Universidad Autónoma de Madrid
Escuela Técnica Superior de Informática
Departamento de Ingeniería Informática

Refinamiento de la Estructura Tridimensional de Macromoléculas Biológicas Aisladas: Principios y Realización en Multiprocesadores

Tesis Doctoral

Luis Gerardo de la Fraga

Enero 1998

Unidad de Bioinformática,
Centro Nacional de Biotecnología-CSIC,
Campus Universidad Autónoma, 28049 Madrid, España.

Esta tesis fue elaborada
con $\LaTeX\,2_\varepsilon$. Disponible en internet,
en España en la dirección:
`<ftp://ftp.rediris.es/mirror/tex-archive>`

—Ustedes saben lo que aquí se trata en realidad —dijo Malcolm—. Todo este intento por controlar ... Estamos hablando de actitudes occidentales que tienen quinientos años de antigüedad. Comenzaron en la época en la que Florencia, en Italia, era la ciudad más importante del mundo. La idea básica de la ciencia, que había una nueva manera de contemplar la realidad, que era objetiva, que no dependía de creencias o nacionalidades, que era *racional*, era una idea fresca y emocionante en aquel entonces, ofrecía promesas y esperanza para el futuro, y borraba de un plumazo el antiguo sistema medieval, que tenía centenares de años de antigüedad. El mundo medieval de la política feudal, de los dogmas religiosos y de las odiosas supersticiones, cayó ante la ciencia. Pero, en honor a la verdad, eso se debía a que el mundo medieval realmente ya no funcionaba: no funcionaba en lo económico, no lo hacía en lo intelectual y no encajaba en el nuevo mundo que llegaba.

Malcolm tosió.

—Pero ahora —continuó— es la ciencia el sistema de creencias que tiene centenares de años de antigüedad. Y, al igual que el sistema medieval que la precedió, la ciencia está empezando a mostrarse inadecuada con el mundo. La ciencia ha obtenido tanto poder que sus límites prácticos comienzan a ser evidentes; es debido a la ciencia, principalmente, miles de millones de nosotros vivimos en un mundo pequeño, muy apretados e intercomunicándonos. Pero la ciencia no puede ayudarnos a decidir qué hacer con ese mundo, o cómo vivir. La ciencia puede elaborar un reactor nuclear, pero no nos puede decir que no lo construyamos. La ciencia puede fabricar plaguicidas, pero no nos puede decir que no lo usemos. Y nuestro mundo empieza a estar contaminado en áreas fundamentales, el aire, el agua y la tierra, como consecuencia de la ingobernable ciencia. —Suspiró—. Todo esto es obvio para cualquiera.

Se produjo un silencio. Malcolm yacía con los ojos cerrados, la respiración laboriosa. Nadie habló, y a Ellie le pareció que finalmente se había quedado dormido. Entonces, se volvió a sentar de forma abrupta.

—Al mismo tiempo, la gran justificación intelectual de la ciencia desapareció. Incluso desde Newton y Descartes, la ciencia nos brindó explícitamente la visión de un control total. La ciencia afirmó tener el poder de, a la larga, conocerlo todo, a través de su compresión de las leyes naturales. Pero, en el siglo XX, esa afirmación se hizo pedazos, más allá de toda posible reparación: primero, el principio de incertidumbre de Heisenberg fijó límites a lo que podemos saber sobre el mundo subatómico. *Oh, está bien —decimos—, ninguno de nosotros vive en un mundo subatómico. Eso no establece diferencia práctica alguna en nuestro paso por la vida.* Después, el teorema de Gödel fijó límites similares a la matemática, el lenguaje formal de la ciencia: los matemáticos solían creer que su lenguaje gozaba de alguna exactitud intrínseca especial, pero provenía de las leyes de la lógica. Ahora sabemos que lo que llamamos *razón* es sólo un juego arbitrario. No es algo especial, de la forma en que pensábamos que era.

⋮

—Estamos siendo testigos del final de una era científica. La ciencia, al igual que otros sistemas pasados de moda, se destruye a sí misma. A medida que gana en poder, se demuestra incapaz de manejar ese poder. Porque, ahora las cosas van demasiado aprisa. Hace cincuenta años, todo el mundo estaba mochales con eso de la bomba atómica. Eso era poder. Nadie podía imaginarse algo más. Sin embargo, apenas una década después de la bomba, empezamos a tener poder genético. Y el poder genético es con mucho más potente que el poder atómico. Y se encontrará en manos de todos. Estará en las herramientas de los hortelanos del patio de atrás. Para los experimentos de los colegiales. En laboratorios baratos para terroristas y dictadores. Y esto forzará a todo el mundo a hacerse idéntica pregunta —*¿Qué debería hacer con mi poder?*— que es precisamente la misma pregunta que la ciencia afirma no saber responder.

Ian Malcolm
Quinta Iteración. Control.

Agradecimientos

- A mi abuela, Dña. Cecilia de la Fraga, por educarme como la persona que soy.
- A ti, Maggie, por haberme convencido a iniciar el doctorado.
- A la Agencia de Cooperación Iberoamericana, del gobierno español, por haberme concedido una beca para iniciar los estudios de doctorado.
- Al Consejo Nacional de Ciencia y Tecnología, México, por concederme la beca que me permitió terminar esta tesis doctoral.
- A la Universidad Autónoma del Estado de Morelos, por concederme una beca parcial para finalizar el último año del doctorado.
- Al Dr. José María Carazo, por el constante apoyo, la confianza puesta en mi persona y en mi trabajo, y, muy especialmente, por la paciencia y comprensión mostrada conmigo en aquellos momentos en los que los resultados no acompañaban.

  Muy especialmente al Dr. Gabor T. Herman a quien le debo parte de las ideas que se hallan desarrolladas en esta tesis.

- Al Centro Nacional de Biotecnología, en especial a la Unidad de Bioinformática, por darme las facilidades para la elaboración de este trabajo.
- La colaboración de nuestro laboratorio con el Dr. Gabor T. Herman fue financiada por el proyecto CRG 960070 de la OTAN. La parte de la actividad de la Unidad de Bioinformática fue financiada con el proyecto BIO95-0768 de la Comisión Interministerial de Ciencia y Tecnología, España.

- A todos mis amigos, por su apoyo moral desinteresado: José Jesús Fernández (¡Así es Jose!), Cristina Risco (¡Hola Cris!), Joaquín Dopazo (Ximo), Sergio Marco.
- Especialmente a Montserrat Bárcena, cronopia cronopia.
- A la banda: Patrick Moutel, Catalina del Moral e Inés Póveda ¡Gracias chicos! Y a Ramón Roca y Sonia de Diego, que llegaron en el último momento.
- A Charo Fernndez, Marta Barradas, María Calle, Esther Calle, Mónica Chagoyen.
- A Marta Maroto, Fernando, Charo, Javi, María Jose, y muchos recuerdos, en donde sea que estés, a ti Manuel.
- A Juanma, Julita, Inés, Manolo, Ester, Ana.
- A ti, Elizabeth, por apoyarme en la distancia.

Y a internet ...

A mis amigos que siempre me acompañaron por el ciberespacio:

Javier de la Hidalga, Teresita Sánchez, Isaac Guzmán, Salvador Rivera, Carlos Dagostino y Clara García (CEGA).

Enrique Torrano, Mario Contreras, José Álvaro Benítez, Nelly Benítez.

Y a los que nunca conoceré: Alen, Tania, Claus.

Índice General

Índice de Figuras

Índice de Tablas

Capítulo 1

Introducción

Esta tesis versa sobre el refinamiento de la geometría en la reconstrucción tridimensional de macromoléculas biológicas a partir de imágenes obtenidas en el Microscopio Electrónico de Transmisión (MET).

El MET es un instrumento que extiende miles de veces las capacidades que ofrece el microscopio óptico, permitiendonos resolver estructuras biológicas desde el nivel celular hasta el atómico. En el estudio de las macromoléculas biológicas con el MET se presentan varios incovenientes, unos asociados a las características de las macromoléculas biológicas y otros referidos a las limitaciones propias del instrumento (estos incovenientes serán tratados con algún detalle más adelante, en el capítulo 2). La necesidad de "iluminar" a la muestra con electrones produce una ionización en los átomos que la componen (en su mayor parte son átomos no pesados) lo que produce un daño en su estructura. A fin de reducir este daño al mínimo es necesario mantener una dosis muy baja de electrones, que se traduce en un importante ruido estadístico en la imágenes. Además, las lentes magnéticas del MET presentan aberraciones que son mucho más importantes y difíciles de corregir que en el microscopio óptico. Estas características son fundamentales de cara a plantearse el trabajo habitual con especímenes biológicos.

Las imágenes producidas por el MET son recogidas en una película fotográfica sensible a los electrones. Estas fotografías del MET son conocidas como *micrografías*. En las micrografías la información tridimensional se colapsa en dos dimensiones, *perdiendo* por lo tanto una dimensión. Como reconstrucción tridimensional se entiende el proceso, desarrollado en una computadora, de combinar muchas micrografías digitalizadas, o partes de ellas, correspondientes a diferentes vistas de una estructura a fin de ganar la dimensión *perdida* y tener la visión en tres dimensiones del espécimen en estudio.

A fin de realizar la reconstrucción tridimensional, necesitamos la información de la localización de las imágenes bidimensionales relativas a un mismo sistema de coordenadas. El problema aquí es que se está severamente limitado en el control de la orientación por el instrumento, además del mismo componente del ruido ya mencionado; así, la orientación de las imágenes finalmente se logra obtener, pero siempre con cierta imprecisión. El termino *refinamiento de la geometría* se refiere, por lo tanto, al proceso de *mejorar* la orientación de las imágenes con respecto al volumen, o lo que es lo mismo, con respecto al sistema de coordenadas fijo a él.

Llegando a este punto ya podemos hacernos una idea de las disciplinas generales donde enmarcar esta tesis: biología, química, física, matemáticas e ingeniería, sin jerarquizar por el orden de las disciplinas, sino pensando en una interacción conjunta entre ellas. Específicamente, esta tesis se relaciona con *procesado de imagen*, *procesamiento digital de señales*, así como con la *programación*, *desarrollo de algoritmos* y su *aplicación en arquitecturas paralelas*.

El trabajo en esta memoria abarca el desarrollo de algoritmos y programas en arquitecturas paralelas para realizar el *refinamiento de la geometría* para dos de las técnicas de refinamiento ya desarrolladas [84, 91], así como la propuesta de tres técnicas nuevas. Las dos técnicas de refinamiento ya desarrolladas se basan en ideas simples propuestas hace varios años [117, 55] aunque no se habían llevado completamente a cabo por demandar grandes cantidades de cómputo (del orden de dias) que hacían prohibitiva su realización; así que tuvieron que esperar hasta que se construyeran ordenadores lo suficientemente potentes. Aunque actualmente ya existen tales ordenadores, el tiempo de ejecución de los programas sigue siendo relativamente alto (varias das o semanas), por lo que en esta tesis se han realizado versiones *paralelas* de los mismos. Un *programa paralelo* es, de forma muy general, aquél que divide el trabajo a realizar entre varias *máquinas* que ejecutan su parte correspondiente simultáneamente. La paralelización permite reducir el tiempo de ejecución y un mejor aprovechamiento de los recursos disponibles en una plataforma dada. Ahora bien, estas técnicas de refinamiento tienen un inconveniente: requieren un volumen de referencia. Este inconveniente puede ser tan grande que para obtener una reconstrucción tridimensional refinada hace necesario el haber obtenido previamente otra. Aún a pesar de este problema, estas dos técnicas son las únicas disponibles actualmente para su aplicación a imágenes obtenidas directamente del MET [96, 100].

Yendo más allá, y pensando en solventar el problema de la necesidad de contar con datos previos, en esta tesis se han desarrollado tres técnicas para el refinamiento de la geometría sin la necesidad de contar con un prototipo.

Una introducción al estudio del MET y los problemas asociados a las imágenes y la reconstrucción tridimensional serán tratados en el capítulo 2. Las técnicas de refinamiento serán expuestas en el capítulo 3. En los capítulos 4 y 5 se estudiarán en detalle las técnicas ya existentes y la descripción del desarrollo de los programas secuencial y paralelo. El desarrollo de las nuevas metodologías propuestas se tratará en el capitulo 6. La discusión del trabajo de tesis se presenta en el capítulo 8 y, finalmente, las conclusiones en el capítulo 9.

En el capítulo 7 se presenta una método híbrido que compara las técnicas ya establecidas y las conjunta con una de las técnicas nuevas propuestas.

Explícitamente no se han aplicado aún las nuevas técnicas expuestas en esta tesis a especímenes biológicos, sino que se han usado *fantasmas* con las condiciones de ruido que se obtendrían en situaciónes reales. Estos *fantasmas* son modelos tridimensionales generados por ordenador en los que podemos conocer nuestro volumen *a priori* (lo cual difícilmente se tiene en el caso de aplicación a especímenes reales). Nuestra hipótesis de trabajo (que habrá que confirmar por estudios ulteriores) es que sí estos métodos funcionan con modelos realistas, previsiblemente también funcionarán con las muestras reales.

Ahora situaremos el tema de este tesis dentro del marco de la microscopía electrónica con una ligera retrospección histórica.

