros de entrada, entre ellos algunos valores físicos, las constantes n_1 , n_2 y n_3 y las separaciones entre segmentos paralelos en el mallado \mathcal{M} se introducen en los archivos `eta.in`, `xsi.in` y `xsi2.in` respectivamente.

Inicialmente se recibe el parámetro NK como entrada, el cual coincide con KS que es un contador del número de veces que se ha de realizar el procedimiento total. Dará como salida los índices n_1 , n_2 y n_3 .

Aquí se definen algunas constantes, presentadas en la tabla 5.1.

También se leen seis constantes:

n_1 : Número de puntos a considerar en el eje η .	n_P : Número de bloques a considerar en el eje η .
n_2 : Número de puntos a considerar en el eje ξ , con $ \xi \leq \frac{1}{2}(\tau - D)$.	n_R : Número de bloques a considerar en el eje ξ , con $ \xi \leq \frac{1}{2}(\tau - D)$.
n_3 : Número de puntos a considerar en el eje ξ , con $1 \geq \xi \geq \frac{1}{2}(\tau - D)$.	n_S : Número de bloques a considerar en el eje ξ , con $1 \geq \xi \geq \frac{1}{2}(\tau - D)$.

Para cada $i \leq n_P$ se lee una tríada (x_i, m_{i1}, m_{i2}) de manera que en el intervalo $[x_{i-1}, x_i]$ se consideran $k_i = \lceil (x_i - x_{i-1}) \frac{m_{i1}}{m_{i2}} \rceil$ puntos equidistantes. Por la manera en la que se realizan los procesos de integración, es muy importante que cada k_i sea par. Con estos valores se forma el arreglo de Extremo-Índice $N_P = [1] * [k_1, \dots, k_{n_P}]$. Obviamente, ha de tenerse $n_1 = 1 + \sum_i k_i$ y éste ha de ser impar. Se hace lo mismo para cada $i \leq n_R$, considerando n_2 , y para cada $i \leq n_S$, considerando n_3 .

Posteriormente, se determinan las particiones $\mathcal{E} = (\eta^{(j)})_{j \leq n_1}$, $\Xi_1 = (\xi^{(i,2)})_{i \leq n_2}$ y $\Xi_2 = (\xi^{(i,3)})_{i \leq n_3}$ cuyos principales parámetros fueron definidos en inicialmente.

A partir de aquí queda determinada la malla de elemento finito:

$$\mathcal{M} = \mathcal{M}_1 + \mathcal{M}_2 \quad (5.1)$$

$$\mathcal{M}_1 + \mathcal{M}_2 = \{(\eta + 1, \xi) | \eta \in \mathcal{E}, \xi \in \Xi_1\} \cup \{(\eta + \xi + (1 - \frac{1}{2}(\tau - D)), \xi) | \eta \in \mathcal{E}, \xi \in \Xi_2\} \quad (5.2)$$

\mathcal{M} se muestra gráficamente en las figuras 3.6 y 3.7 de la sección 3.2.1. \mathcal{M} queda enumerado utilizando dos índices: Para $i \leq n_2 + n_3$ y $j \leq n_1$ sea

$$\mathbf{m}_{ij} = \begin{cases} (\eta^{(j)} + 1, \xi^{(i)}) & \text{si } i \leq n_2, \\ (\eta^{(j)} + \xi^{(i)} + (1 - \frac{1}{2}(\tau - D)), \xi^{(i)}) & \text{si } n_2 < i \leq n_2 + n_3. \end{cases}$$

La escritura final son las soluciones ω_{ijk} , $k \leq 2$, para cada pareja (i, j) con $i \leq n_T$ y $j \leq n_1$. También se escribe la lista de tercetas $(x, y, \omega_0(x, y))$, ya en coordenadas cartesianas, y la lista de tercetas $(x, y, \omega_1(x, y))$, también en coordenadas cartesianas. Esta última escritura se hace aprovechando la simetría cilíndrica de las soluciones. También son escritos los puntos del cuadrante II, los puntos del cuadrante I, los puntos del cuadrante III y, finalmente, los puntos del cuadrante IV.

En la escritura final se recibe como entrada un entero KS que indica a que archivo se enviará la lista de salida.

Dado $\boldsymbol{\omega} \in \mathbb{R}^{n_t \times n_1 \times 2} \approx \mathbb{R}^{n_P}$ se calcula a las funciones $\mathbf{FHS}(\boldsymbol{\omega}) = (FHS_i)_{i \leq n_T}$, $\mathbf{SGM}(\boldsymbol{\omega}) = (SGM_{ij})_{i \leq n_T}^{j \leq n_1}$ y $\mathbf{FEL}(\boldsymbol{\omega}) = (FEL_{ij})_{i \leq n_T}^{j \leq n_1}$ utilizando el resultado que proviene de los valores $\boldsymbol{\omega}$ de entrada, para cada $i \in \llbracket 1, n_T \rrbracket$ calculando a los valores siguientes:

1.

$$FHS_i = \left(r_1 (\omega_{2(n(i-1)+1)-1} + 1) + r_2 (\omega_{2(n(i-1)+1)} + 1) \right) \cdot \\ (2(-\xi_i)^3 - 3(a)(-\xi_i)^2 - (2 - (a)^2)(-\xi_i) + a).$$

$$\text{donde } a = \frac{R + D}{\tau}$$

2. Para cada $j \leq n_1$ se hace

$$SGM_{ij} = (Q_1 r_1 \omega_{2(n(i-1)+j)-1} + Q_2 r_2 \omega_{2(n(i-1)+j)}) \tau^3 (\eta_j^2 - \xi_i^2) \\ FEL_{ij} = (Q_1 r_1 \omega_{2(n(i-1)+j)-1} + Q_2 r_2 \omega_{2(n(i-1)+j)}) (w + m) \\ \text{donde } w = \frac{(\eta_j - \xi_i)(\eta_j \xi_i + 1)}{(\eta_j + \xi_i)^2} \\ m = \frac{(\eta_j + \xi_i)(-\eta_j \xi_i + 1)}{(\eta_j - \xi_i)^2}$$

Además de observar, claro está, los valores calculados, en la terminal se escribe un listado de valores en los archivos omega

$$(\xi_i, \omega_{i11}, \omega_{i12}, FHS_i), \quad i \leq n_T.$$

Posteriormente se realiza un proceso de integración utilizando el elemento finito, con base en triángulos, descrito a continuación y que se muestra en la figura 5.1. Con los parámetros,

$$\alpha_1 = 0,0597158717 \quad \beta_1 = 0,4701420641 \\ \alpha_2 = 0,7974269853 \quad \beta_2 = 0,1012865073$$

se considera los 7 puntos siguientes, dados en coordenadas de área,

$$v_1 = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3} \right) \\ v_2 = (\alpha_1, \beta_1, \beta_1) \quad v_5 = (\alpha_2, \beta_2, \beta_2) \\ v_3 = (\beta_1, \alpha_1, \beta_1) \quad v_6 = (\beta_2, \alpha_2, \beta_2) \\ v_4 = (\beta_1, \beta_1, \alpha_1) \quad v_7 = (\beta_2, \beta_2, \alpha_2)$$

y los pesos de integración,

$$w_1 = 0,2250000000 \\ w_2 = w_3 = w_4 = 0,1323941527 \\ w_5 = w_6 = w_7 = 0,1259391805$$

Se recibe como entrada una función SGM dada como una matriz $SGM = (SGM_{ij})_{(\eta^{(j)}, \xi^{(i)}) \in \mathcal{M}}$ y se va a calcular su integral:

$$I \approx \int_{B_1} SGM.$$

El recorrido de integración se hace de 4 en 4 cuadrángulos como se muestra en la figura ??.

Inicialmente se hace $\forall i \leq n_T, j \leq n_1: I_{ij} = 0$.

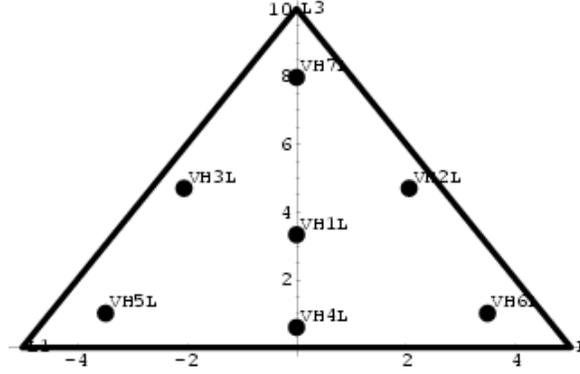


Figura 5.1: Elemento finito triangular de 7 nodos.

Después, en cada grupo de 4 cuadrángulos se considera 2 triángulos. Al definir,

$$P = 4 \cdot \text{área}(T_l) \cdot \left(\sum_{k=1}^7 w_k \right)$$

en cada uno de los seis vértices del triángulo inferior T_{inf} se hace:

$$\begin{aligned} I_{2\mu+1,2\nu-1} &= I_{2\mu+1,2\nu-1} + 0 \cdot P \cdot SGM_{2\mu+1,2\nu-1} \\ I_{2\mu,2\nu-1} &= I_{2\mu,2\nu-1} + \frac{1}{3} \cdot P \cdot SGM_{2\mu,2\nu-1} \\ I_{2\mu,2\nu} &= I_{2\mu,2\nu} + \frac{1}{3} \cdot P \cdot SGM_{2\mu,2\nu} \\ I_{2\mu-1,2\nu-1} &= I_{2\mu-1,2\nu-1} + 0 \cdot P \cdot SGM_{2\mu-1,2\nu-1} \\ I_{2\mu-1,2\nu} &= I_{2\mu-1,2\nu} + \frac{1}{3} \cdot P \cdot SGM_{2\mu-1,2\nu} \\ I_{2\mu-1,2\nu+1} &= I_{2\mu-1,2\nu+1} + 0 \cdot P \cdot SGM_{2\mu-1,2\nu+1} \end{aligned}$$

Similarmente, en cada uno de los seis vértices del triángulo superior T_{sup} se hace:

$$\begin{aligned} I_{2\mu+1,2\nu-1} &= I_{2\mu+1,2\nu-1} + 0 \cdot P \cdot SGM_{2\mu+1,2\nu-1} \\ I_{2\mu+1,2\nu} &= I_{2\mu+1,2\nu} + \frac{1}{3} \cdot P \cdot SGM_{2\mu+1,2\nu} \\ I_{2\mu+1,2\nu+1} &= I_{2\mu+1,2\nu+1} + 0 \cdot P \cdot SGM_{2\mu+1,2\nu+1} \\ I_{2\mu,2\nu} &= I_{2\mu,2\nu} + \frac{1}{3} \cdot P \cdot SGM_{2\mu,2\nu} \\ I_{2\mu,2\nu+1} &= I_{2\mu,2\nu+1} + \frac{1}{3} \cdot P \cdot SGM_{2\mu,2\nu+1} \\ I_{2\mu-1,2\nu+1} &= I_{2\mu-1,2\nu+1} + 0 \cdot P \cdot SGM_{2\mu-1,2\nu+1} \end{aligned}$$

En resumidas cuentas, sólo contribuyen los nodos aristas de los triángulos de la forma T_{inf} y T_{sup} , y cada tal nodo arista que no sea fronterizo de B_1 pertenece a dos tales triángulos.

$$\text{Se devuelve el valor } Res = 2\pi I = 2\pi \sum_{i=1}^{n_T} \sum_{j=1}^{n_1} I_{ij}.$$

Implicitamente se calcula $I(\mathbf{SGM}) = \int_{B_1} \mathbf{SGM}(\omega)$ y $I(\mathbf{FEL}) = \int_{B_1} \mathbf{FEL}(\omega)$. Para cada $i \leq n_T$ sea $RS(\xi^{(i)}) = \int_{\eta} \mathbf{SGM}(\eta, \xi^{(i)}) d\eta$. Se hace:

$$RS_2 = \int_{\xi} RS(\xi) d\xi$$

$$RS_3 = \int_{\xi} \mathbf{FHS}(\xi) d\xi$$

y entonces se define

$$f_1 = Q_3 Q_4 \frac{\epsilon_0^2}{(\epsilon_1 \tau^2)}$$

$$f_2 = \frac{\tau}{2} Q_4 \frac{\epsilon_0^2}{\epsilon_1} I(\mathbf{FEL})$$

$$f_3 = \pi \frac{\tau^2 \rho}{4\beta(j)} RS_3$$

$$f_T = f_1 + f_2 + f_3$$

Se escribe en un archivo $\frac{2\tau}{D}$, f_2 , f_3 , f_T , $I(\mathbf{SGM})$ y en otro se escribe τ , f_T . Posteriormente, se continua con la escritura de los archivos omega, para finalmente imprimir un archivo de este tipo se tienen que realizar todos los pasos descritos anteriormente.

En pasos intermedios se calcula la matriz $\mathbf{A} = F(\mathcal{M})$, la cual es tal que $\forall i, j : A_{ij}$ es un término incluido en el valor $M_{k;ij}$ dado por la ec. 3.13. Si $F : B_1 \rightarrow \mathbb{R}$ es una función, entonces por $F(\mathcal{M})$ nos referiremos a la matriz $(F(\mathbf{m}_{ij}))_{i \leq n_2 + n_3}^{j \leq n_1}$.

En particular, en esta subrutina se calcula una matriz $\mathbf{A} = F(\mathcal{M})$. Si $\frac{R+D}{\tau} \leq 1$ entonces se toma a la función F definida en el recuadro 5.2. En otro caso, se toma a la función F definida en el recuadro 5.3.

En otro caso, se hace $NI = 2$, $NP = 64$, $x_{Mn} = 0$. El valor x_{Mx} es $x_{Mx} = \cos\left(\frac{-D^2 + g_{13}\left(\frac{\tau}{2}\right)^2}{h_{13}\left(\frac{\tau}{2}\right)^2}\right)$ si $RM \geq \pi$, y si esto no ocurre es $x_{Mx} = \pi$.