El análisis cuantitativo de las micrografías electrónicas comenzó hace unos 30 años, con el estudio de los virus. Finch y Klug [44] compararon el registro de sombras de modelos de plástico y madera [1] con micrografías de partículas de virus teñidos negativamente. Aunque las representaciones generadas por ordenador pronto reemplazaron a los modelos, la determinación de la estructura permaneció como un proceso de ensayo-y-error de ajustar un modelo que encajara con las imágenes. La reconstrucción tridimensional eliminó el ensayo-y-error proveiendo un camino directo para obtener, a partir de las imágenes, el mapa tridimensional de la estructura. En 1975, la microscopía electrónica fue usada para visualizar, por primera vez, los agregados de α-hélices en la membrana proteica de la bacteriorrodopsina [62]. Las subunidades de bacteriorrodopsina se unen en la membrana en agregados cristalinos. Estos cristales bidimensionales son buenos especímenes de los cuales se puede extraer datos en alta resolución, porque cada imagen contiene miles de unidades idénticas en una orientación predefinida. El análisis estructural por microscopía electrónica de cristales bidimensionales ahora se extiende hasta esencialmente resolución atómica, estando el límite en 3–4 Å. Más recientemente, la microscopía electrónica se ha usado para resolver, hasta 9 Å de resolución, otra clase diferente de especímenes: agregados de simetría helicoidal [64, 79]. Las imágenes de éstas usualmente tienen menos subunidades que un solo cristal bidimensional, de manera que son necesarias más imágenes para alcanzar la misma resolución. Actualmente se han añadido una tercera clase de agregados a esta lista, partículas aisladas de simetría icosaédrica, un caso en el que aún hay menos unidades (o vistas) equivalentes por partícula, así que son necesarias más imágenes para alcanzar la alta resolución: Böttcher *et al* [7] combinaron 6 400 imágenes para obtener 7.4 Å, mientras que Conway *et al* [21] combinaron 600 para alcanzar 9 Å.

El trabajo de esta tesis se aplicaría a una cuarta clase: partículas aisladas sin, necesariamente, simetría interna. Por ejemplo, la proteína GroEL, que tiene una simetría rotacional de orden 7, se ha reconstruido a 30 Å de resolución [96] aplicando una de las técnicas de refinamiento de la geometría tratada en esta tesis (Cap. 4).

Dentro del contexto de las técnicas que permiten el estudio tridimensional de las macromoleculas biológicas, además de la microscopía electrónica, actualmente se cuenta con la Resonancia Magnética Nuclear (RMN) o la cristalografía de rayos X. También tanto la Microscopía de Efecto Túnel (STM) como la Microscopía de Fuer-

[1] Una presentación de estos modelos y de los instrumentos utilizados para realizar las primeras reconstrucciones tridimensionales de proteínas puede verse en el *Science Museum* de la ciudad de Londres, Inglaterra

zas Atómicas (AFM) pueden proporcionar imágenes de la superficie de las muestras a resolución cuasi-atómica [39]. La cristalografía de rayos X fue el primer método desarrollado [65, 86] y sigue siendo la técnica predominante en el análisis de la estructura atómica de las proteínas, a pesar de dos severas limitaciones: por un lado requiere el uso de grandes cristales tridimensionales de proteína, que no son fáciles de conseguir; por otro lado la recogida de datos de rayos X sólo mide una parte de la información, las amplitudes, debiéndose recurrir a otros medios experimentales para recuperar la otra parte, la fase. Por su parte, la RMN constituye un método muy prometedor para analizar la estructura de proteínas en solución, sin embargo la complejidad de los espectros obtenidos para moléculas de gran tamaño la limita a estudios de partículas de masas moleculares inferiores a unos 30 kD [76].

Las ventajas de la microscopía electrónica frente a la cristalografía de rayos X y la RMN [20] son las siguientes: requiere cantidades menores de proteína, las imágenes contienen información tanto de fase como de amplitud, no es estrictamente necesario trabajar con especímenes cristalinos, y permite el estudio de grandes agregados con o sin simetría. Por otra parte, el análisis estructural mediante microscopía electrónica no está exento de algunos incovenientes: el espécimen debe exponerse a condiciones de alto vacío, la materia biológica proporciona imágenes de bajo contraste, y las muestras se deterioran rápidamente bajo el efecto de la radiación electrónica.

No puedo terminar esta introducción sin formular una pregunta fundamental: ¿Por qué es importante el estudio de las macromoléculas biológicas? Filosóficamente porque se busca la respuesta a la meta-pregunta: ¿qué somos? Científicamente, para desentrañar el problema de la química de la vida [2] para lo que es necesario conocer en detalle su estructura. Comercialmente, el conocimiento de las relaciones estructura/función de las macromoléculas nos llevaría a desarrollar nuevas vacunas y fármacos de diseño [98].

[2]¿Qué pasará cuando conozcamos todo lo que se pueda saber sobre la química de la vida?, aparte de dejar la respuesta para discusiones metafísicas, es posible que se pueda construir entonces una máquina sintetizadora de alimentos como en la serie *Star Trek*

Capítulo 2

La Problemática de la Reconstrucción Tridimensional de Macromoléculas en el Microscopio Electrónico de Transmisión

2.1 La Microscopía Electrónica en Biología

La microscopía electrónica en estudios biológicos se destaca hoy en día como una metodología muy poderosa que proporciona datos con un alto contenido de información, en forma de imágenes. La extracción de información cuantitativa a partir de estas imágenes es una ardua tarea que se acomete con la aplicación de complejas tareas de Procesado Digital de Imágenes, que a su vez requieren gran consumo de recursos informáticos. Centrándonos en el área de la biología que estudia los complejos mecanismos de interacción de las moléculas —conocida como "Biología Molecular"—, los estudios realizados en esta tesis tienen su aplicación en el área de la "Biología Estructural de Macromoléculas Biológicas". El objetivo de estos estudios es siempre obtener información sobre la estructura tridimensional de una macromolécula determinada con el fin último de conocer en detalle su mecanismo de acción: esto es, obtener su estructura tridimensional como un paso hacia el desentrañamiento de su función biológica.

Los desarrollos en la biología moderna pretenden proporcionar una descripción cuantitativa de los complejos químicos que definen los organismos vivos. Para ello se está dedicando gran esfuerzo a desarrollar modelos detallados de complejos macromoleculares biológicos que permitan estudiar las relaciones estructura–función.

La determinación de la estructura de compuestos macromoleculares es en la actualidad uno de los problemas clave de la investigación bioquímica. Muchos procesos biológicos básicos, incluyendo el metabolismo de ácidos nucléicos, la fotosíntesis, la síntesis de proteínas y el ensamblaje de partículas virales, requieren la acción concertada de un gran número de componentes. La comprensión de la organización tridimensional de estos componentes, así como sus detalles estructurales, si es posible a nivel atómico, es imprescindible para la interpretación de su función.

2.2 Macromoléculas: Estructura y Función

El atributo más sobresaliente de los seres vivos es, quizás, su complejidad y su alto grado de organización. Poseen estructuras internas intrincadas que contienen muchas clases de moléculas complejas. Cada una de las partes componentes de un organismo vivo cumple un propósito o función específicos y, por tanto, es completamente legítimo preguntarse cuál es la función (o funciones) de una molécula determinada [1].

Las principales macromoléculas de la célula son los lípidos, los polisacáridos, las proteínas y los ácidos nucléicos. Todos ellas poseen estructuras complejas que les confieren unas propiedades características, habilitándolas para desempeñar la mayor parte de las funciones más importantes de la célula. Esta estructura es, a su vez, la encargada de controlar la interacción entre una macromolécula y otra, o entre las distintas partes de una misma macromolécula [1].

2.2.1 Las proteínas

Uno de los tipos de macromoléculas principales de la célula son las proteínas. Las células están formadas en gran parte por proteínas y son éstas las que determinan en gran medida la forma y la estructura de la célula, y en última instancia, la función celular.

Una de las funciones más importantes de las proteínas consiste en actuar como enzimas, catalizando reacciones químicas específicas. Otra función interesante es la de desempeñar el papel de elementos estructurales, proporcionando mecanismos de regulación de diversos procesos celulares. Las proteínas atienden a otras muchas funciones biológicas y son las más versátiles de las macromoléculas biológicas.

En principio, cualquier macromolécula proteica puede adoptar un número enorme de formas diferentes, llamadas conformaciones. Sin embargo, la mayoría de las proteínas de una célula se pliegan de forma estable adoptando una sola de estas conformaciones. Este comportamiento se debe a que casi toda la información que determina la conformación se halla contenida en la secuencia de *aminoácidos* (un tipo de molécula orgánica básica) que forma la proteína. Sin embargo, a pesar de este hecho, aún no se sabe predecir con exactitud la estructura tridimensional detallada de una proteína cuya secuencia de aminoácidos se conozca [1]. Por consiguiente, la conformación sólo puede ser determinada mediante los diversos métodos biofísicos de determinación de estructuras comentados en la Introducción (al final de la pág. 3): cristalografía de rayos X, resonancia magnética nuclear (RMN) y microscopía electrónica.

Aunque el número de maneras diferentes en que una proteína se podría plegar es astronómicamente elevado, parece que existe una cierta lógica en la forma en que se

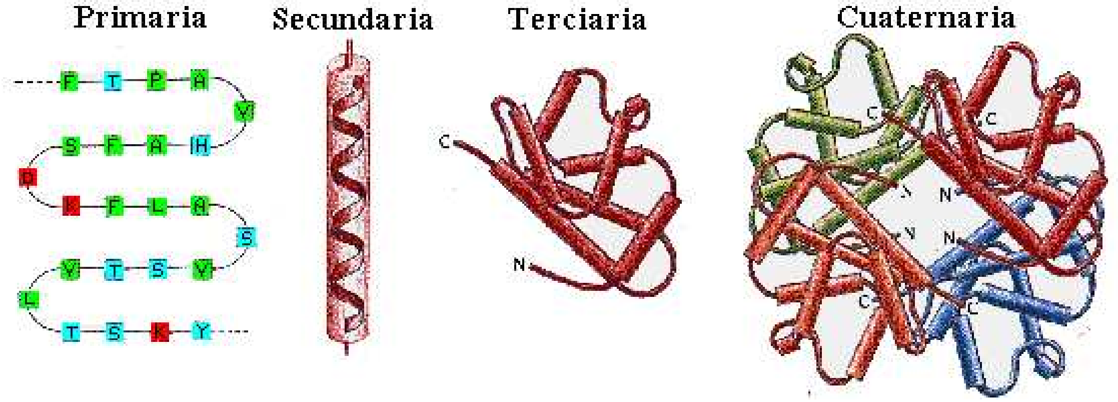


Figura 2.1: Niveles de organización de una estructura proteica. La propia secuencia de aminoácidos recibe el nombre de *estructura primaria*. Las interacciones entre fragmentos contiguos de la secuencia forman estructuras localmente regulares; es lo que se conoce como *estructura secundaria*. La *estructura terciaria* está constituida por el empaquetamiento de dichos elementos estructurales en unidades más o menos compactas denominadas *dominios*. Finalmente, la proteína en su estado final puede contener varios dominios ensamblados en la *estructura cuaternaria*.

produce este plegamiento [1]: Existen cuatro niveles distintos de plegamiento (véase Fig. 2.1), y cada uno de ellos se construye de manera jerárquica a partir del anterior.

La función biológica de una proteína depende en gran medida de las propiedades físicas y químicas de su superficie. La superficie de la estructura de la proteína es la que le confiere su especificidad a la hora de interaccionar con otra(s) macromolécula(s). Como consecuencia de esta especificidad de interacción, la proteína tendrá una determinada función dentro de la célula [1]. El interior de la proteína a menudo es importante únicamente en cuanto que proporciona a la superficie de la molécula la forma y rigidez apropiadas, aunque hay importantes ejemplos de macromoléculas complejas con una estructura interna muy rica, como es el caso de los ribosomas [130, 50].

Es pues evidente que, gracias a un conocimiento de la estructura de las proteínas, o en general de las macromoléculas biológicas, se podría conocer mejor la función que desempeñan dentro de la célula.

2.2.2 El concepto de resolución

Existen muchas definiciones de resolución, pero una de las más empleadas es aquella en la que la resolución se declara como la mínima distancia entre dos puntos que permita aún distinguirlos como entes separados [77].

El orden de resolución de interés en el estudio de las proteínas está localizado entre 1 y 30-50 Å. Una resolución de unos 30 Å proporciona la forma global de una proteína. A unos 7 Å se pueden observar ya algunas estructuras secundarias. Las cadenas de aminoácidos comienzan a distinguirse a resoluciones de 3-4 Å y, finalmente, a 1-2 Å se pueden localizar los átomos individuales.

La cristalografía de rayos X llega a resolución atómica (1-3 Å) para proteínas que formen cristales con un orden intrínseco alto. En cuanto a la microscopía electrónica, ésta cubre un amplio espectro de rangos de resolución, desde aproximadamente los 100 Å para grandes complejos hasta resolución atómica, aunque esta última sólo en algunos casos muy particulares [61, 66] (la resolución alcanzada depende enormemente del tipo de espécimen, del método de preparación empleado, del microscopio, etc.). El hecho de que la microscopía electrónica cubra esta amplitud de resoluciones hace posible que se alcance un continuo en resolución en el estudio de las estructuras biológicas (ver Fig. 2.2). La microscopía electrónica, en ese sentido, es el enlace vital entre la microscopía óptica empleada en Biología Celular y la cristalografía de rayos X [20].

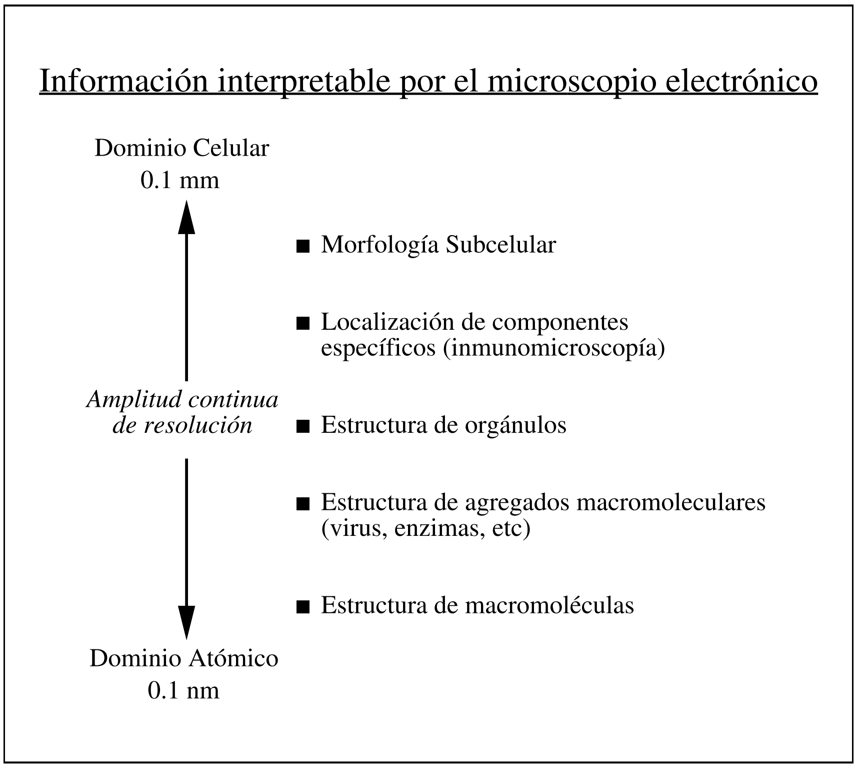


Figura 2.2: Amplitud de aplicación de la microscopía electrónica en el estudio de las estructuras biológicas.