Se definen los pesos \mathbf{a} y \mathbf{w} utilizando el valor de NP (por tanto $NPD2 = 32$). El procedimiento inicia en función de su argumento de entrada IPQ , define dos listas de pesos y un valor de salida $NPD2$.

Esta rutina define: a, w Si $IPQ \leq 7$ se hace $NPD2 = \lfloor \frac{IPQ}{2} \rfloor$ y se define los pesos según las relaciones en los recuadros 5.4, 5.5 o 5.6 de acuerdo con que $NPD2 = 1, 2$ o 3 .

Si $IPQ \geq 8$ se hace $NDIR = \text{Min}\{\lfloor \frac{IPQ}{8} \rfloor, 5\}$, $IPQ = \text{Min}\{IPQ, 64\}$ y $NPD2 = \lfloor \frac{IPQ}{2} \rfloor$ y se definen los pesos según las relaciones en los recuadros 5.7, 5.8, 5.9, 5.10 o 5.11 de acuerdo con el valor de $NPD2$. En otras palabras, al considerar los intervalos de enteros:

$$J_1 = \llbracket 2, 4\llbracket \quad J_2 = \llbracket 4, 6\llbracket \quad J_3 = \llbracket 6, 8\llbracket \quad J_4 = \llbracket 8, 16\llbracket$$

$$J_5 = \llbracket 16, 24\llbracket \quad J_6 = \llbracket 24, 32\llbracket \quad J_7 = \llbracket 32, 40\llbracket \quad J_8 = \llbracket 40, +\infty\llbracket$$

entonces si el valor de IPQ cae en J_i se toma el conjunto de pesos en el i -ésimo recuadro. Observamos que los vectores de pesos, para cada caso, tienen longitudes

$$\ell_1 = 1, \ell_2 = 2, \ell_3 = 3, \ell_4 = 4, \ell_5 = 8, \ell_6 = 12, \ell_7 = 16, \ell_8 = 32.$$

Sea $f(x) = 0$ si $x > \frac{R+3D}{2}$. En otro caso, hacemos

$$\begin{aligned}
 T_i &= \frac{1}{i} \left(\left(\frac{R+D}{2} \right)^i - (x-D)^2 \right) \text{ para } i = 2, \dots, 7 \\
 T_{12} &= \frac{1}{2}c_1 (D^2 - x^2) + 2Ec_2 (D^3 - x^3) - \frac{Ec_2}{10D^3} (D^5 - x^5) \\
 T_{13} &= \left(c_1x + 2Ec_2x^2 - \frac{Ec_2}{10D^3}x^4 \right) \\
 T_{14} &= \left(\frac{1}{2}c_1 + 2Ec_2x - \frac{Ec_2}{5D^3}x^3 \right) \\
 T_{15} &= \frac{1}{3}2Ec_2 - \frac{Ec_2}{5D^3}x^2 \\
 T_{16} &= -\frac{Ec_2}{10D^3}x \\
 T_{17} &= -\frac{Ec_2}{50D^3} \\
 f(x) &= 2\frac{\pi\rho}{x}(T_2T_{12} + T_3T_{13} - T_4T_{14} + T_5T_{15} - T_6T_{16} + T_7T_{17})
 \end{aligned}$$

Entonces

$$F : (\eta, \xi) \mapsto F(\eta, \xi) = f\left(\frac{\tau}{2}(\eta - \xi)\right) + f\left(\frac{\tau}{2}(\eta + \xi)\right)$$

Tabla 5.2: Una primera función F para esta sección.

Si $\frac{R+D}{\tau} > 1$ entonces se hace

$$\xi_{Mn} = \begin{cases} 1 - \frac{R+D}{\tau} & \text{si } \frac{R+D}{\tau} \leq 1, \\ 0 & \text{en otro caso.} \end{cases} \quad \text{y } \xi_{Mx} = 1.$$

Se considera las funciones

- $v_1 : (\eta, \xi; \eta_1, \xi_1) \mapsto XK^+(\eta, \xi; \eta_1, \xi_1)$, según se calcula en núcleo1, y
- $v_2 : (\eta, \xi; \eta_1, \xi_1) \mapsto XK^+(\eta, \xi; \eta_1, \xi_1)$, según se calcula en núcleo2.

Para cada

$$(\eta, \xi) \in B_1 \text{ sea } F(\eta, \xi) = \int_{\xi_{Mn}}^{\xi_{Mx}} d\xi_1 \int_1^{\xi + \frac{R+D}{\tau} - 1} (v_1 + v_2)(\eta, \xi; \eta_1, \xi_1) d\eta_1.$$

El cálculo de F se hace mediante el Teorema de Fubini y la regla de Simpson, , con 150 puntos en cada utilización de la regla.

Tabla 5.3: Una segunda función F para esta sección.

Si $\frac{R+D}{\tau} > 1$ entonces se hace

$$\xi_{Mn} = \begin{cases} 1 - \frac{R+D}{\tau} & \text{si } \frac{R+D}{\tau} \leq 1, \\ 0 & \text{en otro caso.} \end{cases} \quad \text{y } \xi_{Mx} = 1. \text{ Se consideran las funciones}$$

- $v_1 : (\eta, \xi; \eta_1, \xi_1) \mapsto XK^+(\eta, \xi; \eta_1, \xi_1),$
- $v_2 : (\eta, \xi; \eta_1, \xi_1) \mapsto XK^+(\eta, \xi; \eta_1, \xi_1),$

Tomando en cuenta que, para dos puntos $(\eta_1, \xi_1), (\eta_3, \xi_3) \in B_1$ se calcula en cada una de estas rutinas dos valores $XK^+(\eta_1, \xi_1; \eta_3, \xi_3)$ y $XL^+(\eta_1, \xi_1; \eta_3, \xi_3)$ procediendo según el recuadro 5.12.

Se calcula a los parámetros

$$\begin{aligned} g_{13} &= g_{13}(\eta_1, \xi_1; \eta_3, \xi_3) & RN &= \frac{\tau}{2} \sqrt{g_{13} - h_{13}} \\ h_{13} &= 2\sqrt{(1 - \xi_1^2)(1 - \xi_3^2)(\eta_1^2 - 1)(\eta_3^2 - 1)} & RM &= \frac{\tau}{2} \sqrt{g_{13} + h_{13}} \end{aligned}$$

Si $RN \geq D$ entonces

$$XK^+ = 0 \quad \text{y} \quad XL^+ = 0$$

$$a_1 = 0,57735027 \quad w_1 = 1,00000000$$

Tabla 5.4: Pesos cuando $IPQ \in J_1$.

$$\begin{aligned} a_1 &= 0,33998104 & w_1 &= 0,65214515 \\ a_2 &= 0,86113631 & w_2 &= 0,34785484 \end{aligned}$$

Tabla 5.5: Pesos cuando $IPQ \in J_2$.

$$\begin{aligned} a_1 &= 0,23861919 & w_1 &= 0,46791393 \\ a_2 &= 0,66120939 & w_2 &= 0,36076157 \\ a_3 &= 0,93246951 & w_3 &= 0,17132449 \end{aligned}$$

Tabla 5.6: Pesos cuando $IPQ \in J_3$.

$$\begin{aligned} a_1 &= 0,1834346424 & w_1 &= 0,3626837834 \\ a_2 &= 0,5255324099 & w_2 &= 0,3137066459 \\ a_3 &= 0,7966664774 & w_3 &= 0,2223810344 \\ a_4 &= 0,9602898565 & w_4 &= 0,1012285362 \end{aligned}$$

Tabla 5.7: Pesos cuando $IPQ \in J_4$.

$a_1 = 9,5012509838 \times 10^{-02}$	$w_1 = 1,8945061046 \times 10^{-01}$
$a_2 = 2,8160355078 \times 10^{-01}$	$w_2 = 1,8260341504 \times 10^{-01}$
$a_3 = 4,5801677766 \times 10^{-01}$	$w_3 = 1,6915651940 \times 10^{-01}$
$a_4 = 6,1787624440 \times 10^{-01}$	$w_4 = 1,4959598882 \times 10^{-01}$
$a_5 = 7,5540440836 \times 10^{-01}$	$w_5 = 1,2462897126 \times 10^{-01}$
$a_6 = 8,6563120239 \times 10^{-01}$	$w_6 = 9,5158511682 \times 10^{-02}$
$a_7 = 9,4457502307 \times 10^{-01}$	$w_7 = 6,2253523939 \times 10^{-02}$
$a_8 = 9,8940093499 \times 10^{-01}$	$w_8 = 2,7152459412 \times 10^{-02}$

Tabla 5.8: Pesos cuando $IPQ \in J_5$.

$a_1 = 6,4056892863 \times 10^{-02}$	$w_1 = 1,2793819535 \times 10^{-01}$
$a_2 = 1,9111886747 \times 10^{-01}$	$w_2 = 1,2583745635 \times 10^{-01}$
$a_3 = 3,1504267970 \times 10^{-01}$	$w_3 = 1,2167047293 \times 10^{-01}$
$a_4 = 4,3379350763 \times 10^{-01}$	$w_4 = 1,1550566805 \times 10^{-01}$
$a_5 = 5,4542147139 \times 10^{-01}$	$w_5 = 1,0744427012 \times 10^{-01}$
$a_6 = 6,4809365194 \times 10^{-01}$	$w_6 = 9,7618652104 \times 10^{-02}$
$a_7 = 7,4012419158 \times 10^{-01}$	$w_7 = 8,6190161532 \times 10^{-02}$
$a_8 = 8,2000198597 \times 10^{-01}$	$w_8 = 7,3346481411 \times 10^{-02}$
$a_9 = 8,8641552700 \times 10^{-01}$	$w_9 = 5,9298584915 \times 10^{-02}$
$a_9 = 8,8641552700 \times 10^{-01}$	$w_{10} = 4,4277438817 \times 10^{-02}$
$a_{10} = 9,3827455200 \times 10^{-01}$	$w_{11} = 2,8531388629 \times 10^{-02}$
$a_{11} = 9,7472855597 \times 10^{-01}$	$w_{12} = 1,2341229800 \times 10^{-02}$
$a_{12} = 9,9518722000 \times 10^{-01}$	

Tabla 5.9: Pesos cuando $IPQ \in J_6$.

$a_1 = 4,8307665688 \times 10^{-02}$	$w_1 = 9,6540088515 \times 10^{-02}$
$a_2 = 1,4447196158 \times 10^{-01}$	$w_2 = 9,5638720079 \times 10^{-02}$
$a_3 = 2,3928736225 \times 10^{-01}$	$w_3 = 9,3844399081 \times 10^{-02}$
$a_4 = 3,3186860228 \times 10^{-01}$	$w_4 = 9,1173878696 \times 10^{-02}$
$a_5 = 4,2135127613 \times 10^{-01}$	$w_5 = 8,7652093004 \times 10^{-02}$
$a_6 = 5,0689990893 \times 10^{-01}$	$w_6 = 8,3311924227 \times 10^{-02}$
$a_7 = 5,8771575724 \times 10^{-01}$	$w_7 = 7,8193895787 \times 10^{-02}$
$a_8 = 6,6304426693 \times 10^{-01}$	$w_8 = 7,2345794109 \times 10^{-02}$
$a_9 = 7,3218211874 \times 10^{-01}$	$w_9 = 6,5822222776 \times 10^{-02}$
$a_{10} = 7,9448379597 \times 10^{-01}$	$w_{10} = 5,8684093479 \times 10^{-02}$
$a_{11} = 8,4936761373 \times 10^{-01}$	$w_{11} = 5,0998059262 \times 10^{-02}$
$a_{12} = 8,9632115577 \times 10^{-01}$	$w_{12} = 4,2835898022 \times 10^{-02}$
$a_{13} = 9,3490607594 \times 10^{-01}$	$w_{13} = 3,4273862913 \times 10^{-02}$
$a_{14} = 9,6476225559 \times 10^{-01}$	$w_{14} = 2,5392065309 \times 10^{-02}$
$a_{15} = 9,8561151155 \times 10^{-01}$	$w_{15} = 1,6274394731 \times 10^{-02}$
$a_{16} = 9,9726386185 \times 10^{-01}$	$w_{16} = 7,0186100095 \times 10^{-03}$

Tabla 5.10: Pesos cuando $IPQ \in J_7$.

Se calculan luego todos los pesos de cada uno de los vectores wt , xp correspondientes a dos divisiones de $[x_{Mn}, x_{Mx}]$, con NP nodos en cada división. Para esto, recibe 4 argumentos de entrada: $n = NSUB$, $I = IPQ$, $NPD2 = \frac{I}{2}$, $x_{Mx} = XMAX$ y $x_{Mn} = XMIN$. n es el número de subdivisiones del intervalo $[x_{Mn}, x_{Mx}]$ e I es el número de nodos en cada subdivisión.

Se hace $\delta = \frac{1}{n}(x_{Mx} - x_{Mn})$, $c = \frac{\delta}{2}$ y para cada $i \leq n$, $x_i = x_{Mn} + i\delta$.

Para cada $i \in \llbracket 1, n \rrbracket$ se procede como sigue: Sean $t_i = \frac{1}{2}(x_i + x_{i-1})$ el punto medio del intervalo real $[x_{i-1}, x_i]$ y $K_i = I(i - \frac{1}{2})$ el punto medio del i -ésimo intervalo entero consistente de I enteros consecutivos. Entonces, para $j \leq \frac{I}{2}$ se define

$$\begin{aligned} wt_{K_i+j} &= t_i + ca_j & xp_{K_i+j} &= cr_j \\ wt_{K_i+1-j} &= t_i - ca_j & xp_{K_i+1-j} &= cr_j \end{aligned} \quad (5.3)$$

donde los arreglos \mathbf{a} y \mathbf{r} , en \mathbb{R}^I , están dados previamente. Así pues, los arreglos de salida \mathbf{wt} y \mathbf{xp} son de dimensión $n \cdot I$.

Posteriormente se hace $S = \left(s_\nu = \frac{\tau}{2} \sqrt{g_{13} - h_{13} \cos(xp_\nu)} \right)_{\nu \leq 128}$, para finalmente hacer el cálculo de los pesos.

Posteriormente se considera

$$g_{13} = g_{13}^+ = (\eta_1^2 - 1)(1 - \xi_1^2) + (\eta_3^2 - 1)(1 - \xi_3^2) + (\xi_1\eta_1 - \xi_3\eta_3)^2$$

en tanto que en un segundo procedimiento se considera

$$g_{13} = g_{13}^- = (\eta_1^2 - 1)(1 - \xi_1^2) + (\eta_3^2 - 1)(1 - \xi_3^2) + (\xi_1\eta_1 + \xi_3\eta_3)^2.$$

Para cada

$(\eta, \xi) \in B_1$ sea $F(\eta, \xi) = \int_{\xi_{Mn}}^{\xi_{Mx}} d\xi_1 \int_1^{\xi + \frac{R+D}{\tau} - 1} (v_1 + v_2)(\eta, \xi; \eta_1, \xi_1) d\eta_1$. El calculo de F se

$a_1 = 2,4350292663 \times 10^{-02}$	$w_1 = 4,8690957009 \times 10^{-02}$
$a_2 = 7,2993121788 \times 10^{-02}$	$w_2 = 4,8575467442 \times 10^{-02}$
$a_4 = 1,6964442042 \times 10^{-01}$	$w_3 = 4,8344762235 \times 10^{-02}$
$a_3 = 1,2146281930 \times 10^{-01}$	$w_4 = 4,7999388596 \times 10^{-02}$
$a_5 = 2,1742364374 \times 10^{-01}$	$w_5 = 4,7540165715 \times 10^{-02}$
$a_6 = 2,6468716221 \times 10^{-01}$	$w_6 = 4,6968182816 \times 10^{-02}$
$a_7 = 3,1132287199 \times 10^{-01}$	$w_7 = 4,6284796581 \times 10^{-02}$
$a_8 = 3,5722015834 \times 10^{-01}$	$w_8 = 4,5491627927 \times 10^{-02}$
$a_9 = 4,0227015796 \times 10^{-01}$	$w_9 = 4,4590558164 \times 10^{-02}$
$a_{10} = 4,4636601725 \times 10^{-01}$	$w_{10} = 4,3583724529 \times 10^{-02}$
$a_{11} = 4,8940314571 \times 10^{-01}$	$w_{11} = 4,2473515124 \times 10^{-02}$
$a_{12} = 5,3127946402 \times 10^{-01}$	$w_{12} = 4,1262563243 \times 10^{-02}$
$a_{13} = 5,7189564620 \times 10^{-01}$	$w_{13} = 3,9953741133 \times 10^{-02}$
$a_{14} = 6,1115535517 \times 10^{-01}$	$w_{14} = 3,8550153179 \times 10^{-02}$
$a_{14} = 6,1115535517 \times 10^{-01}$	$w_{15} = 3,7055128540 \times 10^{-02}$
$a_{15} = 6,4896547125 \times 10^{-01}$	$w_{16} = 3,5472213257 \times 10^{-02}$
$a_{16} = 6,8523631305 \times 10^{-01}$	$w_{17} = 3,3805161837 \times 10^{-02}$
$a_{17} = 7,1988185017 \times 10^{-01}$	$w_{18} = 3,2057928355 \times 10^{-02}$
$a_{18} = 7,5281990726 \times 10^{-01}$	$w_{19} = 3,0234657072 \times 10^{-02}$
$a_{19} = 7,8397235894 \times 10^{-01}$	$w_{20} = 2,8339672614 \times 10^{-02}$
$a_{20} = 8,1326531512 \times 10^{-01}$	$w_{21} = 2,6377469715 \times 10^{-02}$
$a_{21} = 8,4062929625 \times 10^{-01}$	$w_{22} = 2,4352702569 \times 10^{-02}$
$a_{22} = 8,6599939815 \times 10^{-01}$	$w_{23} = 2,2270173808 \times 10^{-02}$
$a_{23} = 8,8931544600 \times 10^{-01}$	$w_{24} = 2,0134823154 \times 10^{-02}$
$a_{24} = 9,1052213708 \times 10^{-01}$	$w_{25} = 1,7951715776 \times 10^{-02}$
$a_{25} = 9,2956917213 \times 10^{-01}$	$w_{26} = 1,5726030476 \times 10^{-02}$
$a_{26} = 9,4641137486 \times 10^{-01}$	$w_{27} = 1,3463047897 \times 10^{-02}$
$a_{27} = 9,6100879965 \times 10^{-01}$	$w_{28} = 1,1168139460 \times 10^{-02}$
$a_{28} = 9,7332682779 \times 10^{-01}$	$w_{29} = 8,8467598264 \times 10^{-03}$
$a_{29} = 9,8333625388 \times 10^{-01}$	$w_{30} = 6,5044579690 \times 10^{-03}$
$a_{30} = 9,9101337148 \times 10^{-01}$	$w_{31} = 4,1470332606 \times 10^{-03}$
$a_{31} = 9,9634011677 \times 10^{-01}$	$w_{32} = 1,7832807217 \times 10^{-03}$
$a_{32} = 9,9930504174 \times 10^{-01}$	

Tabla 5.11: Pesos cuando $IPQ \in J_8$.

$$\begin{aligned}
 XK^+ &= -2(\eta_3^2 - \xi_3^2) \left(\frac{\tau}{2}\right)^3 \sum_{\nu=1}^{128} wt_\nu \left(c_1 + 6Ec_2s_\nu - \frac{Ec_2}{2D^3}s_\nu^3 \right) \\
 XL^+ &= 2(\eta_3^2 - \xi_3^2) \left(\frac{\tau}{2}\right)^3 \sum_{\nu=1}^{128} wt_\nu \left(s_\nu^{-1} - 2\frac{\gamma}{1+D\gamma} + \left(\frac{\gamma}{1+D\gamma}\right)^2 s_\nu \right)
 \end{aligned}$$

Tabla 5.12: Funciones XK^+ y XL^+ .

hace mediante el Teorema de Fubini y la regla de Simpson, con 150 puntos en cada utilización de la regla. Se procede dando un arreglo de valores $\mathbf{v} = [v_i]_i$, correspondientes a valores de una función f , en puntos igualmente espaciados: $v_i = f(x_{min} + (i-1)h)$, y dos índices i_0, i_1 tales que $1 \leq i_0 < i_1 \leq \dim(\mathbf{v})$ se hace

$$\epsilon = \begin{cases} \frac{h}{2}(-v_{i_1-2} + v_{i_1}) & \text{si } i_1 - i_0 \text{ es impar,} \\ 0 & \text{en otro caso.} \end{cases}$$

para calcular como resultado

$$r = \left[\frac{h}{3} \sum_{i=1}^{\lfloor \frac{i_1-i_0}{2} \rfloor} [v_{i_0+2i-2} + 4v_{i_0+2i-1} + v_{i_0+2i}] + \epsilon \right] \approx \int_{x_{min}+(i_0-1)h}^{x_{min}+(i_1-1)h} f.$$

r es pues una aproximación a la integral, utilizando la *regla de Simpson*, de la función con valores \mathbf{v} en el intervalo especificado.

Las integrales de los productos de funciones básicas de los núcleos K_ℓ , que aparecen en la ec. 3.14 e implícitamente en la ec. 3.15, son calculadas de la siguiente forma: Para cada punto $(\eta_1, \xi_1) \in \mathcal{M}$ se define aquí a 6 matrices

$$\begin{aligned}
 \mathbf{XF1}(\eta_1, \xi_1) &= (\mathbf{XF1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XF4}(\eta_1, \xi_1) = (\mathbf{XF4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} \\
 \mathbf{XYK1}(\eta_1, \xi_1) &= (\mathbf{XYK1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XYK4}(\eta_1, \xi_1) = (\mathbf{XYK4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} \\
 \mathbf{XYL1}(\eta_1, \xi_1) &= (\mathbf{XYL1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XYL4}(\eta_1, \xi_1) = (\mathbf{XYL4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}}
 \end{aligned}$$

Describiremos el proceso de cálculo considerando de manera genérica a una matriz

$$\mathbf{XF}(\eta_1, \xi_1) = (F(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}}$$

que puede ser cualquiera de las 6 calculadas aquí.

Consideremos la triangulación \mathcal{T} en la región B_1 cuyos nodos son los puntos en \mathcal{M} . \mathcal{T} consiste de los triángulos obtenidos al dividir mediante la diagonal (Norte-Oeste)-(Sur-Este) a cada cuadrángulo

$$C_{ij} = [\eta^{(j)}, \eta^{(j+1)}] \times [\xi^{(i)}, \xi^{(i+1)}], \quad i < n_2 + n_3, \quad j < n_1.$$

Cada punto $\mathbf{m}_{i_3j_3} = (\eta^{(j_3)}, \xi^{(i_3)}) \in \mathcal{M}$ que no sea fronterizo es un vértice de 6 triángulos en \mathcal{T} :

$$\begin{aligned}
 T_1(\mathbf{m}_{i_3j_3}) &= \triangle(\mathbf{m}_{i_3j_3}, \mathbf{m}_{i_3,j_3+1}, \mathbf{m}_{i_3+1,j_3}) & ; & \quad T_2(\mathbf{m}_{i_3j_3}) = \triangle(\mathbf{m}_{i_3j_3}, \mathbf{m}_{i_3+1,j_3}, \mathbf{m}_{i_3+1,j_3-1}) \\
 T_3(\mathbf{m}_{i_3j_3}) &= \triangle(\mathbf{m}_{i_3j_3}, \mathbf{m}_{i_3+1,j_3-1}, \mathbf{m}_{i_3,j_3-1}) & ; & \quad T_4(\mathbf{m}_{i_3j_3}) = \triangle(\mathbf{m}_{i_3j_3}, \mathbf{m}_{i_3,j_3-1}, \mathbf{m}_{i_3-1,j_3}) \\
 T_5(\mathbf{m}_{i_3j_3}) &= \triangle(\mathbf{m}_{i_3j_3}, \mathbf{m}_{i_3-1,j_3}, \mathbf{m}_{i_3-1,j_3+1}) & ; & \quad T_6(\mathbf{m}_{i_3j_3}) = \triangle(\mathbf{m}_{i_3j_3}, \mathbf{m}_{i_3-1,j_3+1}, \mathbf{m}_{i_3,j_3+1})
 \end{aligned}$$

Sea $D(\mathbf{m}_{i_3j_3}) = \bigcup_{\iota=1}^6 T_\iota(\mathbf{m}_{i_3j_3})$ la unión de los seis triángulos que contienen al punto $\mathbf{m}_{i_3j_3}$.

Se procede de manera similar a la de calcular integrales de la forma

$$F(\eta_1, \xi_1; \mathbf{m}_{i_3j_3}) \approx \int_{D(\mathbf{m}_{i_3j_3})} h(\eta_1, \xi_1; \eta, \xi) d\eta d\xi = \sum_{\iota=1}^6 \int_{T_\iota(\mathbf{m}_{i_3j_3})} h(\eta_1, \xi_1; \eta, \xi) d\eta d\xi \quad (5.4)$$

en donde cada una de las integrales $\int_{T_\iota(\mathbf{m}_{i_3j_3})} h(\eta_1, \xi_1; \eta, \xi) d\eta d\xi$ se calcula mediante el elemento finito cuyo conjunto $\phi_\iota(V_{(13)})$ de nodos consta de 13 puntos, y es una imagen isomorfa del conjunto $V_{(13)} = (\mathbf{v}_\ell)_{\ell=1}^{13}$, con pesos de integración

$$\begin{aligned} w_1^{(13)} &= -0,149570044467670 \\ w_2^{(13)} &= 0,175615257433204 & w_3^{(13)} &= 0,175615257433204 & w_4^{(13)} &= 0,175615257433204 \\ w_5^{(13)} &= 0,053347235608839 & w_6^{(13)} &= 0,053347235608839 & w_7^{(13)} &= 0,053347235608839 \\ w_8^{(13)} &= 0,077113760890257 & w_9^{(13)} &= 0,077113760890257 & w_{10}^{(13)} &= 0,077113760890257 \\ w_{11}^{(13)} &= 0,077113760890257 & w_{12}^{(13)} &= 0,077113760890257 & w_{13}^{(13)} &= 0,077113760890257 \end{aligned} \quad (5.5)$$

(obsérvese que estos pesos coinciden por renglones). Tendríamos pues

$$F(\eta_1, \xi_1; \mathbf{m}_{i_3j_3}) = \sum_{\iota=1}^6 \text{área}(T_\iota) \sum_{\ell=1}^{13} w_\ell^{(13)} h(\eta_1, \xi_1; \phi_\iota(\mathbf{v}_\ell)). \quad (5.6)$$

Sin embargo, se procede de otra manera. Se hace un recorrido por grupos de 4 cuadrángulos de la forma C_{ij} . De hecho, se supone que n_1 y $n_T = n_2 + n_3$ son impares. Entonces en cada una de las coordenadas η - ξ hay un número par de intervalos. $\forall \mu \leq \frac{n_1-1}{2}$, $\nu \leq \frac{n_2+n_3-1}{2}$ sea

$$\begin{aligned} R_{\mu,\nu} &= C_{2\mu-1,2\nu-1} \cup C_{2\mu-1,2\nu} \cup C_{2\mu,2\nu-1} \cup C_{2\mu,2\nu} \\ &= [\eta^{(2\nu-1)}, \eta^{(2\nu)}] \times [\xi^{(2\mu-1)}, \xi^{(2\mu)}] \cup [\eta^{(2\nu)}, \eta^{(2\nu+1)}] \times [\xi^{(2\mu-1)}, \xi^{(2\mu)}] \cup \\ &= [\eta^{(2\nu-1)}, \eta^{(2\nu)}] \times [\xi^{(2\mu)}, \xi^{(2\mu+1)}] \cup [\eta^{(2\nu)}, \eta^{(2\nu+1)}] \times [\xi^{(2\mu)}, \xi^{(2\mu+1)}] \end{aligned}$$

el dominio obtenido como unión de 4 cuadrángulos. La triangulación arriba descrita induce en ese dominio una subtriangulación más y podemos separar los triángulos resultantes en dos subcolecciones.