Dado que no se espera que surjan métodos que permitan obtener modelos a resolución atómica para grandes complejos celulares, la dirección que está adoptando la Biología Estructural es la de establecer la organización de los componentes de los procesos biológicos complejos mediante la combinación de datos procedentes de cristalografía de rayos X, espectroscopía RMN, microscopía electrónica y otros métodos biofísicos [2].

2.2.3 Combinación de la microscopía electrónica y el análisis de imagen

La mayor parte del esfuerzo en el campo de la microscopía electrónica se dedica a la interpretación directa de las micrografías (fotografías tomadas con el microscopio electrónico) con el objetivo de obtener medidas cualitativas: forma general, dimensiones, presencia de un componente dado, etc. Esta metodología, aunque interesante en un amplio rango de casos, no extrae toda la información contenida en las imágenes. Hay que tener en cuenta que estas micrografías son, en muchos casos, una versión alterada de la estructura del espécimen.

Por un lado tenemos la preparación del espécimen: las muestras biológicas han de ser tratadas previamente a su exposición en el microscopio electrónico de transmisión (MET), y ello conlleva una alteración morfológica del material. Por otro lado está el proceso de formación de imagen del MET: el MET está lejos de ser un instrumento perfecto y, como consecuencia, introduce una serie de artefactos que hacen que la imagen obtenida no sea una réplica exacta del objeto que está siendo visualizado.

Estos problemas de preparación de las muestras y los efectos introducidos por el MET, que tienen relativa importancia en la microscopía electrónica a baja resolución, han de ser inexorablemente solucionados o compensados cuando el objetivo es llegar a la alta resolución.

Además de estos problemas, hay que añadir el hecho de que el MET no proporciona directamente la información acerca de la estructura tridimensional del espécimen, ello es debido a que el tipo de imágenes que proporciona son bidimensionales, una proyección del volumen, y es necesario la aplicación de técnicas de análisis de imagen para recuperar la tercera dimensión.

En este contexto, las técnicas de procesamiento de imagen juegan un papel crucial en la resolución de los problemas que generalmente presentan las imágenes del MET [43]. El objetivo de estas técnicas es extraer y analizar la información estructural a alta resolución encubierta por estos problemas y, por tanto, constituyen una poderosa herramienta de apoyo a la microscopía electrónica en el estudio estructural de macromoléculas biológicas y su localización en ensamblajes subcelulares.

2.3 El Microscopio Electrónico de Transmisión

El microscopio electrónico de transmisión (MET) es en la actualidad una herramienta indispensable en la Biología y la Bioquímica Estructural, ya que proporciona el medio más directo de visualizar una estructura a nivel molecular.

En su diseño general, el MET (ver Fig. 2.3) es similar al más conocido microscopio óptico, aunque sus dimensiones son mucho mayores y está invertido con respecto a éste. Las lentes son magnéticas y son controladas por la intensidad de la corriente eléctrica que circula por las bobinas de cada lente. Con la lente objetivo se controla el enfoque y con la lente proyectora la magnificación. La fuente de “iluminación” es un filamento o cátodo que emite electrones desde lo alto de una columna cilíndrica de

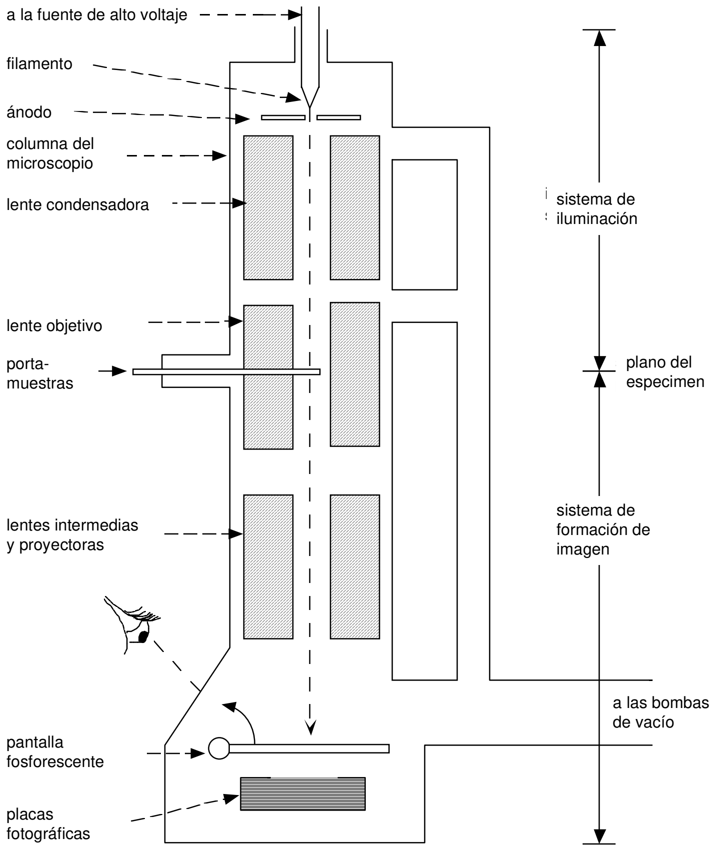


Figura 2.3: Esquema de un microscopio electrónico de transmisión.

unos dos metros de altura. Puesto que los electrones son dispersados por fenómenos de colisión con moléculas de gases, es necesario bombear el aire hacia fuera de la columna para producir vacío. En estas condiciones, los electrones son acelerados desde el filamento mediante un ánodo cercano, pasando por una abertura diminuta formando un fino haz que se dirige hacia la parte inferior de la columna. La muestra (montada en un soporte de unos 3 mm de diámetro) se introduce en el entorno de alto vacío de la columna a través de un sistema de esclusas, y se coloca en la trayectoria del haz, en el centro de la lente objetivo. Algunos de los electrones que pasan a través del espécimen son dispersados de acuerdo con la densidad local del material. Como se explicará más adelante (apartado 2.5), parte de estas dispersiones dan cuenta del contraste de la imagen. Esta imagen se registra, o bien de forma analógica en una placa fotográfica, o sobre una pantalla fluorescente, o bien digitalmente en una cámara CCD.

En general, una radiación determinada no puede ser empleada para resolver detalles estructurales menores que la mitad de su longitud de onda [77]. Esto representa una limitación fundamental de todos los microscopios. Para un microscopio óptico el poder de resolución está limitado por la longitud de onda de la luz visible a unos 0.2 μm [77, 1].

En un MET, el límite de resolución impuesto por la longitud de onda de la luz visible puede ser superado gracias al empleo de electrones en lugar de fotones, ya que los electrones tienen una longitud de onda mucho más reducida. Para un voltaje de aceleración de 100 kV (valor típico en Biología), el límite de resolución sería de 0.002 nm. Sin embargo, las lentes electromagnéticas sufren de aberraciones mucho más difíciles de corregir que las de las lentes ópticas y, como consecuencia, la resolución que se alcanza en la práctica con estos microscopios es, en el mejor de los casos, 0.1 nm (1 Å). Por otro lado, los problemas inherentes a la preparación de las muestras biológicas, su bajo contraste, y el daño que sufren por la radiación electrónica, limitan la resolución para la mayor parte de los especímenes biológicos a un orden de magnitud más (10 Å). Afortunadamente, el empleo de nuevas técnicas de preservación de especímenes biológicos junto con la combinación de datos procedentes de imágenes y de difracción electrónica y, finalmente, la utilización de las técnicas de procesamiento de imagen, están haciendo posible la recuperación de una gran parte del poder resolutivo teórico del MET.

En el MET, la información del espécimen se obtiene a partir de la radiación electrónica que lo ha atravesado. Los electrones tienen un poder muy alto de interacción con la materia y, por tanto, bajo poder de penetración. Consecuentemente, los especímenes que se exponen en un MET han de ser suficientemente finos (típicamente, con un espesor menor de 100 nm) para permitir que los electrones los atraviesen. Por otra parte, un MET estándar tiene una profundidad de foco de varios miles de Å [77]. Como consecuencia de este parámetro y del rango de espesor permitido para las muestras biológicas, las imágenes de MET se forman como superposición de los rasgos estructurales correspondientes a los diferentes niveles de la estructura 3D del espécimen. Tras extensos estudios basados en la teoría de formación de imagen en el MET [58], se ha llegado a la conclusión de que, para aplicaciones biológicas típicas, las imágenes de MET pueden ser consideradas como imágenes de proyección del espécimen.

El problema de obtener las relaciones tridimensionales entre las distintas partes del espécimen a partir de las imágenes que proporciona el MET es, precisamente, el problema de *la reconstrucción tridimensional a partir de imágenes de proyección.* Este es un problema que se encuentra con frecuencia en numerosas disciplinas técnicas, médicas y científicas.

2.4 Preparación y Preservación de las Muestras Biológicas

Hay varios factores a considerar a la hora de elegir un método de preparación de los especímenes para realizar un estudio estructural a media-alta resolución mediante MET. Esto es debido a que las condiciones del MET resultan extremadamente destructivas para la estructura de las muestras.

La técnica de mayor éxito para la media resolución es la llamada *tinción negativa*, en la que el objeto se incluye en una fina capa amorfa de una sal de metal pesado. Esta capa aumenta, por un lado, el contraste de la imagen obtenida y, por otro lado, preserva la estructura del espécimen al protegerla del daño producido por la radiación electrónica del MET. Sin embargo, su uso presenta graves inconvenientes: (i) Limita la máxima resolución obtenida al tamaño del grano del agente de tinción ($\approx$ 20 Å), con lo cual restringe la información a características generales, tal como la morfología global del espécimen. (ii) Sólo es posible obtener información de la superficie de la muestra, siendo impracticable el estudio de la estructura interna. En realidad se reflejan las propiedades del agente de tinción que envuelve a la muestra, y no las de ésta en sí misma.

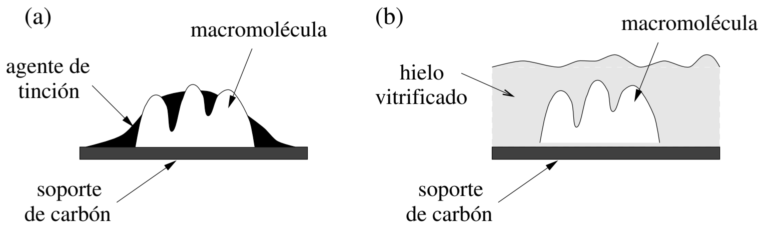


Figura 2.4: Representación esquemática de (a) la tinción negativa y (b) la criomicroscopía.

Cuando se desea obtener información a alta resolución de la muestra biológica, el empleo de un agente de contraste no es adecuado, como se acaba de ver. La microscopía de especímenes no teñidos, que normalmente se realiza a muy baja temperatura y recibe el nombre de *criomicroscopía* [35], resuelve, en principio, los problemas de la tinción negativa. En esta técnica el espécimen se congela rápidamente de forma que su contenido en agua alcanza un estado congelado amorfo (hielo vitrificado) sin pasar por el estado cristalino, que alteraría las estructuras en estudio. En ese mismo estado es expuesto al MET. En la criomicroscopía, la resolución está limitada a la que intrínsecamente posea el espécimen en estudio. Además, permite reconstruir de forma fiable las características estructurales internas de la muestra, no sólo la superficie. Por último, hay que decir que esta técnica conserva el espécimen en estado hidratado, a diferencia de la tinción negativa, lo cual resulta satisfactorio desde el punto de vista de relacionar las imágenes de las muestras con su función biológica en estado nativo.

La acción destructiva de la radiación electrónica sobre la muestra biológica constituye el mayor inconveniente de la microscopía electrónica en lo que se refiere a estudios estructurales. Como consecuencia, y con el objetivo de preservar los detalles de alta resolución, se emplean bajas dosis de radiación. Sin embargo, esto ocasiona que las micrografías obtenidas sean extremadamente ruidosas y la relación señal-ruido sea enormemente pobre. La forma de alcanzar una relación señal-ruido estadísticamente significativa consistiría en disponer de una cantidad suficiente de redundancia estructural. Una de las formas más exitosas de lograr esta redundancia es mediante el empleo de especímenes biológicos que forman ordenaciones cristalinas regulares en dos dimensiones (comúnmente llamados cristales), en las que un gran número de moléculas del espécimen se mantienen con una orientación idéntica y en posiciones espaciadas regularmente. Una imagen de un cristal empleando bajas dosis de radiación no va a mostrar la estructura proyectada de una molécula individual (o celda unidad) directamente, pero, sin embargo, dado que la muestra está compuesta de un número suficientemente grande de celdas unidad, y dada la periodicidad que presentan, estará disponible toda la información necesaria para reconstruir la celda unidad "media".

Empleando la *cristalografía electrónica* (se ha acuñado este término para designar la combinación de microscopía electrónica sobre especímenes cristalinos con la metodología derivada de la cristalografía de rayos X) se han llegado a obtener modelos atómicos de la bacteriorrodopsina (Henderson *et al.*, 1990 [61]) y del complejo clorofílico de recolección de luz (Kühlbrandt *et al.*, 1994 [66]).

2.5 Principio de Formación de Imagen en el MET

El proceso de formación de imagen en un MET es altamente complejo. Aunque existen extensos estudios al respecto (véase por ejemplo Hawkes y Kasper [59]), en la práctica se trabaja con aproximaciones cuya simplicidad, si bien ventajosa, ignora un buen número de fenómenos cuyas magnitudes relativas varían de un espécimen a otro, y que no deben olvidarse por completo a la hora de realizar interpretaciones de las imágenes.

2.5.1 Interacción de los electrones con el objeto

La primera etapa en el proceso de formación de la imagen en un MET es la irradiación de la muestra con el haz de electrones. La mayor parte de los electrones no pasan cerca de ninguno de los átomos de la muestra y, por tanto, lo atraviesan sin sufrir cambios ni en energía ni en fase (electrones no dispersados). Sólo unos pocos electrones interaccionan con los átomos del espécimen. En la mayoría de los casos (para objetos biológicos suficientemente finos) esta interacción es débil (dispersión elástica), y produce un cambio de fase pero no en la energía del electrón. Finalmente, una cantidad mucho menor de electrones interacciona fuertemente con la muestra, perdiendo o ganando energía (dispersión inelástica).

Vemos, pues, que en un MET el contraste se debe a los fenómenos de dispersión electrónica, relacionados con la distribución y características de los átomos que forman el espécimen. La interferencia entre los electrones no dispersados y aquellos que sufren una dispersión elástica débil produce contraste de fase, mientras que la eliminación de los electrones más fuertemente dispersados, mediante la colocación de una apertura en la trayectoria del haz, aporta el contraste de amplitud. La peque na fracción de electrones que sufren dispersión inelástica produce un empeoramiento de la imagen que incide predominantemente en la zona de baja resolución, por lo que es normalmente tolerado, aunque puede interferir con posibles interpretaciones cuantitativas de los datos. Si es posible, esta parte del haz debe eliminarse mediante, por ejemplo, el uso de un sistema de filtración de energía [103].

2.5.2 Función de transferencia de contraste

Como consecuencia de los múltiples estudios que se han llevado a cabo durante muchos años, se ha conseguido formular un modelo general de formación de imagen en microscopía electrónica de transmisión. Esta formulación se ha realizado bajo una aproximación lineal o de primer orden, considerando que las imágenes proporcionadas por el MET son imágenes de proyección, lo cual es cierto para los especímenes finos con los que tratamos. El modelo general de formación de imagen en el MET expresa la relación en el espacio de Fourier entre la imagen y la proyección del objeto [129] como:

$$G(\rho) \; = \; H(\rho)F(\rho) \; + \; N(\rho) \tag{2.1}$$

donde $G(\rho)$ representa la transformada de Fourier de la imagen, $F(\rho)$ es la transformada de Fourier de la proyección del objeto (desconocida) y $N(\rho)$ es la transformada de Fourier del ruido.

El término $H(\rho)$ es lo que se conoce como función de transferencia de contraste (CTF) del MET, la cual representa las distorsiones introducidas por el sistema de formación de imagen del microscopio en cada frecuencia espacial ρ. En general, se han llevado a cabo múltiples estudios bastante profundos acerca de la forma teórica y experimental de la CTF [40, 71, 112, 49, 125, 127, 33, 70, 124], como consecuencia de los cuales se ha llegado a un modelo paramétrico aproximado en el que, para un microscopio dado, las variables más importantes son el desenfoque, el astigmatismo y el porcentaje de contraste de fase en relación al contraste en amplitud (obviando algunos de los efectos atenuantes de la alta resolución que se describen en algunas de las referencias dadas anteriormente).

En la teoría ondular de formación de imagen, los efectos de las aberraciones de las lentes y del desenfoque son modelados por un desfase $\chi(\rho)$ que viene dado en función de la frecuencia espacial por:

$$\chi(\rho) = \pi\lambda \left(\Delta f \, \rho^2 - \frac{1}{2} \, C_s \, \rho^4 \, \lambda^2 \right) \tag{2.2}$$

donde C_s es el coeficiente de aberración esférica del microscopio, Δf es el desenfoque y λ es la longitud de onda de los electrones.

El término que representa la CTF, denotado por $H(\rho)$, refleja las contribuciones de las funciones de transferencia de contraste de amplitud y de fase del microscopio [40] y viene dado por:

$$H(\rho) = -\left(\operatorname{sen}\chi(\rho) + Q(\rho)\cos(\rho)\right) \tag{2.3}$$

Las ecuaciones (2.2) y (2.3) constituyen el modelo matemático de la CTF del microscopio. Estas fórmulas dependen del coeficiente de aberración esférica C_s del microscopio (propio de cada modelo de microscopio), de la longitud de onda de los electrones λ (calculable a partir del voltaje de aceleración empleado) y, finalmente, del desenfoque. Puesto que el coeficiente de aberración esférica y la longitud de onda de los electrones son parámetros dados, el modelo de la CTF del microscopio, de acuerdo con las fórmulas (2.2) y (2.3), está completamente especificado por el desenfoque Δf y la fracción de contraste de amplitud $Q(\rho)$.

La ecuación (2.1) presupone que el haz de electrones que forma la imagen es completamente coherente y monocromático. En la práctica, sin embargo, la fuente de iluminación no es perfectamente puntual, ni el haz es perfectamente monocromático. Debido a esta coherencia parcial, la CTF es atenuada a medida que la frecuencia crece, limitándose de esta forma la resolución. El tratamiento teórico de todos estos fenómenos es bastante complicado y, habitualmente, la influencia de estos efectos se representa mediante un conjunto de funciones envolvente, de forma exponencial, que modulan la CTF. Estas funciones envolvente hacen que la CTF disminuya con la frecuencia espacial, atenuando la región correspondiente a la alta resolución. La función de transferencia de modulación de las micrografías electrónicas [33] y otros efectos que, de forma similar, limitan la resolución [70] también han sido caracterizados por medio de otro tipo de funciones envolvente.

La tesis doctoral de J. Zhu [128] y sus trabajos con J. Frank [129, 130] presentan un estupendo compendio de todas las funciones envolvente que acaban de ser mencionadas en el párrafo anterior. También el capítulo dedicado a *Principios de formación de la imagen en el microscopio electrónico* del libro de J. Frank [50], presenta una excelente revisión de muchos otros aspectos de este campo.

El término $-\operatorname{sen}\chi(\rho)$ de la ec. (2.2) recibe el nombre de *función de transferencia de contraste de fase*. Esta función se caracteriza por alternar zonas de signo positivo y negativo según los valores de la frecuencia espacial. Estas zonas alternadas aparecen en la transformada de Fourier de la imagen procedente de MET como bandas circulares (o elípticas en caso de astigmatismo) brillantes. Las transiciones por el cero implican una inversión de contraste, lo que significa que, dependiendo de la zona de frecuencia espacial, los detalles de la imagen correspondiente a esa frecuencia serán negros cuando debieran ser blancos. Además, esas transiciones por los ceros y su entorno producen que en la imagen no se disponga de información estructural en esos rangos de resolución porque la CTF tiene un valor tan bajo que, al multiplicarse por la transformada de Fourier de la proyección del objeto según la ecuación (2.1), prácticamente se anula la información procedente del espécimen, quedando únicamente información de ruido. La figura 2.5 presenta varios ejemplos unidimensionales de función de transferencia de contraste de fase con y sin envolventes.

El término $-\cos\chi(\rho)$ de la ec. (2.2) recibe el nombre de *función de transferencia de contraste de amplitud* y representa el efecto de la pérdida de electrones en el proceso de formación de la imagen. En muchas aplicaciones se hace la simplificación adicional de despreciar la proporción de contraste de amplitud frente al de fase, tomando pues como CTF la función $H(\rho) = -\operatorname{sen}\chi(\rho)$.

La compensación de los efectos introducidos por el sistema de formación de imagen del MET (es decir, la CTF) es un paso clave para la correcta interpretación de los resultados estructurales. Asimismo, es un proceso crítico y extremadamente importante cuando se quiere realizar un análisis en alta resolución (ver [7, 21], por ejemplo).

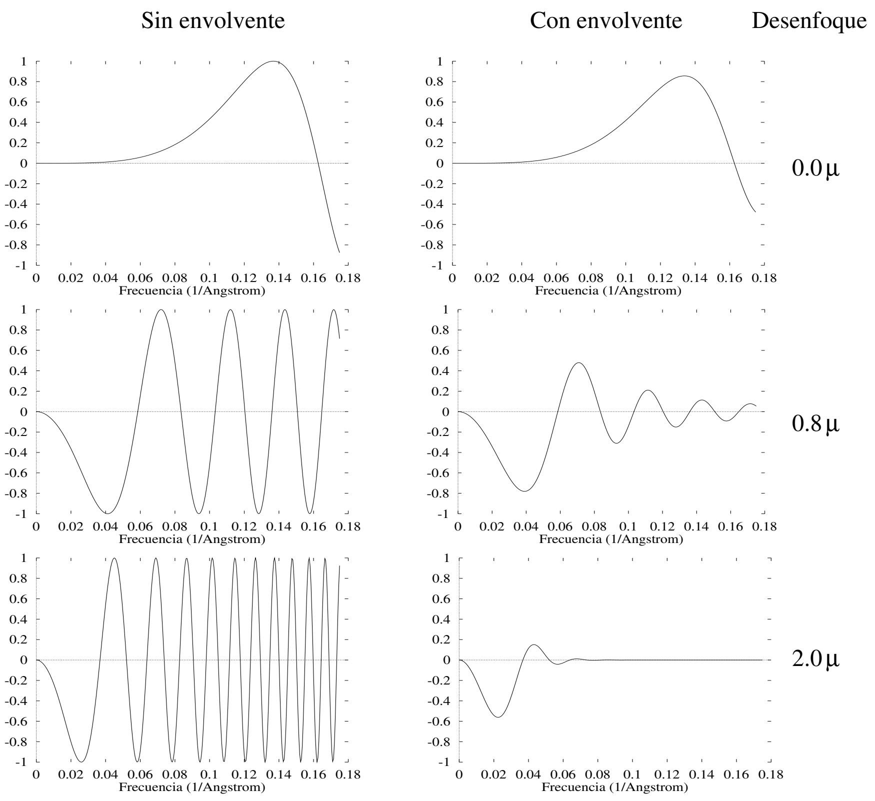


Figura 2.5: Función de transferencia de contraste de fase representada en función de la frecuencia espacial para un microscopio con las siguientes características: $C_s = 5.6\ mm$, $\lambda = 0.037$ Å, tamaño de la fuente $= 0.5\ mrad$, coeficiente de aberración cromática $= 4\ mm$, dispersión de energía $= 1\ eV$. La CTF se representa para varios valores de desenfoque, dados en μm. En la columna de la derecha se muestra el efecto de la envolvente debida a la coherencia parcial del haz.

2.6 Reconstrucción 3D a Partir de Imágenes de Proyección

Como ya se ha mencionado, debido a la gran profundidad de foco del MET, la imagen obtenida representa una proyección del espécimen en el plano perpendicular a la dirección del haz de electrones. Ahora bien, es posible reconstruir la estructura 3D del objeto a partir de una serie de imágenes de proyección tomadas a diferentes ángulos de inclinación y ejes de rotación. La teoría matemática que sustenta esta afirmación fue desarrollada originalmente por Radon en 1917 [94], y a partir de ella se han desarrollado algoritmos de reconstrucción en distintos campos de aplicación, tales como tomografía y radiología en medicina, radioastronomía y microscopía electrónica (para una revisión, véase Herman, 1980 [63]). En este campo, la primera aplicación con éxito de la teoría de Radon fue la reconstrucción de la cola del bacteriófago T4 por De Rosier y Klug en 1968 [28].

El *teorema de Radon* afirma que una función real en un espacio euclídeo n-dimensional que obedece a ciertas condiciones de regularidad está definida unívocamente por sus integrales a lo largo de todos los hiperplanos de dimensión $n-1$. En la aplicación que aquí se trata, esto quiere decir que una función definida en tres dimensiones (el volumen, la estructura) puede ser reconstruida a partir de sus proyecciones bidimensionales (2D). Tradicionalmente, los algoritmos de reconstrucción 3D de un objeto a partir de sus proyecciones 2D se han agrupado en dos grandes familias [50]:

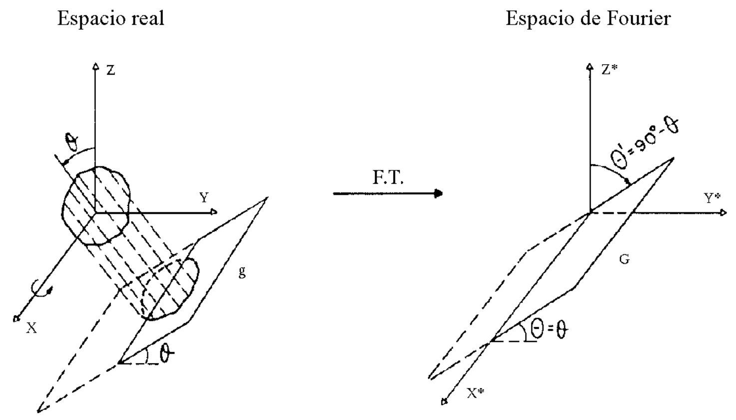


Figura 2.6: El teorema de la sección central. Los ejes de coordenadas en el espacio real están etiquetados con (X,Y,Z), mientras que los del espacio de Fourier lo están con (X^*,Y^*,Z^*). A la izquierda se muestra un objeto en el espacio real localizado cerca del origen de coordenadas. El plano g representa una imagen de proyección obtenida por la irradiación del objeto con haces paralelos inclinados un ángulo ϑ con respecto al eje Z. El eje de inclinación está localizado sobre el eje X. A la derecha se muestra la transformada del objeto extendiéndose sobre todo el espacio de Fourier. La transformada de Fourier del plano de proyección g es una sección G que pasa por el origen de coordenadas del espacio de Fourier (es una sección central), inclinada un ángulo ϑ' con respecto al eje Z^*.