Las sumatorias en 5.6 pueden calcularse por separado en cada uno de los triángulos del dominio de la subtriangulación.

De acuerdo con la ec. 3.14, la función que integra h es el producto de una función básica, en el elemento finito, por un núcleo.

Dado un punto $(\eta, \xi) \in B_1$ existen $\mu \leq \frac{n_1-1}{2}$, $\nu \leq \frac{n_2+n_3-1}{2}$ tales que $(\eta, \xi) \in R_{\mu,\nu}$ y ahí se tiene que

- $(\eta, \xi) \in T_{inf} = \Delta(\mathbf{m}_{2\mu-1,2\nu-1}, \mathbf{m}_{2\mu+1,2\nu-1}, \mathbf{m}_{2\mu-1,2\nu+1})$ o bien
- $(\eta, \xi) \in T_{sup} = \Delta(\mathbf{m}_{2\mu+1,2\nu-1}, \mathbf{m}_{2\mu+1,2\nu+1}, \mathbf{m}_{2\mu-1,2\nu+1})$.

T_{inf}		T_{sup}	
\mathbf{m}	$f_{\mathbf{m}}$	\mathbf{m}	$f_{\mathbf{m}}$
1	$f_1 = G_{21} = L_1(2L_1 - 1)$	7	$f_7 = G_{21} = L_1(2L_1 - 1)$
2	$f_2 = G_{24} = 4L_1L_2$	8	$f_8 = G_{26} = 4L_3L_1$
3	$f_3 = G_{22} = L_2(2L_2 - 1)$	9	$f_9 = G_{24} = 4L_1L_2$
4	$f_4 = G_{26} = 4L_3L_1$	10	$f_{10} = G_{23} = L_3(2L_3 - 1)$
5	$f_5 = G_{25} = 4L_2L_3$	11	$f_{11} = G_{25} = 4L_2L_3$
6	$f_6 = G_{23} = L_3(2L_3 - 1)$	12	$f_{12} = G_{22} = L_2(2L_2 - 1)$

Tabla 5.13: Funciones asociadas a vértices en triángulos.

Los vértices del triángulo en el que está (η, ξ) son

$$(\mathbf{L}_I, \mathbf{L}_J, \mathbf{L}_K) = \begin{cases} (\mathbf{m}_{2\mu-1, 2\nu-1}, \mathbf{m}_{2\mu+1, 2\nu-1}, \mathbf{m}_{2\mu-1, 2\nu+1}) & \text{si } (\eta, \xi) \in T_{inf}, \\ (\mathbf{m}_{2\mu+1, 2\nu-1}, \mathbf{m}_{2\mu+1, 2\nu+1}, \mathbf{m}_{2\mu-1, 2\nu+1}) & \text{si } (\eta, \xi) \in T_{sup}. \end{cases}$$

Sean L_I, L_J, L_K las correspondientes funciones coordenadas de área. Estas son funciones afines tales que $L_M(\mathbf{L}_N) = \delta_{MN}$. Consideremos las seis funciones básicas, que dan una interpolación cuadrática. De acuerdo con su posición relativa en los triángulos T_{inf} o T_{sup} , donde a cada punto \mathbf{m} se le asocia una función $f_{\mathbf{m}}$ según se ve en el recuadro 5.13. La ec. 5.6 se escribe entonces como

$$F(\eta_1, \xi_1; \mathbf{m}_{i_3j_3}) = \sum_{\iota=1}^6 \text{área}(T_\iota) \sum_{\ell=1}^{13} w_\ell^{(13)} f_{\mathbf{m}}(\phi_\ell(\mathbf{v}_\ell)) g(\eta_1, \xi_1; \phi_\ell(\mathbf{v}_\ell)). \quad (5.7)$$

Las funciones g , para cada una de las seis matrices definidas aquí, están construídas como sigue:

XF1: $g_{\mathbf{XF1}}(\eta_1, \xi_1; \eta, \xi) = XF^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en `nucleo1`

XYK1: $g_{\mathbf{XYK1}}(\eta_1, \xi_1; \eta, \xi) = XK^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en `nucleo2`.

XYL1: $g_{\mathbf{XYL1}}(\eta_1, \xi_1; \eta, \xi) = XL^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en `nucleo3`.

XF4: $g_{\mathbf{XF4}}(\eta_1, \xi_1; \eta, \xi) = XF^-(\eta_1, \xi_1; \eta, \xi)$, según se calcula en `nucleo1`.

XYK4: $g_{\mathbf{XYK4}}(\eta_1, \xi_1; \eta, \xi) = XK^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en `nucleo2`.

XYL4: $g_{\mathbf{XYL4}}(\eta_1, \xi_1; \eta, \xi) = XL^+(\eta_1, \xi_1; \eta, \xi)$, según se calcula en `nucleo3`.

Ni la matriz \mathbf{A} , calculada, ni las integrales de núcleos, calculados, dependen del arreglo a calcular $(\omega_{ijk})_{i \leq n_T, j \leq n_1, k \in [0,1]}$.

Del archivo `ptosiniciales.out`, se leen a dos arreglos $\omega^+, \omega^- \in \mathbb{R}^{n_T \times n_1}$, los cuales dan valores iniciales:

$$\begin{aligned} \omega_0 &= (\omega^+ , \omega^-) \\ \omega_1 &= (\omega^- , \omega^-) \end{aligned}$$

Así pues, $\boldsymbol{\omega}_0, \boldsymbol{\omega}_1 \in \mathbb{R}^{n_P}$, donde $n_P = 2n_T n_1$ (son propiamente matrices cúbicas, en $\mathbb{R}^{n_T \times n_1 \times 2}$, donde los índices más a la derecha varían más rápido).

Sea $\boldsymbol{\omega} \mapsto \mathbf{FK}(\boldsymbol{\omega}) \in \mathbb{R}^{n_P}$ la función cuyo valor en cada $\boldsymbol{\omega} \in \mathbb{R}^{n_P}$ está dado como la exponencial que aparece en el lado derecho de la ec. 3.15. Se ha de calcular una solución de la ecuación $\boldsymbol{\omega} + \mathbf{1} = \mathbf{FK}(\boldsymbol{\omega})$. Para esto se procederá de acuerdo con el método de Newton-Raphson. Es decir, se introducen aquí dos arreglos iniciales. Para n_1 y $n_T = n_2 + n_3$ se lee los arreglos

$$\begin{aligned} \boldsymbol{\omega}_0^+ &\in \mathbb{R}^{n_t \times n_1} & : & \text{arreglo inicial} \\ \boldsymbol{\omega}_0^- &\in \mathbb{R}^{n_t \times n_1} & : & \text{otro arreglo inicial} \end{aligned}$$

cada uno de los cuales puede ser visto como un arreglo indicado por \mathcal{M} .

Se les almacena como arreglos lineales que son módulo de la función de enumeración $\sigma_1 : (i, j) \mapsto (i - 1)n_1 + j$.

Habiéndolos leído, se les compagina para formar dos nuevos arreglos:

$$\begin{aligned} \boldsymbol{\omega} &\in \mathbb{R}^{n_t \times n_1 \times 2} & : & \text{arreglo obtenido al compaginar consecutiva-} \\ & & & \text{mente los valores de } \boldsymbol{\omega}_0^+ \text{ y } \boldsymbol{\omega}_0^-, \\ \boldsymbol{\omega}_1 &\in \mathbb{R}^{n_t \times n_1 \times 2} & : & \text{arreglo obtenido al repetir consecutivamente} \\ & & & \text{los valores de } \boldsymbol{\omega}_0^-, \end{aligned}$$

cada uno de los cuales se almacena como un arreglo lineal que es módulo de la función de enumeración $\sigma_2 : (i, j, k) \mapsto 2(\sigma_1(i, j) - 1) + k$.

Posteriormente, para una $\boldsymbol{\omega} \in \mathbb{R}^{n_P}$ actual, se calcula $\boldsymbol{\omega}' = \mathbf{FK}(\boldsymbol{\omega})$ y el error, considerado éste como una distancia entre $\boldsymbol{\omega}'$ y $\boldsymbol{\omega} + \mathbf{1}$. Para el punto inicial, se tiene

$$\text{ERROR} = \sum_{i=1}^{n_P} \left[\frac{\omega'_i - (\omega_i + 1)}{(\omega_i + 1)} \right]^2$$

y en lo sucesivo

$$\text{ERROR} = \|\boldsymbol{\omega}' - (\boldsymbol{\omega} + \mathbf{1})\|_\infty = \text{Max}_{1 \leq i \leq n_P} |\omega'_i - (\omega_i + 1)|.$$

De forma que recibe como argumento de entrada un valor MM que indica una situación de inicio o no.

Habiéndose ya definido, los arreglos

$$\begin{aligned} \mathbf{XF1}(\eta_1, \xi_1) &= (\mathbf{XF1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XF4}(\eta_1, \xi_1) = (\mathbf{XF4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} \\ \mathbf{XYK1}(\eta_1, \xi_1) &= (\mathbf{XYK1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XYK4}(\eta_1, \xi_1) = (\mathbf{XYK4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} \\ \mathbf{XYL1}(\eta_1, \xi_1) &= (\mathbf{XYL1}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} & ; & \quad \mathbf{XYL4}(\eta_1, \xi_1) = (\mathbf{XYL4}(\eta_1, \xi_1; \eta_3, \xi_3))_{(\eta_3, \xi_3) \in \mathcal{M}} \end{aligned}$$

con el arreglo actual $\boldsymbol{\omega} \in \mathbb{R}^{n_t \times n_1 \times 2}$ se procede como sigue: Sean $\forall i \leq n_t, j \leq n_1$

$$\begin{aligned} S_{0;ij} &= \sum_{\mu \leq n_t} \sum_{\nu \leq n_1} (\mathbf{XF1}(\eta^{(j)}, \xi^{(i)}; \eta^{(\nu)}, \xi^{(\mu)}) + \mathbf{XF4}(\eta^{(j)}, \xi^{(i)}; \eta^{(\nu)}, \xi^{(\mu)})) [q_1 r_1 \omega_{\mu\nu 1} + q_2 r_2 \omega_{\mu\nu 2}] \\ S_{1;ij} &= \sum_{\mu \leq n_t} \sum_{\nu \leq n_1} (\mathbf{XL1}(\eta^{(j)}, \xi^{(i)}; \eta^{(\nu)}, \xi^{(\mu)}) + \mathbf{XL4}(\eta^{(j)}, \xi^{(i)}; \eta^{(\nu)}, \xi^{(\mu)})) [q_1 r_1 \omega_{\mu\nu 1} + q_2 r_2 \omega_{\mu\nu 2}] \\ S_{2;ij} &= \sum_{\mu \leq n_t} \sum_{\nu \leq n_1} (\mathbf{XK1}(\eta^{(j)}, \xi^{(i)}; \eta^{(\nu)}, \xi^{(\mu)}) + \mathbf{XK4}(\eta^{(j)}, \xi^{(i)}; \eta^{(\nu)}, \xi^{(\mu)})) [r_1 \omega_{\mu\nu 1} + r_2 \omega_{\mu\nu 2}]. \end{aligned}$$

Con estos valores se calcula

$$\begin{aligned} \text{FK}_{1;ij} &= \exp \left[-Cq_1 \left(\left(\frac{q_3}{\tau(\eta^{(j)} - \xi^{(i)})} + \frac{q_4}{\tau(\eta^{(j)} + \xi^{(i)})} \right) + S_{0;ij} - S_{1;ij} \right) + S_{2;ij} + \mathbf{A}_{ij} \right] \\ \text{FK}_{2;ij} &= \exp \left[-Cq_2 \left(\left(\frac{q_3}{\tau(\eta^{(j)} + \xi^{(i)})} + \frac{q_4}{\tau(\eta^{(j)} - \xi^{(i)})} \right) + S_{0;ij} - S_{1;ij} \right) + S_{2;ij} + \mathbf{A}_{ij} \right] \end{aligned}$$

Si $\text{MM} = 1$ entonces se está en una situación inicial y en este caso,

$$\text{Error} = \sum_{ijk} \left[\frac{\text{FK}_{k;ij} - (\omega_{ijk} + 1)}{\omega_{ijk} + 1} \right]^2.$$

Mas si $\text{MM} = 0$ el error resulta de comparar a ω con su valor anterior,

$$\text{Error} = \text{Max}_{ijk} |\omega_{ijk} - g_{ijk}|.$$

Al final, se hace $\forall i, j, k : g_{ijk} := \omega_{ijk}$, para una posterior iteración.

En tanto que el error sea superior a 10^{-9} , se realiza una actualización del punto actual ω de acuerdo con el método de Newton-Raphson: El nuevo punto ha de ser ω_1 tal que

$$\omega_1 = \omega - [J\mathbf{G}(\omega)]^{-1} \cdot \mathbf{G}(\omega)$$

donde $\mathbf{G} : \omega \mapsto \mathbf{FK}(\omega) - \omega - \mathbf{1}$ y J es el operador *jacobiano* o equivalentemente,

$$J\mathbf{G}(\omega)(\omega_1 - \omega) = \mathbf{G}(\omega) \tag{5.8}$$

Es decir, sea $\sigma_2 : \llbracket 1, n_t \rrbracket \times \llbracket 1, n_1 \rrbracket \times \llbracket 1, 2 \rrbracket \mapsto \llbracket 1, n_P \rrbracket$, donde $n_P = 2n_t n_1$, la función de almacenamiento lineal $(i, j, k) \mapsto 2((i-1)n_1 + (j-1)) + k$. Todo arreglo tridimensional con índices en $\llbracket 1, n_t \rrbracket \times \llbracket 1, n_1 \rrbracket \times \llbracket 1, 2 \rrbracket$ se identifica, mediante la enumeración σ_2 , con un arreglo lineal con índices en $\llbracket 1, n_P \rrbracket$.