- *Métodos directos*.- Los métodos directos utilizan la relación matemática entre la densidad del objeto y sus proyecciones, y trabajan por tanto en el espacio real. Esta relación puede formularse como un sistema de ecuaciones lineales, pero en la práctica el número de variables es tan enorme que imposibilita que el problema pueda solucionarse de forma directa. Se utilizan, por tanto, técnicas de aproximación iterativas, que tratan de llegar a una solución mediante la minimización de la discrepancia entre las proyecciones de un volumen estimado (que se actualiza en cada iteración) y las proyecciones experimentales. Debido a su alto coste computacional, estos métodos han sido poco empleados en el campo de la microscopía electrónica hasta muy recientemente [83, 72].

- *Métodos de Fourier*.- Estos métodos hacen uso del *teorema de la sección central*. Este teorema establece que la transformada de Fourier de una imagen de proyección 2D es exactamente una sección central de la transformada de Fourier 3D del objeto [78] (véase fig. 2.6). Según este teorema, podemos construir la transformada de Fourier 3D del objeto mediante el ensamblaje de las distintas secciones centrales que proporcionen las transformadas de Fourier de un número suficiente de proyecciones 2D del objeto. Finalmente, la distribución 3D de densidad del objeto puede obtenerse calculando la transformada inversa de Fourier 3D. La figura 2.7 representa un esquema de este principio de reconstrucción 3D.

La separación entre métodos directos y de Fourier no es en absoluto rígida, más aún considerando la dualidad formal entre el espacio real y el de Fourier. La *retroproyección pesada* [90], que se considera una tercera clase de métodos de recontrucción [50], es el método empleado con más frecuencia en el campo de la microscopía electrónica y está basado el Teorema de la Sección Central pero se aplica en el espacio real (una demostración del algoritmo de retroproyección pesada puede verse en [126] cap. 27).

2.7 Geometrías de Adquisición de datos

Hemos visto en el apartado anterior cómo, si se obtienen proyecciones 2D de un objeto desde varias orientaciones, se puede, en principio, reconstruir su estructura 3D. En la práctica, las diferentes proyecciones se obtienen inclinando el objeto en el microscopio con la ayuda del goniómetro del microscopio electrónico. Obviamente, la situación ideal es aquella en que las proyecciones cubren todo la amplitud angular. Sin embargo, existen algunas limitaciones técnicas respecto al máximo ángulo que se pude alcanzar y que normalmente esta en el entorno de $\pm 60°$. Esto es debido al soporte donde se sitúa la muestra [57] [sec. 4.3.3(d)], su grosor efectivo crece proporcionalmente a 1/cos(ángulo de inclinación).

Así pues, el problema general en microscopía es la reconstrucción tridimensional de una estructura a partir de un conjunto finito de proyecciones bidimensionales

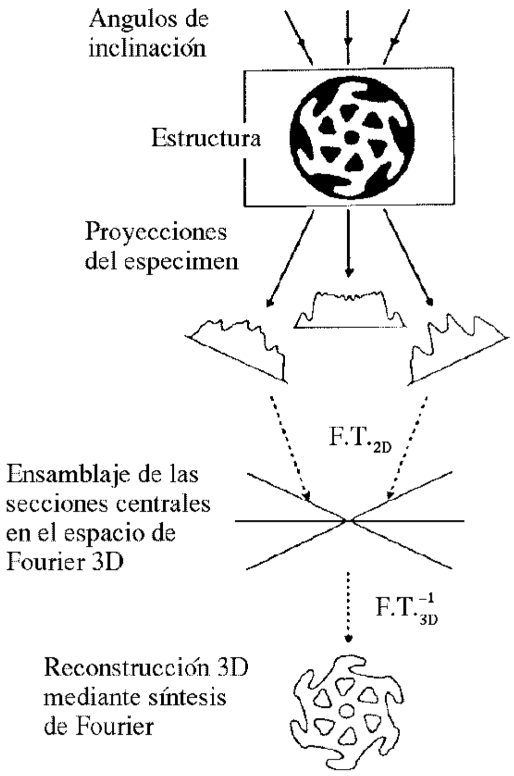


Figura 2.7: Principio de la reconstrucción tridimensional. Las proyecciones 2D del objeto tomadas a diversos ángulos proporcionan distintas secciones centrales de la transformada de Fourier del objeto. Si el número de proyecciones es suficiente, la transformada de Fourier puede completarse mediante un proceso de interpolación. El objeto original puede obtenerse entonces tras una transformada inversa de Fourier.

ruidosas, que abarcan una amplitud angular restringida. Los efectos de estas limitaciones espaciales se observan claramente utilizando el formalismo del espacio de Fourier, y dependen del tipo de geometría de recolección de datos utilizado. Dos son los tipos principales de geometrías utilizados habitualmente: la geometría de eje único ("single axis tilting") y la geometría cónica ("conical tilting").

2.7.1 Geometría de eje único

En este sistema de adquisición de los datos el objeto se inclina en torno a un eje fijo en pequeños incrementos, y en cada orientación se toma una micrografía (ver Fig. 2.8). Calculando la transformada de Fourier de cada proyección y utilizando el teorema de la sección central, se obtiene un conjunto de planos en el espacio de Fourier que tienen un eje en común. Como puede observarse, en este caso el cálculo de la transformada de Fourier 3D del espécimen puede descomponerse en una colección de problemas 2D. Además, en la Fig. 2.8 también se aprecia que existe una región de la transformada 3D, llamada cuña perdida, donde no se recogen datos.

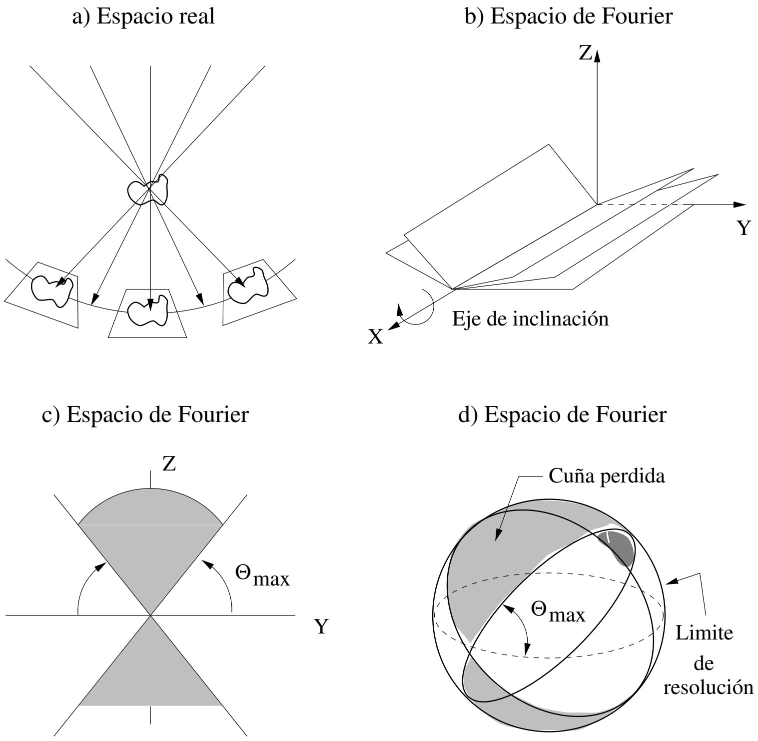


Figura 2.8: Geometría de recolección de datos de eje único. (a) Inclinación en el espacio real en torno a un eje perpendicular al plano de la página. (b) Representación en el espacio de Fourier de la información presentada en (a), donde las diferentes proyecciones son planos centrales de la transformada de Fourier del objeto con el eje de inclinación (X) en común. El eje Z se corresponde con el eje óptico del microscopio, y es perpendicular al plano de la muestra sin inclinar. (c) Vista de (b) desde el eje de inclinación, mostrando una cuña de información perdida alrededor del eje Z. (d) Representación espacial en el espacio de Fourier en una situación práctica con un ángulo máximo de inclinación $\Theta_{\max}$ y resolución limitada.

2.7.2 Geometría cónica

Una serie de imágenes relacionadas mediante una geometría cónica se realiza inclinando la muestra al máximo ángulo posible $\theta_{\max}$ y girando en el plano de inclinación a pequeños incrementos, tomando una micrografía en cada posición. También, como en el caso anterior, hay una región indeterminada de la transformada 3D, esta vez con forma de cono (ver Fig. 2.9).

Una variante interesante de este sistema de recolección de datos es la geometría cónica aleatoria ("random conical tilting"), que por su especial incidencia en el campo de la reconstrucción 3D de partículas individuales se describe en la siguiente sección.

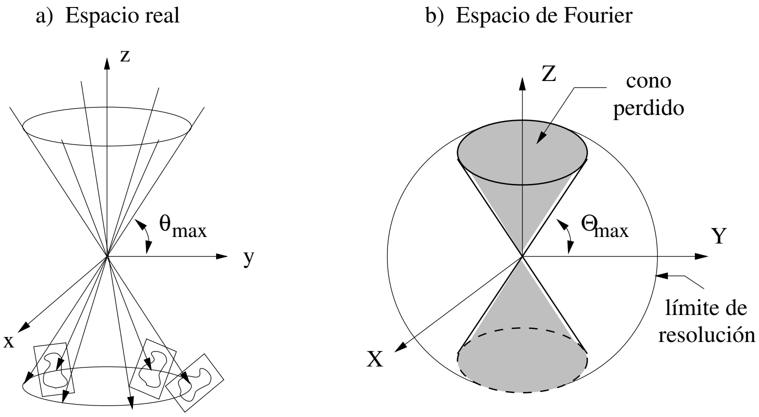


Figura 2.9: Geometría cónica de recolección de datos. (a) Inclinación de un ángulo $\theta_{\max}$ en el espacio real alrededor de múltiples ejes en el plano xy. (b) Representación en el espacio de Fourier de la información recogida en (a); obsérvese que no hay datos experimentales en un "cono perdido" alrededor del eje Z, que, como en el caso anterior, se corresponde con el eje óptico del microscopio y es perpendicular al plano de la muestra sin inclinar.

2.8 Reconstrucción Tridimensional de Partículas Individuales

En el campo de la microscopía electrónica existen diversos métodos para la reconstrucción de especímenes biológicos a partir de sus proyecciones. La situación más simple la presentan las macromoléculas que se agregan en estructuras helicoidales. En este caso, una sola vista aporta información suficiente para reconstruir el volumen hasta una cierta resolución (DeRosier y Klug, 1968 [28]; DeRosier y Moore, 1970 [29]). Existen otros tipos de macromoléculas biológicas que poseen también un alto grado de simetría: este es el caso, por ejemplo, de los virus icosaédricos, donde una sola vista del espécimen proporciona otras 59 (Crowther et al., 1970 [23]); o de los cristales bidimensionales, que presentan simetría translacional (Henderson y Unwin, 1975 [62]). Es obvio, sin embargo, que en un planteamiento general no puede contarse con simetrías características de patrones de agregación específicos, y que debe considerarse la problemática de especímenes individuales y asimétricos. En este último caso se habla de partículas individuales: esto es, partículas aisladas, orientadas aleatoriamente, y, en principio, con estructura idéntica. Hasta mediados de los años 80, era prácticamente imposible utilizar este tipo de muestras en reconstrucción tridimensional. Solamente los objetos simétricos o cristalinos permitían el empleo de imágenes tomadas con dosis mínimas de electrones, ya que en estos sistemas la proyecciones pueden ser mejoradas aprovechando la repetición del motivo unidad o la simetría del espécimen. Aun en la actualidad estos dos métodos siguen produciendo las reconstrucciones de mejor calidad, pero se encuentran en desventaja de cara a su aplicación general ante la existencia de numerosas macromoléculas biológicas de interés que no pueden cristalizarse y no son simétricas.

El único modo de recoger una cantidad suficiente de datos para la reconstrucción

3D de una muestra no cristalina y asimétrica en condiciones de dosis mínima consiste en combinar imágenes de un gran número de partículas. Dos son las técnicas que se han diseñado para reconstruir partículas individuales (salvo casos especiales, como la presencia de geometría cilíndrica [48]): *la reconstitución angular* y el *esquema cónico-aleatorio*. La más comúnmente utilizada es la segunda.

2.8.1 Reconstitución angular

El método de reconstitución angular fue desarrollado por van Heel [118]. Aquí los valores de los ángulos para cada proyección son asignados a posteriori basándose en las líneas comunes [23].

Las *líneas comunes* son una consecuencia directa del teorema de la sección central (ver Fig. 2.6) El concepto de las líneas comunes será muy usado en los capítulos siguientes de esta memoria. Se usará ahora un modelo simple: dos proyecciones arbitrarias de un objeto representadas en el espacio de Fourier 3D por sus secciones centrales asociadas [ver Fig. 2.10(a)]. Estas se intersectan sobre una línea a través del origen, su *línea común*.

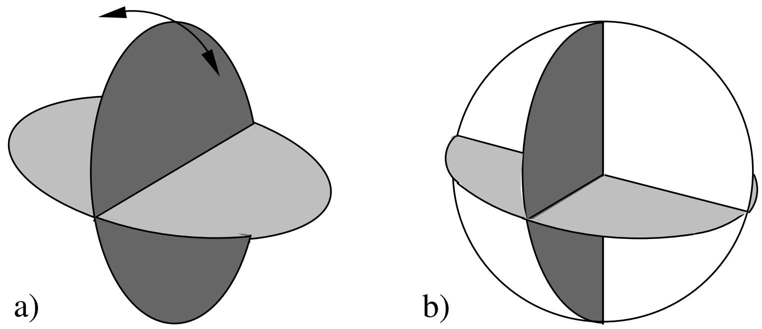


Figura 2.10: Principio de la reconstitución angular: (a) Dos proyecciones de un objeto 3D corresponden a dos secciones centrales (a través del origen) en el espacio 3D de Fourier, (b) Introduciendo una tercera sección central en el sistema de la Fig. (a), resulta en dos lineas comunes más y fija la orientación relativa de todas las tres secciones centrales.

Una vez que la línea común es encontrada, se fija la rotación (en el plano) de dos secciones centrales de Fourier (y, por lo tanto, de las dos proyecciones correspondientes). Obsérvese que las dos secciones centrales aun pueden moverse, alrededor de la línea común fijada (movimiento que puede visualizarse como el de una “bisagra”), esto es, la línea común encontrada no fija completamente la orientación espacial de las dos proyecciones, debido a que todavia existe una rotación relativa de las secciones centrales alrededor del eje común. Introduciendo una tercera proyección y su sección central, que no sea paralela a ninguna de las otras dos [ver la Fig. 2.10(b)], fijará el movimiento ”de la bisagra”, y resultará en la completa determinación de los ángulos relativos entre las tres proyecciones (aunque queda una ambigüedad en cuanto a la mano del objeto: las dos soluciones posibles, la combinación de mano derecha o de izquierda de las secciones centrales, son matemáticamente correctas). Partiendo de este sistema de orientaciones, nuevas projecciones pueden adicionarse combinandolas con pares de proyecciones ya fijadas. Esto es, en esencia, el método de reconstitución angular.

Aunque el método descrito en el parrafo anterior es elegante en su planteamiento conceptual, en la práctica no puede usarse para alinear imágenes individuales sin promediar, debido a la baja razón señal-a-ruido de estas imágenes. Si es aplicable a aquellas situaciones en que las imágenes a alinear tengan una relativamente alta relación señal-ruido, como sería el caso de partir de imágenes promediadas. Sin embargo, el promediado de imágenes puede implicar una pérdida de resolución al realizar el promediado entre imágenes con similares —pero no idénticas— orientaciones. La adición de un método de refinamiento, descritos en los capítulos siguientes, puede lograr la resolución completa de los datos ([50] p. 195).

El requerimiento inexcusable para la aplicación de este método a la reconstrucción tridimensional de macromoléculas asimétricos es que ésta exhiba múltiples vistas (al menos tres, puesto que es el número mínimo para para la determinación relativas de sus orientaciones, y que no estén relacionadas por la rotación alrededor de un solo eje).

Ejemplos de la aplicación de este método, con criomicroscopia, pueden verse en [34].

2.8.2 Esquema cónico aleatorio

Esta técnica fue desarrollada en 1988 por Radermacher y colaboradores [89]). Mediante su uso se consigue una serie de imágenes de proyección que no pertenecen estrictamente a la misma molécula, sino a muchas moléculas idénticas (en principio) en distintas orientaciones, evitando así el problema de una irradiación excesiva. Su uso requiere el cumplimiento de una única ligadura: que las macromoléculas se depositen en la rejilla en una sola o unas pocas vistas preferentes. Aunque este requerimiento puede parecer restrictivo, es muy habitual que las macromoléculas biológicas interaccionen con la película soporte produciendo una serie limitada de vistas.

Cuando una muestra de partículas individuales se inclina un ángulo θ, las proyecciones contenidas en una micrografía se corresponden con muchas vistas diferentes, cuya orientación debe ser determinada con precisión. Este conjunto de vistas es formalmente equivalente a una serie cónica de inclinación con variaciones aleatorias del ángulo azimutal (ver Fig. 2.11). Para calcular este ángulo se toman dos micrografías de cada área de interés. La primera, inclinada θ grados, proporciona las vistas que se utilizarán en la reconstrucción 3D, mientras que la segunda, tomada sin inclinar, se utiliza para calcular los ángulos azimutales mediante técnicas de correlación cruzada [101]. Esto es, la micrografía sin inclinar proporciona la información necesaria sobre la geometría exacta de recolección de los datos; esta información se aplicará al combinar las proyecciones inclinadas en la reconstrucción 3D. En este método esta implícita la presunción de que todas las proyecciones derivan del mismo tipo de objeto tridimensional, o de que es posible clasificarlas previamente para que así sea.

Sin embargo, la metodología descrita en el parrafo previo produce una estimación menos que perfecta de los parámetros geométricos de las proyecciones con respecto al sistema de coordenadas fijas al volumen por varias razones: El ángulo de inclinación realmente no es idéntico para todas las partículas, sino que está dentro de un amplio margen que puede ser tanto como 20°, dado que el soporte no es completamente plano; los ángulos de rotación son determinados a partir de las imágenes sin inclinar, las cuales ya has sido previamente expuestas (miestras se tomó la imagen inclinada) al haz de electrones y pueden haber sido dañadas; la localización del origen de las proyecciones con respecto al sistema de coordenadas supuesto (ligado a las partículas) puede presentar serios problemas, algo que ciertamente ocurre al aplicar métodos como los descritos en [89, 13] a ciertos tipos de grandes macromoléculas. Adicionalmente, la clasificación de las partículas en base a su vista a 0° puede dar resultados ambiguos, toda vez que en algunos casos los métodos de clasificación no consiguen realizar una separación precisa entre las diferentes vistas de un mismo objeto (ejemplos típicos serían imágenes muy ruidosas de criomicroscopía, por las pequeñas variaciones de la orientación en torno a la vista dominante, ó especímenes que generan imágenes de proyección parecidas a distintas orientaciones).

Estas limitaciones pueden ser resultas en gran medida con el uso de un método de refinamiento. La esencia de tal método es el siguiente: tener un primer volumen estimado, calcular los parámetros geométricos necesarios por la consistencia de las proyecciones inclinadas con las estimadas, reconstruir con los nuevos parámetros y repetir hasta la convergencia. Dos de estos métodos se han desarrollado al mismo tiempo y ya se han utilizado con éxito a partir de imágenes individuales sin promediar: el método de comparación directa de las proyecciones en 3D [84] y el método de la transformada 3D de Radon [91].

Para partículas individuales, la reconstrucción se lleva a cabo habitualmente en el espacio real utilizando la retroproyección convolucionada ("weighted backprojection") [90], o técnicas algebraicas de reconstrucción (ART, por sus siglas en inglés) [72].

Ejemplos de la aplicación del esquema cónico-aleatorio, usando criomicroscopía, pueden verse en [69, 68, 6, 25]. Por último, una revisión en detalle del esquema cónico aleatorio puede encontrarse en el libro de J. Frank [50].

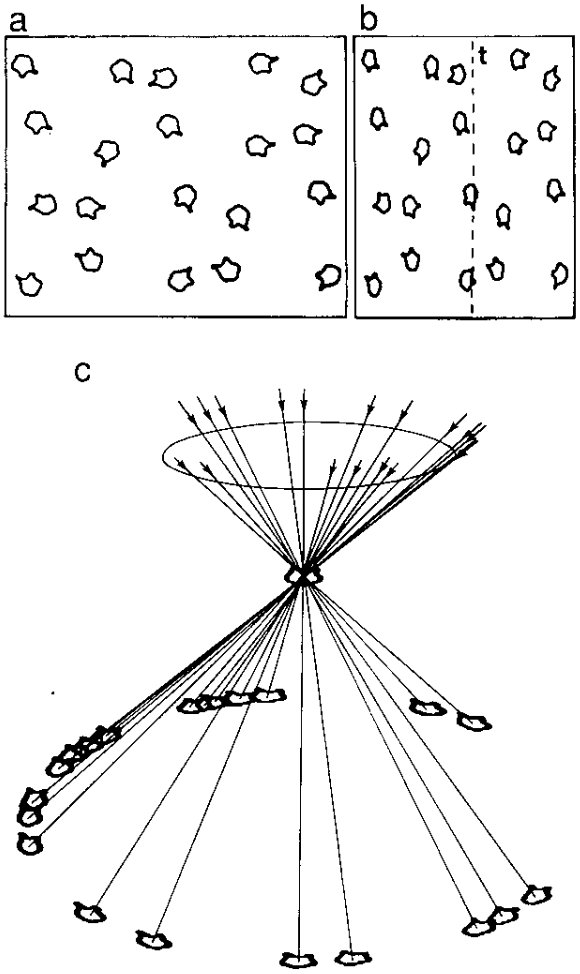


Figura 2.11: Ilustración del principio básico del esquema de reconstrucción cónico aleatorio. (a) Vista de una muestra con partículas orientadas al azar. (b) Proyección de la misma muestra inclinada. La línea punteada representa el eje de inclinación. Las imágenes extraídas de (b) forman la serie cónica de inclinación que se muestra en (c), equivalente a una serie de inclinación de una sola partícula con direcciones de proyección aleatorias, todas ellas situadas en la superficie de un cono.

" ... it is much easier to have some vague notion about any subject, no matter what, than to arrive at the real truth about a single question, however simple that may be."

—René Descartes (1596-1650), in his 1629 text, *Rules for the Direction of the Mind.* This mathematician-philosopher set precedents for mathematical notation, such as reserving the last letters of the alphabet for unknown quantities and the firts letters for known quantities.

Tomado de SPECTRUM, Ago. 1997, p. 15.

Capítulo 3

Métodos de Refinamiento de la Geometría

3.1 La Necesidad del Refinamiento

Las suposiciones básicas cuando se realiza una reconstrucción tridimensional son que los ángulos de Euler, que describen la orientación de cada una de las proyecciones bidimensionales con respecto a un mismo marco de referencia, y que las dos coordenadas de la localización, sobre las proyecciones bidimensionales, del origen del sistema de coordenadas del volumen, son conocidos correctamente. Estas suposiciones son ciertas en las reconstrucciones tridimensionales medicas [63], ya que en estas aplicaciones todas las proyecciones son tomadas de un solo volumen, el paciente, y el sistema de coordenadas del volumen puede asumirse que está fijo al instrumento que recoge los datos. Desafortunadamente, en el caso de aplicaciones de microscopía electrónica estas suposiciones no son ciertas —los ángulos de Euler y el origen del sistema de coordenadas son conocidos sólo indirectamente y con una precisión limitada—. Para incrementar ésta precisión podría pensarse tanto en colectar una cantidad mucho más grande de partículas de las que se están usando, como también en ser más restrictivos durante el paso de clasificación, esperando encontrar grupos de imágenes realmente homogéneos para ser usados en el proceso de reconstrucción 3D. Sin embargo, éstas aproximaciones, que posiblemente nos llevaría a una reconstrucción de alta calidad, son difíciles de aplicar en la práctica en el caso de especímenes empotrados en hielo. Esto es así por los problemas adicionales que se presentan: los muy altos niveles del ruido, la variabilidad intrínseca del espécimen, las inexactitudes en la asignación de los ángulos de Euler a muchas de las imágenes, así como problemas del alineamiento de las partículas a 0°, esto por mencionar solamente algunos. Por lo tanto, parece más razonable asumir que los ángulos de Euler iniciales sólo son conocidos aproximadamente y que es necesario desarrollar un procedimiento para refinarlos.

El *refinamiento angular* es, por tanto, el proceso por el que se da a cada proyección una oportunidad de "encontrar una mejor colocación" en términos de su orientación y origen de fase. Este proceso está demostrado que mejora la resolución de la reconstrucción sustancialmente. Para el caso del ribosoma 70S un mejoramiento, partiendo de 1/47.5 Å, y alcanzando 1/40 Å fue presentada por Penzeck *et al* [84].

3.2 Métodos Basados en un Prototipo

Las técnicas existentes de refinamiento angular esencialmente están basadas en el mismo principio: usan una reconstrucción 3D existente en baja resolución como un *prototipo*. Teniendo este primer volumen estimado, el prototipo, se calculan los parámetros geométricos necesarios por la consistencia de las imágenes de proyección con las proyecciones a ángulos conocidos del volumen estimado, se reconstruye con los nuevos parámetros y se repite el proceso hasta la convergencia.

Es necesario que el método que se use para para la corrección de los parámetros sea robusto, puesto que tiene que funcionar para las imágenes típicas de criomicroscopía electrónica que presentan un SNR (Signal-to-Noise Ratio) menor que 1.

3.2.1 Métodos de comparación directa

El *esquema de comparación directa de las proyecciones* desarrollado por Penczek *et al* [84] es iterativo: Las proyecciones de referencia (reproyecciones) son computadas a partir de una estructura prototipo (a baja resolución) de tal forma que cubran todo el espacio angular regularmente (equiespaciadamente). Como estructura prototipo se usado un volumen ya existente, el cual podría haber sido el resultado de unir varias grupos de proyecciones cónicas-aleatorias [6, 25]. Una proyección experimental dada es correlacionada con todas los proyecciones de referencia. El pico más grande de la Función de Correlación Cruzada (CCF en sus siglas en inglés) da los valores de los ángulos a la proyección. La CCF entre la proyección de referencia y la proyección actual da, al mismo tiempo, el desplazamiento y la orientación azimutal de la proyección de la partícula bajo consideración. Usando los nuevos parámetros para cada proyección experimental, una nueva reconstrucción es calculada, la cual normalmente tiene una resolución mejorada. La nueva reconstrucción es usada como un nuevo prototipo, etc.

El *esquema de comparación directa de las proyecciones* descrito arriba también ha sido aplicado al caso de virus [19]. Ambos esquemas tienen similitudes con técnicas de comparación directa formuladas hace tiempo, cuando las técnicas de búsqueda exhaustiva con ordenador eran bastante consumidoras de tiempo y por lo tanto aún imprácticas. Estos vínculos serán resumidos a continuación:

(i) van Heel [117] propuso un método para obtener la orientación de una proyección usando la siguiente secuencia:

Paso 0:

Asignar orientaciones aleatorias a las proyecciones.

Paso 1:

Calcular la reconstrucción 3D.

Paso 2:

Proyectar la reconstrucción 3D en todas las direcciones en el espacio para comparar las proyecciones experimentales. Los parámetros a los cuales se ajustan mejor las comparaciones resultan en las nuevas orientaciones para los proyecciones; calcular las diferencias entre las proyecciones del modelo actual y las proyecciones experimentales; si la suma del cuadrado de las diferencias es más grande que un valor predefinido entonces VOLVER al paso 1, de otro modo PARAR.