Para cada $I = \sigma_2(i, j, k)$, $(i, j, k) \in \llbracket 1, n_t \rrbracket \times \llbracket 1, n_1 \rrbracket \times \llbracket 1, 2 \rrbracket$, el arreglo \mathbf{FAC}_I se identifica con el arreglo $\mathbf{f}_I \in \mathbb{R}^{n_P+2}$ tal que $\mathbf{f}_I|_{n_P} = \mathbf{FAC}_I$ (módulo la enumeración σ_2) y $\mathbf{f}_I(n_P + 1) = \mathbf{f}_I(n_P + 2) = p_{ijk}$.

Sea $\mathcal{F} \in \mathbb{R}^{n_P \times (n_P+2)}$ la matriz que para cada $I \leq n_P$ tiene como I -ésimo renglón al arreglo \mathbf{f}_I . Denotemos por \mathbf{f}^J a la J -ésima columna de \mathcal{F} , $J \leq n_P + 2$, y por $\mathcal{F}^{(n_P)} = [\mathbf{f}^1 \dots \mathbf{f}^{n_P}]$ a la matriz consistente de las primeras n_P columnas de \mathcal{F} . Se resolverá aquí a los sistemas de ecuaciones

$$\mathcal{F}^{(n_P)}\omega_0 = \mathbf{f}^{n_P+1}, \quad \mathcal{F}^{(n_P)}\omega_1 = \mathbf{f}^{n_P+2} \tag{5.9}$$

o equivalentemente, al tomar transpuestas en todas partes,

$$\omega_0^T (\mathcal{F}^{(n_P)})^T = (\mathbf{f}^{n_P+1})^T \omega_1^T (\mathcal{F}^{(n_P)})^T = (\mathbf{f}^{n_P+2})^T \tag{5.10}$$

Así pues, se busca triangular a \mathcal{F}^T mediante el procedimiento de *reducción gaussiana*. Con éste, se obtiene matrices triangulares \mathcal{V} , \mathcal{U} , *inferior* y *superior* respectivamente, tales que

- $\mathcal{V} \in \mathbb{R}^{(n_P+2) \times n_P}$, $\mathcal{U} \in \mathbb{R}^{n_P \times n_P}$,
- \mathcal{U} es no-singular,
- los elementos en la diagonal de \mathcal{V} son 1 y
- $\mathcal{F}^T \cdot \mathcal{U}^{-1} = \mathcal{V}$.

De hecho, si no aparece involucrada ninguna matriz de permutación, entonces $\mathcal{V} \cdot \mathcal{U} = \mathcal{F}^T$ es una suerte de descomposición-LU de \mathcal{F}^T .

Evidentemente, las soluciones de las ecuaciones 5.10 han de serlo también de las ecuaciones

$$\omega_0^T \mathcal{V}_{(n_P)} = \mathbf{v}_{n_P+1} \quad , \quad \omega_1^T \mathcal{V}_{(n_P)} = \mathbf{v}_{n_P+2} \quad (5.11)$$

las cuales pueden ser resueltas directamente pues su matriz de coeficientes es triangular.

Para almacenar matrices triangulares inferiores, consideremos la enumeración por columnas, de izquierda a derecha, de la parte debajo de la diagonal principal de \mathcal{V} :

$$\phi_{inf,n_P}(I, J) = I + (n_P + 2)(J - 1) - \frac{1}{2}J(J + 1),$$

mediante la cual \mathcal{V} se realiza como un arreglo lineal $\phi_{inf,n_P}(\mathcal{V})$ de $\frac{1}{2}(n_P+1)(n_P+2)$ entradas. Así, se tiene que las entradas no-nulas en la J -ésima columna de \mathcal{V} , $[v_{IJ}]_{I=J+1, \dots, n_P+2}$, quedan en el bloque de entradas contiguas

$$\phi_{inf,n_P}(\mathcal{V}) \llbracket n_P(J - 1) - \frac{1}{2}J(J - 5) - 1, J(n_P - \frac{1}{2}(J - 3)) \rrbracket.$$

Se omite a las entradas en la diagonal pues éstas quedan supuestas con el valor 1.

Similarmente, para almacenar matrices triangulares superiores, consideramos la enumeración por renglones, de arriba a abajo, de la parte de encima de la diagonal principal de \mathcal{U} , incluyendo a ésta:

$$\phi_{sup,n_P}(I, J) = n_P(I - 1) - \frac{1}{2}I(I - 5) + J - 2,$$

mediante la cual \mathcal{U} se realiza como un arreglo lineal $\phi_{sup,n_P}(\mathcal{U})$ de $\frac{1}{2}(n_P + 2)(n_P + 3)$ entradas. Así, se tiene que las entradas no-nulas en el I -ésimo renglón de \mathcal{U} , $[u_{IJ}]_{J=I, \dots, n_P+2}$, quedan en el bloque de entradas contiguas

$$\phi_{sup,n_P}(\mathcal{U}) \llbracket n_P(I - 1) - \frac{1}{2}I(I - 7) - 2, n_P I - \frac{1}{2}I(I - 5) \rrbracket.$$

Por $\mathcal{F} = (f_{IJ})_{\substack{I \leq n_P+2 \\ J \leq n_P+2}}$ nos referiremos a los valores actuales de la matriz cuya transpuesta ha de ser triangulada. De acuerdo con el procedimiento de *reducción gaussiana*:

1. Inicialmente, para $J = 1$ se hace $v_{IJ} = \frac{f_{JI}}{f_{JJ}}$, $I = J + 1, \dots, n_P + 2$.
2. Para cada $J = 2, \dots, n_P + 1$ se actualiza a los valores siguientes:

- a) A los de \mathcal{F} , premultiplicando a su transpuesta por el bloque ya calculado de \mathcal{V} . Es decir, para $J_1 = 1, \dots, J-1$ y para $J_2 = J_1 + 1, \dots, n_P + 2$: $f_{JJ_2} := f_{JJ_2} - v_{J_2 J_1} f_{JJ_1}$.
- b) A los de \mathcal{V} para usarlos como pivotes en la siguiente iteración: Para $I = J + 1, \dots, n_P + 2$, se hace $v_{IJ} = \frac{f_{II}}{f_{JJ}}$.

Enumeremos a la matriz triangular inferior $\mathcal{V} \in \mathbb{R}^{(n_P+2) \times n_P}$ por renglones:

$$\mathcal{V} = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_{n_P} \\ \mathbf{v}_{n_P+1} \\ \mathbf{v}_{n_P+2} \end{bmatrix}$$

donde para cada $i \leq n_P$ el renglón \mathbf{v}_i tiene 1 en su i -ésima entrada y una lista de $n_P - i$ ceros:

$$\mathbf{v}_i = \mathbf{v}'_i \oplus [1] \oplus \mathbf{0}_{n_P-i}^T, \quad \mathbf{v}'_i \in \mathbb{R}^{i-1}.$$

Sea $\mathcal{V}_{(n_P)}$ la matriz que se obtiene al suprimir en \mathcal{V} sus dos últimos renglones. Una solución $\boldsymbol{\omega} \in \mathbb{R}^{n_P}$ del sistema de ecuaciones

$$\boldsymbol{\omega}^T \mathcal{V}_{(n_P)} = \mathbf{b}^T \tag{5.12}$$

se calcula directamente pues $\mathcal{V}_{(n_P)}$ es triangular inferior. Evidentemente:

$$\begin{aligned} \omega_{n_P} &= b_{n_P} \\ \forall j = 1, \dots, n_P - 1 : \omega_{n_P-j} &= b_{n_P-j} - \sum_{k=n_P+1-j}^{n_P} v_{k, n_P-j} \omega_k \\ &= b_{n_P-j} - \sum_{k=1}^j v_{n_P-(k-1), n_P-j} \omega_{n_P-(k-1)} \end{aligned}$$

Posteriormente se calcula una solución $\boldsymbol{\omega}_0$ de 5.12, con $\mathbf{b}^T = \mathbf{v}_{n_P+1}$, y una solución $\boldsymbol{\omega}_1$ de 5.12, con $\mathbf{b}^T = \mathbf{v}_{n_P+2}$ con lo cual se resuelve el par de ecuaciones 5.10.

Si $M = 0$ la solución $\boldsymbol{\omega}_0$, de la primera ecuación en 5.11, se almacena en una variable llamada OME, en tanto que si $M = 1$ se almacena en una variable llamada ODE. En cualquier caso, la solución $\boldsymbol{\omega}_1$, de la segunda ecuación en 5.11, se almacena en una misma variable llamada OME1. Finalmente se calcula a todos los elementos necesarios para plantear el sistema de ecuaciones 5.8 por renglones. Al calcular una solución $\boldsymbol{\omega}_2 = \boldsymbol{\omega}_1 - \boldsymbol{\omega}$, triangulando el sistema de ecuaciones 5.8, se puede considerar como punto actual $\boldsymbol{\omega}_2 + \boldsymbol{\omega}$ para realizar una nueva iteración de este proceso de aproximación.

En este procedimiento se recibe como argumento de entrada el índice $I = \sigma_2(i, j, k)$, con $i \leq n_t$, $j \leq n_1$ y $k \leq 2$. Como salida, producirá tres arreglos $n_t \times n_1$: **COX**_{ij}, **COK**_{ij} y **FAC**_{ij} y un escalar p_{ijk} .

Utilizando las matrices formadas se hace

$$\begin{aligned}\mathbf{COX}_{ij} &= \left((XF1 + XF4 - XL1 - XL4)_{ij\mu\nu} \right)_{\mu \leq n_t}^{\nu \leq n_1} \\ \mathbf{COK}_{ij} &= \left((XK1 + XK4)_{ij\mu\nu} \right)_{\mu \leq n_t}^{\nu \leq n_1}.\end{aligned}$$

Para cada $\mu \leq n_t$, $\nu \leq n_1$, $\kappa \leq 2$, se define

$$\mathbf{FAC}_{ijk\mu\nu\kappa} = \mathbf{FK}_{\kappa;\mu\nu} r_k (\mathbf{COK}_{ij;\mu\nu} - q_k q_\kappa \mathbf{COX}_{ij;\mu\nu}),$$

y sólo cuando $(\mu, \nu, \kappa) = (i, j, k)$ se reduce en 1 el correspondiente valor

$$\mathbf{FAC}_{ijkijk} := \mathbf{FAC}_{ijkijk} - 1.$$

Se calcula también el valor

$$p_{ijk} = \sum_{\mu\nu\kappa} \mathbf{FAC}_{ijk\mu\nu\kappa} \omega_{\mu\nu\kappa} - \mathbf{FAC}_{ijkijk} + 1.$$

Se produce como salida la lista de arreglos:

$$\left(\mathbf{COX}_{ij}, \mathbf{COK}_{ij}, \mathbf{FAC}_{ijk} = (\mathbf{FAC}_{ijk\mu\nu\kappa})_{\mu,\nu,\kappa}, p_{ijk} \right)$$

Teniendo ya una solución $\omega \in \mathbb{R}^{n_P}$ de la ec. 3.15, se le escribe en los archivos omega, y se calculan otros valores dependientes de ω .

5.2. Entradas y salidas

A continuación se explica un análisis detallado de las variables utilizadas y que es lo que representan

Variable	Descripción	Tipo
TEMP	Temperatura	real
EPS	Constante dieléctrica	real
DIAM	Diámetro de las esferas, una en el eje positivo y otra en el eje negativo, el diámetro de ambas es igual	real
RR	Distancia entre los centros de las esferas	real
TAU	Distancia del centro de la esfera al eje que las separa	real
RO1	Distancia entre el extremo superior de la esfera en el eje de las abcisas si el valor es negativo indica que las esferas se traslapan	real
RO2	Distancia del extremo inferior de la esfera a el eje de las abcisas si el valor es negativo indica que las esferas se traslapan	real

N1	Tamaño del plano en el eje de las abcisas	entero
N2 + N3	Tamaño del plano en el eje ordenado no debe ser mayor a 1	entero
NPAR	Densidad del plano en el eje de las abcisas	entero
NPAS	Densidad del plano en el eje de las ordenadas	entero
NPAT	Densidad del plano en el eje de las ordenadas a partir del valor de N3	entero
Q1,Q2,Q3,Q4	Cargas utilizadas en el sistema	entero
KS	Indica el inicio del sistema	binario
IC	Controla el primer ciclo del sistema	entero
IS	Controla el segundo ciclo del sistema	entero
XYT	Normaliza el valor de TAU	entero

Entre otros que indican cuantas veces es necesario que el sistema se ejecute para una mejor aproximación del resultado a experimentos realizados.

Ademas de *eta*, *ksi1* y *ksi2*, y los valores que indican la densidad del plano, se solicita al usuario un conjunto de coordenadas límite y los índices en los que se encuentra cada coordenada, con el propósito de generar una malla dentro del plano de trabajo con densidades variables, también a partir de estos valores se genera la representación original que solicita el sistema que resuelve la ecuación OZ. Los puntos iniciales del MNR, son calculados aleatoriamente dentro del plano definido para trabajar.

Las gráficas del comportamiento de la solución se obtienen a partir de la manipulación de los valores que va arrojando el sistema como son a partir de la primer coordenada *ksi2* se realiza un desplazamiento de la malla que ésta dada por el valor de la distancia del centro de la esfera a el eje que separa las dos esferas, con este al realizar la conversión de coordenadas planas a coordenadas prolatas se observa en la gráfica una esfera, consecuencia de la distancia descrita anteriormente.

Como salida genera un conjunto de gráficas que muestran el comportamiento del sistema, los datos para estas gráficas son leídos de archivos de texto por GNUPLOT, y un conjunto de archivos en los que se encuentran las aproximaciones a las que se llegó con los datos introducidos, y se escriben en caso de que después de 100 ejecuciones internas del programa no se ha conseguido una aproximación de 1×10^{-9} el sistema termina su ejecución.

Capítulo 6

Conclusiones y trabajo futuro

6.1. Conclusiones

El tratamiento de la ecuación de Ornstein - Zernike, es sumamente compleja, pues implica el procesamiento de diversas matrices, lo que involucra mucho tiempo de ejecución pues tiene una complejidad del orden $O((knm)^2)$, tanto en tiempo como en espacio, donde k es el número de partículas consideradas.

Las especificaciones de mallas presentadas permiten determinar de diversas maneras una misma región en el plano, distinguiendo subregiones con diversos refinamientos en la malla. Así que cada especificación puede usarse según sea conveniente en las diversas etapas del procesamiento numérico de la ecuación integral de que se trate.

El sistema ha sido diseñado buscando además de una ejecución eficiente, que sea amigable para el usuario sin amplios conocimientos de la física del problema, proporcionando adecuadamente los datos solicitados por el sistema para su correcto funcionamiento. También impide que en un momento determinado el sistema caiga en condiciones de *overflow* y de *underflow*, ya que al especificar la malla se controla este tipo de sucesos a lo largo de la ejecución del programa, y se cuenta con una mayor claridad de tipo geométrico dando números que siempre están bien definidos.

El sistema se implementó en un servidor UNIX permitiendo la interacción del sistema con el usuario, por medio de la interfaz creada en PHP. Debido a la complejidad del sistema al encontrar un punto en el que converja la ecuación de Ornstein - Zernike, el tiempo de ejecución del sistema es grande, dependiendo del número y refinamiento que se desee utilizar en el sistema, por lo que el usuario tiene que esperar largo tiempo para obtener resultados del sistema, pero en este lapso de tiempo, tiene oportunidad de checar como es que se va comportando el sistema y en un momento determinado cancelar su ejecución.

6.2. Trabajo futuro

Existe el proyecto de paralelizar el programa ya que existen varias rutinas las cuales no dependen una de la otra por lo que se puede hacer que trabajen en paralelo buscando reducir su tiempo de ejecución, se espera reducir su complejidad al orden de $O(knm \log(knm))$ para lo cual se requiere conocer la simulación matemática a fondo para que la implementación del sistema en un lenguaje de programación paralelo sea comprensible y fácil de realizar. Además se puede reducir el tiempo de cómputo, con la limitación del número de procesadores.

Apéndice A

Manual de usuario del sistema

En el presente manual se provee una guía para un usuario que desee utilizar el sistema que resuelve la ecuación OZ, sin necesidad de conocer el comportamiento interno del programa, es decir sin tener un conocimiento profundo de la Física involucrada en el problema, ni los fundamentos matemáticos, ni los métodos de programación utilizados a este usuario solo le interesan los resultados de los datos introducidos en el sistema y, sí acaso, el análisis del comportamiento de la solución.

Al iniciar el sistema el usuario se encuentra con la pantalla presentada en [A.1](#)

Existen 3 tipos de usuario, los que nos competen como *usuario no docto* los hemos denominado Autorizado y Sin privilegios, en la sección [A.1](#) mostramos las pantallas del usuario autorizado, y en la sección [A.2](#) las del usuario Sin privilegios.

A.1. Usuario autorizado

Este usuario es aquel que tiene el permiso de ejecutar el sistema completo que resuelve la ecuación OZ, además de consultar los procesos y compararlos, y puede hacer pruebas del sistema introduciendo algunos parámetros para la generación de las gráficas.

Después de haber accedido como este tipo de usuario el sistema muestra la pantalla mostrada en [A.2](#) el usuario tiene la opción de elegir entre las tres formas de trabajo que se mencionaron, es decir puede **Iniciar un proceso**, **Consultar un proceso** o **Realizar pruebas** con algunos parámetros. Si el usuario elige **Iniciar un proceso**, se encuentra con la pantalla [A.3](#). En ella se le solicita la introducción de un conjunto de parámetros que utiliza el sistema en su totalidad, por lo que si el usuario omite algún dato, el sistema por *default* asignará el valor de 0, a las variables en general, excepto a los valores de las variable N1, N2, N3, pues estas tiene la función de generar un conjunto de campos adicionales. A estas variables se les asigna el valor 1, por *default*, en la tabla [A.1](#) se describen los datos introducidos al sistema y su función.

Al momento de pulsar el botón **Dar coordenadas**, se generan unos cuadros para introducir los valores de las coordenadas como se explicó en [3.5.1](#), y se muestra la pantalla [A.4](#). Al dar click en el botón **Generar datos iniciales del procedimiento de NEWTON - RAPHSON** y habiéndole dado un nombre al proceso, se crean los datos de inicio del procedimiento de Newton-Raphson, que es utilizado durante la ejecución del sistema. Después de la creación de estos datos iniciales aparece la pantalla mostrada en [A.5](#) en la cual también

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:
[Dr. Guillermo Morales Luna](#)

Para acceder:

Usuario:

Password:

Figura A.1: Pantalla de acceso.

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:
[Dr. Guillermo Morales Luna](#)

Eres Con permiso

Procesos

¿Que desea hacer?

Iniciar proceso Consultar proceso Pruebas de los parámetros de los proceso

[Salir](#)

Figura A.2: Opciones de trabajo.

Estudio computacional de la ecuación de Ornstein-Zernike

Datos entrada proceso

Introduzca los datos

Nombre del proceso	<input type="text"/>			
NTRY	<input type="text" value="0"/>	NASI	<input type="text" value="0"/>	
Q1	<input type="text" value="0"/>	Q2	<input type="text" value="0"/>	Q3 <input type="text" value="0"/> Q4 <input type="text" value="0"/>
RO1	<input type="text" value="0.0"/>	RO2	<input type="text" value="0.0"/>	
TEMPERATURA	<input type="text" value="0.0"/>	CONSTANTE DIELECTRICA	<input type="text" value="0.0"/>	
XYT	<input type="text" value="0"/>	IC	<input type="text" value="0"/>	LREAD <input type="text" value="0"/>
DIAMETRO	<input type="text" value="0.0"/>	RR	<input type="text" value="0.0"/>	TAU <input type="text" value="0.0"/>
N1	<input type="text" value="1"/>	N2	<input type="text" value="1"/>	N3 <input type="text" value="1"/>
NPAR	<input type="text" value="0"/>	NPAS	<input type="text" value="0"/>	NPAT <input type="text" value="0"/>

Dar coordenadas

[Salir](#)

[Regresar a opciones](#)
[Usuario Autorizado](#)

Figura A.3: Entrada de datos.

Dato	Descripción
NTRY	Cota de IS el cual es el número de veces que se ha de realizar el procedimiento total, esta debe ser un número entero.
NASI	Cota de IS en el segundo ciclo, el cual es el número de veces que se ha de realizar el procedimiento total, ésta debe ser un número entero.
Q1,Q2,Q3,Q4	Valor de las cargas eléctricas, estas deben ser un número entero.
TEMP	Temperatura del líquido a evaluar
EPS	Constante dieléctrica del liquido a evaluar
DIAM	Diámetro de la partícula
RR	Separación entre la partícula y el eje de las abscisas
TAU	Separación entre ambas partículas que forman el complejo.
N1	Número de puntos a considerar en el eje η , ésta debe ser un número entero.
N2	Número de puntos a considerar en el eje ξ , con $ \xi \leq \frac{1}{2}(\tau - DIAM)$, ésta debe ser un número entero.
N3	Número de puntos a considerar en el eje ξ , con $1 \geq \xi \geq \frac{1}{2}(\tau - DIAM)$, esta debe ser un número entero.
NPAR	Número de bloques a considerar en el eje η , esta debe ser un número entero.
NPAS	Número de bloques a considerar en el eje ξ , con $ \xi \leq \frac{1}{2}(\tau - DIAM)$, ésta debe ser un número entero.
NPAT	Número de bloques a considerar en el eje ξ , con $1 \geq \xi \geq \frac{1}{2}(\tau - D)$, ésta debe ser un número entero.
IS	Contador del número de veces que se ha de realizar el procedimiento total, esta debe ser un número entero.

Tabla A.1: Tabla de las variables solicitadas

Datos entrada proceso

Introduzca los datos

Nombre del proceso

NTRY NASI

Q1 Q2 Q3 Q4

RO1 RO2

TEMPERATURA CONSTANTE DIELECTRICA

XYT IC LREAD

DIAMETRO RR TAU

N1 N2 N3

NPAR NPAS NPAT

N1

Extremo	Separación
<input type="text" value="1"/>	<input type="text" value="1"/>
<input type="text" value="1"/>	<input type="text" value="0"/>
<input type="text" value="1"/>	<input type="text" value="0"/>

N2

Extremo	Separación
<input type="text" value="0"/>	<input type="text" value="1"/>
<input type="text" value="1"/>	<input type="text" value="0"/>
<input type="text" value="1"/>	<input type="text" value="0"/>
<input type="text" value="1"/>	<input type="text" value="0"/>
<input type="text" value="1"/>	<input type="text" value="0"/>

N3

Extremo	Separación
<input type="text" value="0"/>	<input type="text" value="1"/>
<input type="text" value="1"/>	<input type="text" value="0"/>

Generar datos iniciales del procedimiento de NEWTON - RAPHSON

[Salir](#) [Regresar a opciones de Usuario Autorizado](#)

Figura A.4: Entrada de datos y coordenadas.

Estudio computacional de la ecuación de Ornstein-Zernike

Datos entrada proceso

Introduzca los datos

Nombre del proceso	<input type="text" value="nuevo"/>		
NTRY	<input type="text" value="0"/>	NASI	<input type="text" value="0"/>
Q1	<input type="text" value="0"/>	Q2	<input type="text" value="0"/>
RO1	<input type="text" value="0.0"/>	RO2	<input type="text" value="0.0"/>
TEMPERATURA	<input type="text" value="0.0"/>	CONSTANTE DIELECTRICA	<input type="text" value="0.0"/>
XYT	<input type="text" value="0"/>	IC	<input type="text" value="0"/>
DIAMETRO	<input type="text" value="0.0"/>	RR	<input type="text" value="0.0"/>
N1	<input type="text" value="3"/>	N2	<input type="text" value="5"/>
NPAR	<input type="text" value="4"/>	NPAS	<input type="text" value="5"/>
		LREAD	<input type="text" value="0"/>
		TAU	<input type="text" value="0.0"/>
		N3	<input type="text" value="2"/>
		NPAT	<input type="text" value="2"/>

Se han generado los Puntos Iniciales del procedimiento de Newton-Raphson

[Salir](#)

 [Regresar a opciones de Usuario Autorizado](#)

Figura A.5: Generación de los datos de iniciales del procedimiento de Newton-Raphson.

Estudio computacional de la ecuación de Ornstein-Zernike

Se esta ejecutando el proceso Fortran

Salir

Regresar a opciones de
Usuario Autorizado

Figura A.6: Ejecución del sistema que resuelve la ecuación.

Estudio computacional de la ecuación de Ornstein-Zernike

Necesito un nombre para el proceso

Salir

Dar nombre al proceso

Regresar a opciones de
Usuario Autorizado

Figura A.7: Mandar a ejecutar el proceso sin un nombre definido.

se puede apreciar el botón `Ejecutar el proceso de la ecuación`, dando click a éste empieza la ejecución completa del sistema, como se muestra en la pantalla [A.6](#), en caso de no haber dado un nombre al proceso el sistema muestra entonces la pantalla [A.7](#) y una liga para proveer este nombre.

Al dar inicio al proceso del sistema, este se empieza a ejecutar en el servidor, y constantemente crea nuevos archivos que se pueden visualizar seleccionando la opción `Consultar proceso`, que vemos a continuación. El tiempo de ejecución del sistema varía dependiendo del tamaño de los valores $N1$, $N2$, y $N3$, por consiguiente es recomendable antes de elegir `Iniciar proceso`, elegir `Hacer pruebas`, para manipular de manera adecuada estos parámetros. Al terminar la ejecución del sistema aparece la pantalla mostrada en [A.8](#), que nos indica el tiempo que duro la ejecución las gráficas y los archivos generados se pueden ver en [A.9](#).

Si el usuario, en cambio, selecciona `Consultar proceso`, tendrá acceso a la pantalla mostrada en [A.9](#) Aquí puede apreciar todos los procesos que ha ejecutado, y también algún proceso que él haya puesto en ejecución actualmente, la única diferencia consiste en la cantidad de gráficas que le podrán ser mostradas, esto dependiendo del avance del sistema. El usuario al elegir que procesos quiere consultar sólo se le podrán mostrar de dos en dos debido al tamaño que requieren las gráficas para ser apreciadas, como se muestra

Estudio computacional de la ecuación de Ornstein-Zernike

El proceso: crear2 ha concluido
Tiempo de inicio: 2:30 am del 3 de febrero de 2004
Tiempo de termino: 15:34 pm del 25 de marzo de 2004

Salir

Regresar a opciones de
Usuario Autorizado

Figura A.8: Proceso concluido.

Estudio computacional de la ecuación de Ornstein-Zernike

Consulta procesos

Selecciona los procesos a consultar de la lista

Selecciona lo que deseas hacer

- Consultar
 Borrar

- crear
 f
 nuevo
 prueba
 prueba2
 prueba3
 prueba4
 res

Aceptar

Salir

Regresar a opciones de Usuario Autorizado

Figura A.9: Pantalla Consultar proceso.

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:

[Dr. Guillermo Morales Luna](#)

Mandaste borrar:

- nuevo
- prueba
- prueba2

Selecciona el que deseas restaurar

Aceptar

[Salir](#) [Regresar a consulta](#)

Figura A.10: Pantalla Borrar procesos.

en [A.11](#). También aquí puede elegir que gráficas desea ver.

Como se puede apreciar en esta pantalla [A.9](#) el usuario también tiene oportunidad de borrar procesos, el deberá elegir que hacer. Si selecciona borrar se guarda un respaldo de los procesos que eliminó en su carpeta, y en la pantalla [A.12](#) se le informa que procesos borró y tiene la opción de restaurarlos. Si el usuario escoge **Pruebas de los parámetros de los procesos** se le muestra la pantalla [A.13](#), el usuario debe llenar todos los campos si no lo hace por *default* los datos DIAMETRO, RR, TAU, NPAR, NPAT y NPAS tienen un valor de 0, y los datos N1, N2, N3 un valor por *default* 1 ya que estos 3 últimos campos crean nuevos campos como se puede ver en [A.14](#). Al dar click al botón **Dar coordenadas**. Al momento de dar click al botón **Hacer pruebas**, es necesario dar un nombre al proceso, sí no, el sistema manda un mensaje de error como se muestra en [A.7](#). Habiendo dado un nombre el sistema muestra la figura [A.16](#) en donde el usuario debe elegir el tipo de prueba que desea realizar, las opciones son: Generar pruebas ENPAR, Xkernel, Xkernel1, Xkernel12, XMTX. Después de elegir las el sistema muestra la pantalla en la cual indica que pruebas está realizando como se muestra en la figura [A.17](#).