Este procedimiento contiene los ingredientes principales del algoritmo de Penczek *et al.*, excepto por el cambio en el punto de arranque (paso 0), lo que resulta en la práctica es que el algoritmo encuentra difícilmente una solución satisfactoria excepto en el caso, totalmente fortuito, donde el asignamiento aleatorio de los ángulos genere una distribución cercana a los valores correctos. En todos los otros casos, la reconstrucción inicial no se parecerá a una versión en baja resolución de la estructura real, y fallará en conducir los nuevos asignamientos de la orientación hacia la dirección correcta.

(ii) La aproximación de *Harauz y Ottensmeyer* [54, 55] difiere de los discutidos hasta ahora ya que usa un modelo generado por computadora de la estructura predicha mas que una reconstrucción experimental como un prototipo 3D inicial, y usa también una combinación de análisis visual y computacional para obtener una ajuste óptimo para cada proyección. Debido al pequeño tamaño del objeto (centros de nucleosomas cuya señal de fósforo fue obtenida por filtración de energía) y el tipo de preparación del espécimen (secado al aire) el resultado ha sido acogido con gran escepticismo. Hay también una cuestión importante en el uso de una referencia 3D impuesta (opuesto a una referencia 3D experimental) que podría influenciar, o nó, en el resultado. Hay casos publicados acerca de promediados inducidos-por-la-referencia en el caso bidimensional [92, 4].

Otro método similar, aplicado al caso de cristales, ha sido desarrollado por Dengler [31].

3.2.2 Método de la Transformada de Radon

En principio este método es totalmente similar al método de comparación directa de las proyecciones descrito anteriormente. Aunque se distingue por su forma elegante de formular el problema y que le permite poner de manifiesto ciertas relaciones que no son evidentes en las otras aproximaciones.

El método de la transformada de Radon, (desarrollado por Radermacher [91]) compara la transformada 2D de la proyección experimental con la transformada 2D de Radon de la proyección de referencia, extraída esta última de la transformada 3D de Radon del volumen usado como prototipo.

La comparación es realizada por correlación cruzada de dos transformadas 2D de Radon que permite hacer el alineamiento translacional (equivalente a “poner en fase” en el espacio de Fourier) simultáneamente.

La transformada 2D de Radon de una imagen es la versión discreta de la transformada general de Radon y es conocida también como *senograma* [118].

La correlación cruzada entre senogramas ya fue usada por van Heel [118] para la *reconstitución angular* (ver la sec. 2.8.1) como un medio para determinar las orientaciones relativas entre dos o más proyecciones aisladas. La aplicación de este procedimiento a tales imágenes sin promediar falla debido a la baja señal ruido de tales imágenes, a menos que presenten una alta simetría (como en al caso de los virus icosaédricos). Sin embargo, el método sí es viable cuando se aplica para comparar una proyección *experimental* con una proyección *promediada* (obtenida a partir del prototipo), así como a la comparación de una proyección promediada con otra (ver [104, 34]).

Este método será tratado en detalle en el capítulo 5.

3.3 Métodos No Basados en un Prototipo

Un método de refinamiento no basado en un prototipo debiese resolver el problema siguiente: determinar *de forma precisa* las orientaciones relativas de las proyecciones 2D del objeto 3D a reconstruir. Esto es, asignar de forma precisa los valores de los parámetros a las proyecciones *a posteriori*.

El problema de la determinación de los ángulos relativos de proyecciones de objetos 3D tiene dos soluciones: el método de momentos y las aproximaciones por líneas comunes. La primera [53, 99], basada en teoría de momentos generales, es altamente sensible a errores en los datos, por lo que no es aplicable en casos reales de microscopía electrónica. La segunda solución, las aproximación por líneas comunes, están basadas en el teorema de la sección central, el cual nos indica que en el espacio real dos proyecciones cualesquiera de un objeto 3D comparten una proyección 1D común. La formulación equivalente en el espacio de Fourier dice que las transformadas de Fourier de dos proyecciones cualquiera de un objeto 3D tienen secciones centrales en el transformada de Fourier 3D del objeto, y por la tanto una línea común en el espacio de Fourier. Las direcciones conocidas de tres de líneas comunes fijan la orientación de tres planos (ya sea en el espacio real o en el recíproco) y son suficientes para la determinación de las orientaciones de las tres proyecciones asociadas (excepto en una ambigüedad en la mano en la reconstrucción 3D). Por lo tanto, en principio, uno podría necesitar sólo tres proyecciones de un objeto 3D para calcular los ángulos requeridos. Esta asignación de valores *a posteriori* es posible realizarla por el método de *reconstitución angular*, cuyo *núcleo* está sostenido por la triangulación de líneas comunes entre tripletas de proyecciones (ver también la sección 2.8.1). Sin embargo, la reconstitución angular no realiza una asignación *precisa* debido a que el método funciona sobre imágenes promediadas a partir de clases que pueden tener una variabilidad interna apreciable. "Pero por el momento es imposible encontrar la línea común entre dos imágenes de baja-dosis. Puede resultar mejor si se cuenta con simetría, pero el mejoramiento en el SNR puede ser insuficiente si la simetría no es alta"(Radermacher 1996, comunicación personal).

Así, un método que utilice líneas comunes para buscar las orientaciones entre proyecciones debe minimizar los errores en los valores de los parámetros asociados, pero no refinará estos valores como los métodos basados en prototipo. Sin embargo este tipo de esquemas plantean una alternativa para hallar el prototipo inicial, y puede plantearse entonces, cómo su objetivo, la búsqueda *del mejor* volumen de referencia que resuelva el problema de como iniciar el proceso iterativo en la aplicación de un método de refinamiento con prototipo.

Farrow y Ottesmeyer [41] realizaron una elaboración interesante de la aproximación por líneas comunes. Ellos reconocen que aún bajo condiciones modestas de ruido el error promedio en la posición del eje común llega a ser inaceptablemente grande. Para incrementar la exactitud del método ellos usan un número grande de proyecciones y usan matemáticas de cuaterniones [56] para encontrar una solución consistente y minimizar los errores. Aunque los autores han intentado mejorar el rendimiento del método de las líneas comunes mediante la inclusión de múltiples comparaciones de líneas comunes, fueron incapaces de incluir todas las posibles correlaciones. Otra desventaja del método es que las orientaciones de las líneas comunes y las discrepancias correspondientes son calculadas solamente una vez, durante el primer paso del método. Todos los refinamientos de la solución son hechos forzando la consistencia de la solución, sin recalcular el error correspondiente en el espacio de líneas comunes. Sin embargo, el método parece ser lo suficientemente robusto como para ser aplicado directamente a imágenes obtenidas en un microscopio electrónico de transmisión con barrido [24].

La técnica de reconstitución angular realiza una aproximación de *fuerza bruta* para el problema de la inclusión de proyecciones adicionales. Después de que las orientaciones de tres proyecciones (promedio) han sido determinadas, proyecciones adicionales son incluidas usando búsquedas exhaustivas. En el curso de todas estas búsquedas, se intentan todas las posibles orientaciones de la nueva proyección mientras sirven como referencia los datos cuya orientación ya ha sido fijada. Eventualmente, una vez que se ha encontrado la orientación para un número suficientemente grande de proyecciones, una reconstrucción 3D inicial puede ser calculada. Y, como ya se ha sugerido, aquí se podría aplicar el proceso de refinamiento, usando esta reconstrucción 3D como prototipo.

La desventaja principal de estos esquemas es que si la orientación del primer grupo de tres proyecciones es incorrecta, la estructura final será probablemente incorrecta. Y si la primera solución está sesgada de una cierta manera, este sesgo podría ser propagado y amplificado en este proceso. Vale la pena indicar en este contexto que la solución implícita para el problema de la orientación para tres proyecciones es altamente no lineal. Los métodos ya mencionados usan la detección del máximo de la llamada *función de correlación de senogramas* [118] calculada entre pares de imágenes representadas como senogramas (o la transformada de Radon discreta de las proyecciones 2D). Dada la no linearidad de las ecuaciones involucradas, aun un pequeño error en la localización del máximo puede trasladarse a un error grande en los ángulos derivados.

Para superar las limitaciones descritas, el método podría beneficiarse si se resolviese el problema de la orientación *simultáneamente* para cualquier número de proyecciones más grande que tres. Aún si el SNR es bajo para encontrar una solución para tres proyecciones, el incremento en el número de proyecciones de entrada debería incrementar el SNR efectivo y facilitar la búsqueda de la solución.

El método de *minimización simultánea* propuesto muy recientemente por Penczek *et al* en 1996 [85] apunta hacia una solución para esta cuestión. Esta aproximación ya ha sido aplicado para imágenes del ribosoma 70S de E. coli provenientes del microscopio electrónico con filtración de energía, con resultados aún bajo evaluación.

En esta tesis de propone un método semejante aplicable al esquema cónico aleatorio (Capitulo 6, sec. 2) que intenta un primer paso de refinamiento de la asignación inicial de valores de los ángulos. También se propone un método exclusivo para obtener el mejor asignamiento inicial en los desplazamientos (o el mejor origen de fase, en el espacio recíproco) (Capítulo 6, sec. 1).

Además se presenta un método para encontrar la mejor asignación inicial en el ángulo de inclinación para el caso de tener una partícula con simetría cilíndrica (Capitulo 6, sec. 3).

"1972 Dennis Ritchie develops C at Bell Labs, so named because its predecessor was named B."

Tomado de COMPUTER, Oct. 1996, p. 81.

Capítulo 4

Método de Refinamiento por Comparación Directa de las Proyecciones

4.1 Introducción

En este capítulo se describirá el *método de comparación directa de las proyecciones* presentado por Penczek [84]. Éste método utiliza una estructura 3D ya obtenida como una referencia para refinar los parámetros de translación y orientación de las proyecciones usadas. El método, por lo tanto, combina dos procesos: (1) *El refinamiento de los parámetros de translación* y (2) *El método de refinamiento de la orientación*

El paso (1) es bastante sencillo: la estructura de referencia, el prototipo (ver la sec. 3.2 en el capítulo anterior), es proyectada en la dirección de las proyecciones originales, y usando funciones de correlación cruzada, se corrigen los parámetros de translación. Seguido, se aplica el paso (2) que realiza el refinamiento de los parámetros de rotación. El paso (2) es mucho más complejo y será desarrollado en la sección siguiente. Una vez que se tienen corregidos los párametros de la proyecciones originales, se calcula una nueva reconstrucción 3D refinada, la cual es usada subsecuentemente como la estructura de referencia para la siguiente iteración del método.

Usando el mismo principio del método de refinamiento por comparación directa de las proyecciones, resulta muy fácil incluir los datos de proyecciones adicionales en el proceso. El cálculo del coeficiente de correlación cruzada entre la proyección cálculada a partir del prototipo (de ahora en adelante también llamada *reproyección*) y la proyección experimental dada (de ahora en adelante también llamada simplemente como *imagen*), permite remover aquellas proyecciones que no concuerden muy bien con la estructura de referencia.

El mismo método de comparación directa de las proyecciones, hacer posible el cálculo de una reconstrucción 3D de alta resolución para una macromolécula de la manera siguiente: se usa una reconstrucción 3D ya calculada de la misma macromolécula como un "núcleo" y un nuevo grupo de imágenes colectado de micrografías únicas (y no de pares de micrografías como en el esquema cónico-aleatorio); tal "núcleo" es usado como una referencia en el primer paso del método. En este caso, el conocimiento inicial de los valores para los ángulos de Euler no es necesario, y el procedimiento de colectar pares de micrografías se evita al mismo tiempo. Por lo tanto, los pasos iniciales, tediosos, del análisis de los datos, pueden realizarse mucho más rápidamente. En particular, resulta relativamente simple usar los datos obtenidos a un enfoque más bajo, posibilitando la obtención de reconstrucciones 3D de alta resolución.

Un método similar al desarrollado en este capítulo fue originalmente descrito por Dengler [31] como una técnica de "bootstrapping". Él desarrolló un procedimiento de refinamiento para una reconstrucción 3D en eje único, el cual fue usado para corregir dos parámetros de translación y un ángulo (la rotación en el plano) de cada proyección por comparación de esta con la proyección de la estructura 3D en la dirección correspondiente.

4.2 El Método de Refinamiento de la Orientación

El *método de refinamiento de la orientación* es un método eficiente computacionalmente de búsqueda exhaustiva de los tres ángulos de Euler, que no requiere un conocimiento inicial de los parámetros de orientación. El método comienza calculando las proyecciones 2D de un volumen de forma que cubran uniformemente el espacio. Esto es necesario para tener un número sufiente de reproyecciones calculadas en las direcciones de las proyecciones experimentales. La orientación de una proyección en el espacio está definido por los tres ángulos de Euler (θ, φ, ψ). Dos de estos ángulos, (θ y φ), son responsables de la dirección de la proyección en el espacio 3D, mientras que el tercero (ψ) describe la rotación de la proyección "en el plano". Por ende, basta con calcular las reproyecciones de la estructura 3D de referencia variando sólo los primeros dos ángulos y calcular el coeficiente de correlación cruzada entre la reproyección y la imagen dada, todo en coordenadas polares. Un segundo paso determinaría la similitud y el tercer ángulo ψ al mismo tiempo. Este cálculo puede realizarse eficientemente usando Transformadas Rápidas de Fourier (FFT) unidimensionales (1D).