Hasta el momento hemos descrito el sistema que esta dedicado al Usuario autorizado, en la siguiente sección describimos el uso que puede dar al sistema el usuario Sin privilegios.

A.2. Usuario sin privilegios

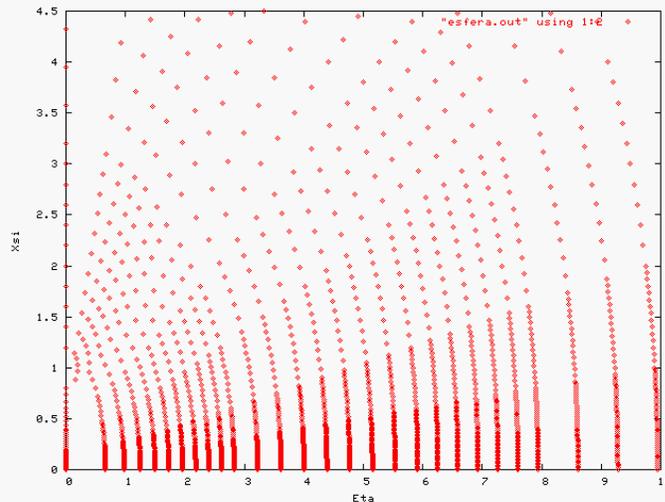
Este usuario es aquel que no tiene permiso de ejecutar el sistema completo que resuelve la ecuación OZ, únicamente puede consultar los procesos de prueba y compararlos introduciendo algunos parámetros para la generación de las gráficas que le es permitido

Estudio computacional de la ecuación de Ornstein - Zernike

Escoja que gráficas desea ver

proceso Y

- Cordenadas cartesianas
- Malla Malla desplazada
 - Función calculo de los términos de grado uno en el Exponente de la EOZ
 - Omega positivo Omega negativo
 - Xkernel positivo Xkernel negativo
 - Xkernel1 positivo Xkernel1 negativo
 - Xkernel2 positivo Xkernel2 negativo
 - Función Xmtx1 Función Xmtx2 Función Xmtx3
 - Función Xmtx4 Función Xmtx5 Función Xmtx6
- Coordenadas prolatas
- Malla Malla desplazada
 - Omega positivo Omega negativo
 - Xkernel positivo Xkernel negativo



proceso X

- Cordenadas cartesianas
- Malla Malla desplazada
 - Función calculo de los términos de grado uno en el Exponente de la EOZ
 - Omega positivo Omega negativo
 - Xkernel positivo Xkernel negativo
 - Xkernel1 positivo Xkernel1 negativo
 - Xkernel2 positivo Xkernel2 negativo
 - Función Xmtx1 Función Xmtx2 Función Xmtx3
 - Función Xmtx4 Función Xmtx5 Función Xmtx6
- Coordenadas prolatas
- Malla Malla desplazada
 - Omega positivo Omega negativo
 - Xkernel positivo Xkernel negativo
 - Xkernel1 positivo Xkernel1 negativo
 - Xkernel2 positivo Xkernel2 negativo
 - Función Xmtx1 Función Xmtx2 Función Xmtx3
 - Función Xmtx4 Función Xmtx5 Función Xmtx6

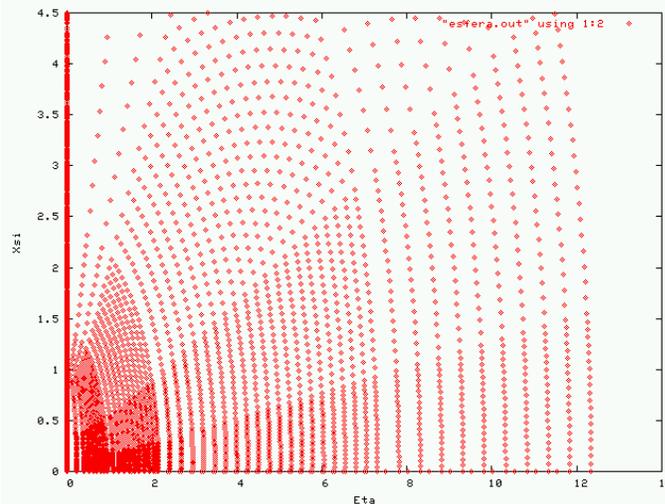


Figura A.11: Pantalla Consultar procesos por pares.

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:
[Dr. Guillermo Morales Luna](#)

Mandaste borrar:

- nuevo
- prueba
- prueba2

Selecciona el que deseas restaurar

Aceptar

[Salir](#) [Regresar a consulta](#)

Figura A.12: Pantalla Borrar procesos.

Estudio computacional de la ecuación de Ornstein-Zernike

Datos prueba proceso

Introduzca los datos

Nombre del proceso

DIAMETRO RR TAU

N1 N2 N3

NPAR NPAS NPAT

Dar coordenadas

[Salir](#) [Regresar a opciones Usuario Autorizado](#)

Figura A.13: Datos de entrada a pruebas.

Estudio computacional de la ecuación de Ornstein-Zernike

Datos prueba proceso

Introduzca los datos

Nombre del proceso

DIAMETRO	<input type="text" value="0.0"/>	RR	<input type="text" value="0.0"/>	TAU	<input type="text" value="0.0"/>
N1	<input type="text" value="1"/>	N2	<input type="text" value="1"/>	N3	<input type="text" value="1"/>
NPAR	<input type="text" value="0"/>	NPAS	<input type="text" value="0"/>	NPAT	<input type="text" value="0"/>

N1
Extremo Separación

N2
Extremo Separación

N3
Extremo Separación

[Salir](#) [Regresar a opciones Usuario Autorizado](#)

Figura A.14: Datos de entrada y coordenadas a pruebas.

Estudio computacional de la ecuación de Ornstein-Zernike

Necesito un nombre para el proceso

[Salir](#) [Dar nombre al proceso](#) [Regresar a opciones de Usuario Autorizado](#)

Figura A.15: Mandar a hacer pruebas del proceso sin un nombre definido.

Estudio computacional de la ecuación de Ornstein-Zernike

Pruebas de procesos

El proceso de prueba :

prueb

¿Que desea hacer?

Generar pruebas Enpar Generar pruebas Xkernel Generar pruebas Xkernel1 Generar pruebas Xkernel2 Generar pruebas XMTX

Aceptar

[Salir](#) [Regresar a opciones Usuario Autorizado](#)

Figura A.16: Tipos de pruebas.

Estudio computacional de la ecuación de Ornstein-Zernike

Pruebas enparfnpar

[Salir](#) [Regresar a opciones Usuario Autorizado](#) [Regresar a realizar pruebas](#)

Figura A.17: Ejecución de las pruebas.

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:
[Dr. Guillermo Morales Luna](#)

Eres Sin permiso

Introduzca los datos

Nombre del proceso

DIAMETRO	<input type="text" value="0.0"/>	RR	<input type="text" value="0.0"/>	TAU	<input type="text" value="0.0"/>
N1	<input type="text" value="1"/>	N2	<input type="text" value="1"/>	N3	<input type="text" value="1"/>
NPAR	<input type="text" value="0"/>	NPAS	<input type="text" value="0"/>	NPAT	<input type="text" value="0"/>

[Salir](#)

Figura A.18: Datos de entrada a pruebas de usuario no autorizado.

apreciar.

Después de haber accedido como este tipo de usuario el sistema muestra la pantalla mostrada en A.18, el usuario debe llenar todos los campos. Si no lo hace por *default* los datos DIAMETRO, RR, TAU, NPAR, NPAT y NPAS tienen un valor de 0, y los datos N1, N2, N3 un valor por *default* 1 ya que estos 3 últimos campos crean nuevos campos como se puede ver en A.19, al dar click al botón Dar coordenadas. Al momento de dar click al botón Hacer pruebas, es necesario dar un nombre al proceso de prueba. Sí no, el sistema manda un mensaje de error como se muestra en A.20. En cambio si el usuario introdujo un nombre para el proceso se le muestra la pantalla A.21, teniendo la opción de ver algunas gráficas de las funciones que involucra el sistema en su totalidad.

Damos por concluido el manual de usuario no docto del sistema que resuelve la ecuación OZ.

Estudio computacional de la ecuación de Ornstein-Zernike

Datos prueba proceso

Introduzca los datos

Nombre del proceso

DIAMETRO RR TAU

N1 N2 N3

NPAR NPAS NPAT

N1

Extremo Separación

N2

Extremo Separación

N3

Extremo Separación

[Salir](#)

Figura A.19: Datos de entrada y coordenadas a pruebas.

Estudio computacional de la ecuación de Ornstein-Zernike

Necesito un nombre para el proceso

[Salir](#) [Regresar a datos de entrada](#)

Figura A.20: Sin nombre de proceso de prueba.

Estudio computacional de la ecuación de Ornstein - Zernike

proceso

Cordenadas cartesianas

Malla Malla desplazada

Función calculo de los términos de grado uno en el Exponente de la.EOZ

Coordenadas prolatas

Malla Malla desplazada

Omega positivo Omega negativo

Xkernel positivo Xkernel negativo

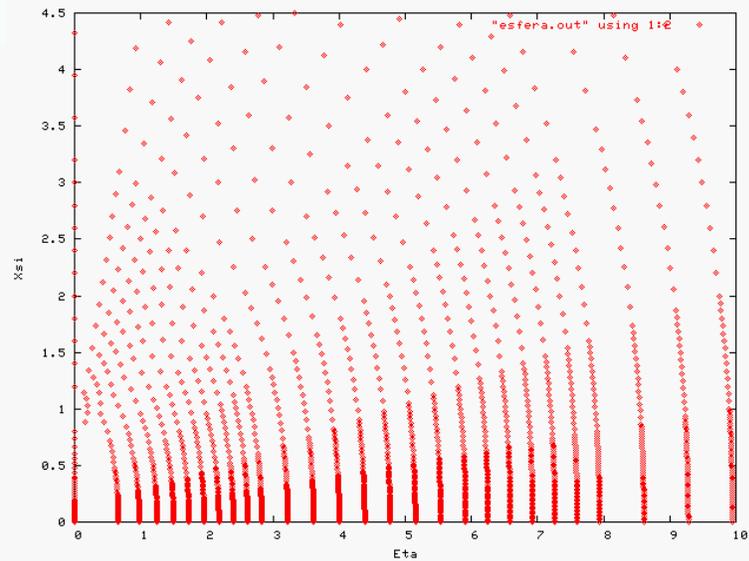


Figura A.21: Consultar proceso de prueba.

Apéndice B

Manual de administrador del sistema

Este manual está dirigido al administrador del sistema. En él se especifica el software con el que debe contar el servidor para que el sistema se ejecute correctamente, las carpetas adicionales que deben ser creadas y funciones con que cuenta el programa dirigidas al administrador.

B.1. Requerimientos de software

El sistema está basado en una arquitectura cliente - servidor, debe ser implementado en un servidor UNIX o LINUX dedicado para que su proceso no afecte otras aplicaciones, ni lo afecten a él en recursos. Tiene que instalarse un servidor APACHE, para trabajar con PHP4. Es necesario también un compilador para FORTRAN 77 y uno para C ANSI. Deberá contar con GNUPLOT instalado, un servidor de MySQL para la base de datos, capacidad para ejecutar SCRIPTS y espacio en disco suficiente para estar creando continuamente carpetas con el `login` del usuario. En cada una de ellas se almacena una carpeta con el nombre del proceso que ejecuta y dentro de esta carpeta se encuentran los archivos generados por cada uno de los procesos y el cliente debe poseer un navegador que permita visualizar imágenes `png`.

Una vez instalado el servidor APACHE el sistema crea la carpeta `/var/www`, en esta se deben almacenar todos los programas escritos en PHP4, que tienen la extensión `.php`, esto permite al APACHE visualizar las pantallas en un navegador *Web*.

B.2. Carpetas que deben ser creadas

Se necesita crear una carpeta llamada `/home/proyecto`, en ella se almacenan las carpetas de todos los usuarios y las de todos los procesos que ellos ejecutan, el dueño de esta carpeta debe ser `www-data`, deben copiarse aquí los SCRIPTS `proceso.sh` y `procesousuario.sh` (estos contienen código SHELL para la manipulación de las carpetas) para que puedan ser localizados por el sistema. Es necesario crearse la carpeta `/home/proyecto/respaldo` en esta se guardan los archivos en formato `.tar.bz2` de los procesos de aquellos usuarios que han sido dados de baja por el usuario Administrador si decidió respaldar todos los procesos de éste.

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:
[Dr. Guillermo Morales Luna](#)

Para acceder:

Usuario:

Password:

Figura B.1: Estudio computacional de la ecuación de Ornstein-Zernike.

Cada que se crea un usuario el administrador debe proporcionar una contraseña para que el usuario pueda acceder al sistema, y después de haber sido dado de alta se crea su carpeta de trabajo.

B.3. Manipulación de la base de datos

Como se mencionó en la sección 4.3 la base de datos *Petroleo* se encuentra diseñada en *MySQL*, y cuenta con una tabla llamada *usuario*, que esta definida por los campos *TipoUsuario*, *Nombre*, *login* y *contrasena*, directamente en *MySQL* el Administrador puede manipular ésta con la sintaxis propia de éste lenguaje.

La manipulación de la base de datos se hace a través de PHP, siendo posible administrar a los usuarios dandolos de alta, baja o haciendo modificaciones a sus datos, inclusive impidiendoles el acceso al sistema, ya que si no se encuentra en la base de datos, el usuario no podrá acceder al sistema.

B.4. Interfaz del usuario administrador

En principio el usuario debe dar una *contraseña* y dependiendo de ésta y su *login* tiene acceso a las diferentes opciones del sistema, en este momento nos abocaremos a describir las funciones del usuario Administrador.

Para acceder al sistema como ya se ha mencionado es necesario dar al sistema una *contraseña* y su *login*, como se muestra en B.1

Habiendo el usuario Administrador accedido al sistema se encontrará con la pantalla mostrada en B.2, en ésta como se puede apreciar existen las opciones de *Alta*, *Baja*, *Modificaciones* para administrar a los usuarios, y con respecto a los procesos se tiene la opción de *Consultar* o *Cancelar*.

Si elige la opción de *Alta* se encuentra con la pantalla mostrada en B.3. En ésta se le solicita elegir que tipo de usuario desea dar de alta, nombre completo, login, contraseña y

Estudio computacional de la ecuación de Ornstein-Zernike

Dentro:

Dirigida por:

[Dr. Guillermo Morales Luna](#)

Eres administrador

Alta usuario

Baja usuario

Modificar usuario

Consultar Proceso

Cancelar Proceso

[Salir](#)

Figura B.2: Opciones de administración del usuario **Administrador**.

la confirmación de la contraseña. Al pulsar el botón registrar se crea la carpeta del usuario que fue dado de alta, en caso de que el login del usuario ya exista se manda un mensaje como se muestra en la figura [B.4](#) Los tipos de **Alta** del usuario son:

- **Administrador**, un usuario dado de alta con este tipo, tiene privilegios como son administrar a los usuarios dandolos de alta, baja haciendo modificaciones a sus datos así como cancelar o consultar procesos.
- **Usuario autorizado**, este usuario tiene la posibilidad de iniciar la ejecución completa del sistema que resuelve la ecuación OZ, consultar sus procesos actuales y anteriores, así como hacer pruebas con algunos de los parámetros involucrados en el sistema para la generación de gráficas.
- **Usuario sin privilegios**, este usuario solo puede hacer pruebas con algunos de los parámetros de los procesos enfocado básicamente a la generación de gráficas de funciones que no demandas muchos recursos al servidor. Este es el tipo de **Alta** por *default* que se hace si no se escoge otro.

Si elige la opción **Baja** se muestra la pantalla [B.5](#) aquí se le solicita elegir que tipo de baja desea para el usuario, el nombre completo y el login. Si el usuario al que se quiere dar de Baja, no existe, ya fue borrado de la base de datos o no se dieron correctos los datos, el sistema muestra la pantalla [B.6](#) Los tipos de **Baja** del usuario son:

- **Guardar respaldo**, con esta opción se borra al usuario de la base de datos y se hace un respaldo comprimido de la información almacenada en su carpeta. Esta es la opción por *default* si no se selecciona alguna.

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:
[Dr. Guillermo Morales Luna](#)

Alta de usuarios:

Tipo de usuario Administrador Usuario autorizado Usuario sin privilegios

(max 30 caracteres):

Nombre de usuario (max 30 caracteres):

Login (max 10 caracteres):

Contraseña (entre 6 y 16 chars):

Confirma

[Atras](#) [Salir](#)

Figura B.3: Opción Alta.

Estudio computacional de la ecuación de Ornstein-Zernike

Problema:

Nombre de usuario existente, regrese y escoga otro.

Figura B.4: Opción Alta con problema.

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:
[Dr. Guillermo Morales Luna](#)

Baja de usuarios:

Guardar respaldo Borrar todo Acceso denegado al sistema

Nombre de usuario
(max 30 caracteres):

Login
(max 10 caracteres):

[Salir](#) [Ir Opciones Administrador](#)

Figura B.5: Opción Baja.

- **Borrar todo**, en esta opción se da de Baja al usuario de la base de datos y se borra su carpeta con todos sus procesos y pruebas, además no se guarda ningún respaldo de lo realizado por el.
- **Acceso denegado al sistema**, aquí se le da de Baja al usuario de la base de datos solamente, su carpeta y todos sus procesos quedan intactos.

Si se elige la opción **Modificar** se muestra la pantalla [B.7](#) aquí se solicita se escojan todas las opciones que se deseen modificar de algún usuario como son: **nombre**, **login**, **contraseña** y **tipo de usuario**. Si falta algún dato por llenar, o no se llenó correctamente la forma el sistema muestra la pantalla [B.8](#). Si el usuario que se desea modificar no existe, se muestra la pantalla [B.9](#) Es necesario seleccionar que opciones se desean modificar, y en el campo contiguo escribir el cambio a realizar, en el caso de tipo de usuario, es necesario hacer una selección si se marca, pues el usuario se convertirá en usuario **Sin Privilegios** por *default*.

Si elige la opción **Consultar proceso**, se muestra una tabla con los procesos de todos los usuarios sin distinción como se muestra en [B.10](#) aquí además de consultar los procesos también los puede borrar, se guarda una copia de respaldo comprimido en la carpeta `/home/proyecto/respaldo` por si posteriormente se quiere reinstalar. Se puede elegir borrar todos los procesos de la lista, pero para consultarlos solo es posible hacerlo de dos en dos, pues las opciones de las vistas de las gráficas son grandes como se muestra en [B.11](#).

Estudio computacional de la ecuación de Ornstein-Zernike

Problema:

No existe el usuario.

[Salir](#) [Regresar a Baja Usuario](#)

Figura B.6: Opción Baja con problema.

Estudio computacional de la ecuación de Ornstein-Zernike

Dirigida por:
[Dr. Guillermo Morales Luna](#)

Modifica datos usuarios:

Solo las opciones seleccionadas cambiarán

Login	<input type="text"/>		
<input type="checkbox"/> Modificar Nombre	Nuevo Nombre (max 30 caracteres):	<input type="text"/>	
<input type="checkbox"/> Modificar login	Nuevo login (max 10 caracteres):	<input type="text"/>	
<input type="checkbox"/> Modificar contraseña (entre 6 y 16 caracteres):	Nueva contraseña:	<input type="text"/>	Confirma contraseña: <input type="text"/>
<input type="checkbox"/> Modificar tipo de usuario	<input type="radio"/> Administrador	<input type="radio"/> Usuario autorizado	<input checked="" type="radio"/> Usuario sin privilegios
		<input type="button" value="Modificar"/>	
Salir	Ir a opciones de Usuario Administrador		

Figura B.7: Opción Modificar.

Estudio computacional de la ecuación de Ornstein-Zernike

Problema

La longitud debe ser entre 1 y 30 caracteres

[Salir](#)

[Regresar a Modificar Usuario](#)

Figura B.8: Opción Modificar con problema de llenado.

Estudio computacional de la ecuación de Ornstein-Zernike

Problema:

Nombre de usuario no existente, regrese y escoga otro.

[Salir](#)

[Regresar a Modificar Usuario](#)

Figura B.9: Opción Modificar con problema de usuario inexistente.

Estudio computacional de la ecuación de Ornstein-Zernike

Consulta procesos

Selecciona los procesos a consultar de la lista

Selecciona lo que deseas hacer

- Consultar
 Borrar
-

- hol
 prueba
 prueba1
 prueba2
 prueba3
 prueba4
 res

Aceptar

Salir

[Ir a Opciones Usuario Administrador](#)

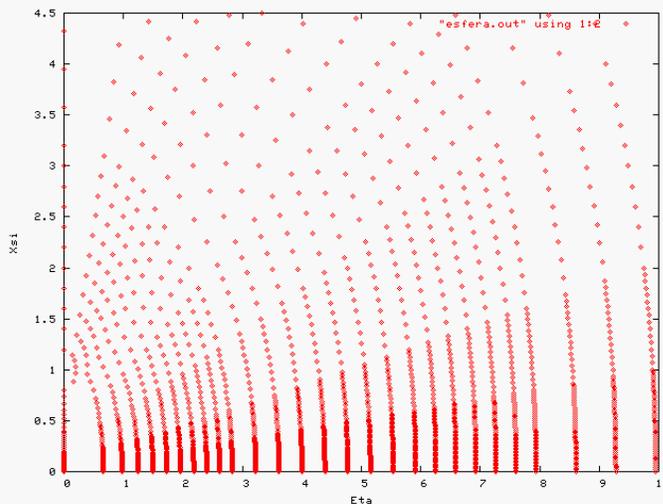
Figura B.10: Opción Consultar.

Estudio computacional de la ecuación de Ornstein - Zernike

Escoja que gráficas desea ver

proceso Y

- Cordenadas cartesianas
- Malla Malla desplazada
 - Función calculo de los términos de grado uno en el Exponente de la EOZ
 - Omega positivo Omega negativo
 - Xkernel positivo Xkernel negativo
 - Xkernel1 positivo Xkernel1 negativo
 - Xkernel2 positivo Xkernel2 negativo
 - Función Xmtx1 Función Xmtx2 Función Xmtx3
 - Función Xmtx4 Función Xmtx5 Función Xmtx6
- Coordenadas prolatas
- Malla Malla desplazada
 - Omega positivo Omega negativo
 - Xkernel positivo Xkernel negativo



proceso X

- Cordenadas cartesianas
- Malla Malla desplazada
 - Función calculo de los términos de grado uno en el Exponente de la EOZ
 - Omega positivo Omega negativo
 - Xkernel positivo Xkernel negativo
 - Xkernel1 positivo Xkernel1 negativo
 - Xkernel2 positivo Xkernel2 negativo
 - Función Xmtx1 Función Xmtx2 Función Xmtx3
 - Función Xmtx4 Función Xmtx5 Función Xmtx6
- Coordenadas prolatas
- Malla Malla desplazada
 - Omega positivo Omega negativo
 - Xkernel positivo Xkernel negativo
 - Xkernel1 positivo Xkernel1 negativo
 - Xkernel2 positivo Xkernel2 negativo
 - Función Xmtx1 Función Xmtx2 Función Xmtx3
 - Función Xmtx4 Función Xmtx5 Función Xmtx6

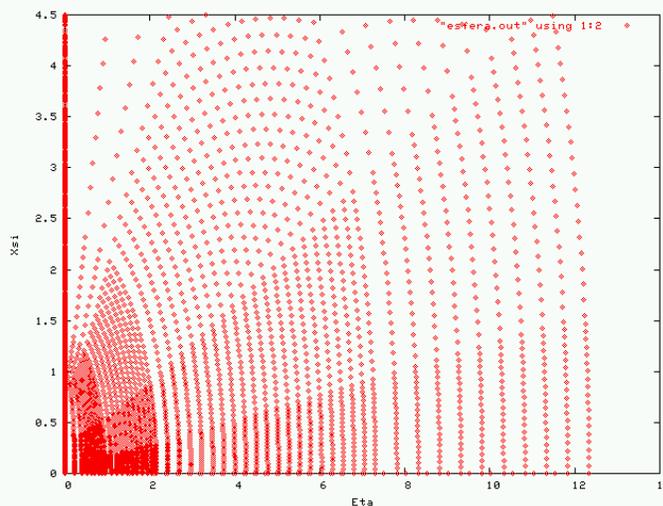


Figura B.11: Consultar procesos por pares.

Bibliografía

- [1] J. K. Ousterhout. Scripting: Higher-level programming for the 21st century. *IEEE Computer*, pages 23–30, 1998.
- [2] *PHP4*. En: <http://www.php.net/>.
- [3] J. A. White and A. González. The extended variable space approach to density functional theory in the canonical ensemble. *Journal of Physics: Condensed Matter*, 8 November 2002. Departamento de Física Aplicada, Universidad de Salamanca, Salamanca, España.
- [4] J-L López-Martín and M. Rassinger. An integral-equation approach to the electronic structure of liquid silicon. *Journal of Physics: Condensed Matter*, 21 April 1997. Rocasolano Chemical Fisica Institute , British, Columbia.
- [5] M.D. Carbajal Tinoco and Gonzalez Mozuelos. Effective attractions between like-charged colloidal particles. *Journal Of Chemical Physics*, 2002. Volume 117, Number 5.
- [6] Guillermo Morales Luna, Marcelo Lozada-Cassou, and Felipe Jimenez Angeles. *OZS: Numerical system to treat the Ornstein - Zernike equation (Reference Manual)*. no publicado, 2002.
- [7] Chi-Lun Lee and George Stell. Self-consistent ornstein zernike approach for monolayer films. *Journal of Physics : condensed mater*, May 2002. pp 4815 - 4822.
- [8] b R. Asgaria, B. Davoudia and B. Tanatarc. Hard-core yukawa model for two-dimensional charge stabilized colloids. *Journal of Statistical Physics*, October 2001. pp 891- 898.
- [9] L. Blum. Scaling for mixtures of hard ions and dipoles in the mean spherical approximation. *Journal of Statistical Physics*, February 2002. pp 431, 476, vol. 106.
- [10] Felipe Jimenez-Angeles, René Mesina, Chistian Holm, and Marcelo Lozada-Cassou. *Three particle correlation functions for an electrolyte: theory and molecular simulation results*. Cinvestav, 28 May 2002. arXiv:cond-mat/o2o5591 v1.
- [11] *Gnuplot viene incluido en el GNU freeware software*. En: <ftp://prep.ai.mit.edu/pub/gnu> (USA) ó <ftp://ftp.rediris.es/mirror/gnu> (España).
- [12] R. Stoer, I. *Introdution to Numerical Analysis*. Prentice Hall, 1980. Springer-Verlag.

- [13] *MySQL documentation*. En: <http://www.mysql.com/documentation> y <http://mx2.php.net/mysql>.

Los abajo firmantes, integrantes del jurado para el examen de grado que sustanterá la **Ing. Luz Virginia Morales Morón**, declaramos que hemos revisado la tesis titulada:

**Interfaces para un sistema de cómputo
que resuelve la ecuación de Ornstein-Zernike**

Y consideramos que cumple con los requisitos para obtener el Grado de Maestría en Ciencias en la especialidad de Ingeniería Eléctrica opción Computación.

Atentamente,

Dr. Guillermo Benito Morales Luna -----

Dr. José Oscar Olmedo Aguirre -----

Dr. Luis Gerardo de la Fraga -----