Una distribución equiespaciada de reproyecciones puede encontrarse en base al siguiente razonamiento: la dirección de un vector normal al plano de la proyección está determinado por los dos ángulos θ y φ (ver la Fig. 4.1); el ángulo ψ indica una rotación "en el plano" de la misma proyección tomada a (θ, φ). El vector normal correspondiente está definido por:

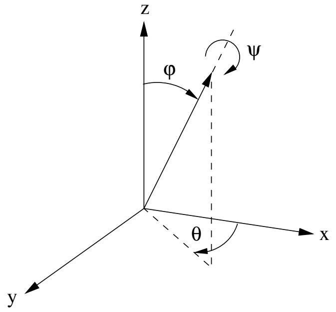


Figura 4.1: Los ángulos de Euler relativos a un sistema de coordenadas de mano izquierda

$$\vec{\eta} = \begin{pmatrix} \cos\theta \operatorname{sen}\varphi \\ \operatorname{sen}\theta \operatorname{sen}\varphi \\ \cos\varphi \end{pmatrix} \tag{4.1}$$

mientras que el requerimiento para una distribución equiespaciada significa que la norma de la derivada de este vector debe ser constante:

$$||\mathrm{d}\vec{\eta}|| = (\operatorname{sen}\varphi \; \mathrm{d}\theta)^2 + (\mathrm{d}\varphi)^2 = \text{constante} \tag{4.2}$$

de la Ec. (4.2) se deduce que para un incremento constante dado del ángulo φ, el incremento del ángulo θ debe ser escogido de acuerdo a:

$$\mathrm{d}\theta = \frac{\mathrm{d}\varphi}{\operatorname{sen}\varphi} \tag{4.3}$$

Por lo tanto, dado un incremento constante del ángulo φ, el incremento del ángulo θ debe variar de acuerdo a la Ec. (4.3). Y para ser consistentes, el incremento en el ángulo ψ debe escogerse igual al incremento del ángulo φ.

El programa diseñado para determinar los ángulos de Euler de proyecciones incluye los siguientes pasos:

1. Escoger el incremento en el ángulo φ (el ángulo de inclinación) y calcular el grupo de ángulos de Euler (θ, φ) .

2. Las proyecciones de la estructura 3D de referencia son calculadas de acuerdo a los ángulos precalculados en el paso anterior.

3. Todas las proyecciones de referencia se muestrean en coordenadas polares, se calcula la transformada de Fourier de todos los anillos escogidos y se almacenan en disco .

4. Se leen los datos de proyección a ser comparados, se muestrean en coordenadas polares y se calculan las transformadas de Fourier de los datos a lo largo de los anillos .

5. Se calculan las funciones de correlación cruzada 1D, en coordenadas polares, según la variable radial, entre los datos de proyección y todas las proyecciones de referencia .

6. La dirección de la proyección (en términos de los ángulos φ y θ) está determinada por el más grande coeficiente de correlación cruzada. El ángulo ψ está determinado por la posición del máximo en la correspondiente función de correlación cruzada en coordenadas polares.

Ya que las proyecciones calculadas en la dirección (θ, φ) tiene su versión "reflejada" en la dirección ($180° + \theta, 180° - \varphi$), solamente la mitad de las proyecciones tienen que ser calculadas en el paso 1, de forma que la amplitud de φ puede restringirse a (0°, 90°). Esta redundancia se toma en cuenta en el paso 5, donde dos CCF's (Cross-Correlation Funtions) son calculadas: una usando los complejos conjugados de las transformadas de Fourier de referencia, y la segunda sin ellos. El último paso corresponde a un reflejo implícito de la imagen de referencia que reduce la cantidad de memoria requerida a la mitad.

Los precálculos de los pasos 1-3 son realizados solamente una vez, sin importar el número de proyecciones experimentales que tengan que ser comparadas. Los pasos de precálculo y el paso 4, juntos, limitan sustancialmente la cantidad de tiempo de cómputo requerida. Los cálculos hechos en el paso 5 tienen que repetirse tantas veces como haya proyecciones de referencia para cada imagen a ser comparada, pero la cantidad de cómputo requerida es solamente un número pequeño de multiplicaciones de números complejos y una FFT 1D inversa.

4.3 Descripción del Programa Secuencial

Una iteración del *método de refinamiento por comparación directa de las proyecciones* se realiza ejecutando *cinco* programas secuenciales: **alignxy, voea, proj3d, align3d y vomd**. Los principales programas son dos: alignxy y align3d. A continuación se describirán en detalle cada uno de ellos.

Como entrada a una iteración tenemos dos conjuntos de datos: primero un grupo de N imágenes 2D de proyección, a las cuales ya se han asignado los valores iniciales en los ángulos de Euler de acuerdo a los procedimientos habituales para el método de colección de datos para la geometría de inclinación cónica aleatoria [89]; segundo tenemos un primer volumen reconstruido con el mismo grupo de imágenes. Los nombres de los archivos para cada imagen están almacenados en un archivo al que llamaremos *input.sel*.

Programa **alignxy**:
Sintaxis: alignxy volume_name input.sel
Uso: Realiza el refinamiento de los parámetros de translación. Los nuevos parámetros de translación quedan almacenados en el cabecero de cada imagen. *volume_name* será usado también por el programa *proj3d*.

Programa **voea**:
Sintaxis: voea theta1 theta2 phi1 phi2 delta > angles_file
Uso: Generar *angles_file*, un archivo con los tres ángulos de Euler muestreados equiespaciadamente en las direcciones de proyección. *theta1* es el valor del ángulo

de rotación inicial y *theta2* el final; por defecto son 0° y 359.9°, respectivamente. *phi1* es el valor inicial del ángulo de inclinación y *phi2* el final; los valores por defecto son 0° y 90°, respectivamente. *delta* es el incremento en el ángulo φ.
Nota: Para estructuras simétricas, la amplitud de los ángulos deben ser ajustados apropiadamente, dependiendo de la misma simetría.

Programa **proj3d**:
Sintaxis: proj3d volume_name angles_file template_output_images
Uso: Proyecta la estructura 3D (volume_name) de acuerdo a los ángulos almacenados en *angles_file* (y que fueron generados con voea). *template_output_images* se recomienda que sean de tres a cuatro letras y representará el nombre de las imágenes proyectadas con el formato prj????. Aquí "prj" es el patrón y "????" la numeración de la proyección que comienza en la 0000 hasta el número total de ternas de ángulos en *angles_file*. Ésta información angular es guardada en el cabecero de cada imagen.

Programa **align3d**:
Sintaxis: align3d reference.sel input.sel r1 r2 skip > out_align_file
Uso: Realiza el refinamiento de la orientación descrito arriba en la sección 4.2. *reference.sel* es el nombre del archivo creado con los nombres de todas la imágenes generadas por *proj3d*. *input.sel* es el mismo con que se comenzó la iteración. *r1* y *r2* son los radios de los anillos; sólo los datos en el intervalo *r1*–*r2* serán analizados. Si *skip* es igual a 1, entonces se toma cada anillo entre *r1* y *r2*; para *skip*= 2 cada dos, y así sucesivamente. *out_align_file* es el archivo resultante que contiene cuatro campos con el formato "*Id n distancia* ψ":

1. Id.- Identificador, número secuencial de cada campo. El primero es 0 (cero).
2. n.- número de la proyección de referencia más similar.
3. distancia.- Puede usarse como una medida de similitud.
4. ψ.- El ángulo de rotación "en el plano".

Programa **vomd**:
Sintaxis: vomd angles_file out_align_file > out_ok_file
Uso: Convertir el archivo de salida del programa *align3d*, "out_align_file", a un archivo que contenga la información de los tres ángulos de Euler, esto es, con el mismo formato que el archivo "angles_file" de salida del programa *voea*.

4.4 El Programa Secuencial del Método del Refinamiento de la Orientación

A continuación se presenta el pseudocódigo para el programa secuencial **align3d**. El análisis de este programa es necesario dada su complejidad (ver la sec. 4.2). El programa se ejecuta con la siguiente sintaxis:

align3d reference.sel input.sel r1 r2 skip > out_align_file

Condiciones:
I nombres de las reproyecciones en *reference.sel*.
J nombres de imágenes en *input.sel*. Estas representan las proyecciones a alinear.

Programa secuencial **align3d**

1. **for** $0 \leq i < I$ **do**
   (a) **read** imagen de referencia im_i
   (b) Se normaliza im_i
   (c) Se muestrea a coordenadas polares entre los radios *r1* y *r2* al incremento del intervalo, *skip*, dado.
   (d) Se realiza la FFT de los anillos.
   (e) Se almacena en memoria.
2. **endfor**
3. **for** $0 \leq j < J$ **do**
   (a) **read** imagen a alinear im_j
   (b) Se normaliza im_j
   (c) Se muestrea a coordenadas polares entre los radios *r1* y *r2* al incremento del intervalo, *skip*, dado.
   (d) Se realiza la FFT de los anillos.
   (e) **for** $0 \leq i < I$ **do**
      i. $c_i = \mathrm{CCF}(\mathrm{polar\_IM}_i, \mathrm{polar\_IM}_j)$
      ii. $cf_i = \mathrm{CCF}(\mathrm{polar\_IM}_i^*, \mathrm{polar\_IM}_j)$
   (f) **endfor**
   (g) **find** el máximo coeficiente de correlación cruzada en c_i y cf_i.
   (h) **print** datos de salida:
   1.- El identificador: j
   2.- El número de la proyección mas similar (i ó $I + i$)
   3.- El valor del mayor coeficiente de correlación.
   4.- El valor del ángulo ψ correspondiente.
4. **endfor**

End programa **align3d**

c_i, en general, es el coeficiente de correlación cruzada entre los anillos de las imágenes im_i y im_j. cf_i, en específico, es el coeficiente de correlación cruzada entre la imagen a alinear, polar_IM$_j$, y la versión reflejada de la proyección de referencia (el complejo conjugado de la transformada de Fourier de los anillos de la imagen im$_i$ muestreada en coordenadas polares) representada por polar_IM$_i^*$.

La normalización de las imágenes se realiza para simplificar las ecuaciones de la correlación cruzada [101]. La distribución de los valores de los pixels dentro de la región anular r1$-$2 y r2+2 es transformada linealmente a otra distribución de media 0 y desviación estándar 1. La razón de normalizar una zona de la imagen mayor que la comprendida entre r_1 y r_2 radica en el algoritmo para transformar la imagen a coordenadas polares (una interpolación cuadrática).

4.5 Dos Propuestas de Paralelización de los Programas

4.5.1 La justificación de la paralelización

Reportado en [84], el tiempo de cálculo para el *método de comparación directa de las proyecciones* (ver sec 4.1) viene a ser un poco más de dos días para 1000 proyecciones a alinear y para 5266 proyecciones de referencia (que corresponde a un muestreo de 2° en el ángulo φ). Estos corresponden a una estación de trabajo VAX 4000 modelo 60; que genera 1000 reproyecciones/hora y con un tiempo de 3 minutos para procesar la comparación de una imagen con 5266 proyecciones, ya generadas, de referencia.

Aunque en [84] se menciona que el tiempo de cálculo es "aceptable", éste calificativo hay que entenderlo en el contexto de la comparación con otro método de alineamiento que consumía 3 semanas para 1000 proyecciones [84]. De esta forma la idea de "paralelizar" los programas que conlleven más número de operaciones y/o consumo de recursos tiene una justificación clara.

Un análisis de los datos arriba expuestos indica que si tenemos 1000 imágenes a alinear y consideramos un intervalo de muestreo de 2° en el ángulo φ, entonces el programa *proj3d* debe generar 5266 reproyecciones que tardarán 5.266 horas de cálculo, a una velocidad de 1000 reproyecciones/hora. Y para el programa *align3d*, considerando que tarda unos 3 min en comparar una imagen con cada una de las 5266 reproyecciones, resulta que necesitamos 50 horas de cálculo, ó lo que es lo mismo, 2 días y 2 horas, para comparar todas las 1000 imágenes.

Además hay que considerar la cantidad de almacenamiento necesario, ya sea en disco o en memoria. En la Tabla 4.1 se dan diversas cifras correspondientes al espacio necesario para diferentes intervalos de muestro del ángulo φ, suponiendo un tamaño de 16KB por reproyección (que corresponde aproximadamente a una imagen de 64×64 pixels en coma flotante (4 bytes/pixel)). Este tamaño de imagen es normal para las aplicaciones con las que se trabaja en nuestro laboratorio, aunque hay campos de la microscopía electrónica que requieren imágenes mucho más grandes, de hasta 512×512 pixels, para efectuar la reconstrucción 3D a una alta resolución.

$d\varphi$	No. de reproyecciones	Tamaño total (MB)
10.0	229	3.58
5.0	869	13.58
3.0	2365	36.95
2.5	3390	52.97
2.0	5266	82.28
1.5	9314	145.53
1.0	20847	325.73

Tabla 4.1: Volumen de datos que es necesario almacenar según el incremento en φ escogido

4.5.2 Las propuestas

Aquí se presentan dos propuestas para la paralelización de los programas que realizan una semi-iteración del *método de comparación directa de las proyecciones.*

1. Paralelizar el programa *align3d.*

   Ventajas:
   — Simple.
   — Acorde a lo ya establecido. Se paraleliza un programa ya existente.

   Desventajas:
   — Espacio en disco: se guardan las imágenes producidas por *proj3d* en disco.
   — Son 5 programas a usar.

2. Paralelizar la unión de los programas: *voea*, *proj3d* y *align3d.*

   Ventajas:
   — Reducción del número de programas a usar.
   — Usa sólo memoria para guardar las imágenes y no espacio en disco.

   Desventajas:
   — Complejo.
   — Mayor cantidad de memoria usada por mayor almacenamiento de código.

El programa *alignxy* queda fuera del alcance de la paralelización porque está de por medio una reconstrucción tridimensional que debe realizarse al término de su ejecución. Dicho programa de reconstrucción ni se ha considerado como parte de los programas de alineamiento y conlleva código totalmente distinto, es más, pueden usarse diferentes aproximaciones, esto es, distintos programas de reconstrucción.

Aparte es un programa sencillo y poco consumidor de recursos: sólo es necesario reservar memoria para el volumen y una imagen.

El uso del programa *vomd* debe mantenerse ya que su trabajo es simple y permite la selección, por ejemplo, de las proyecciones con los coeficientes de correlación más altos. Como lo realizado en [95], donde para las reconstrucciones finales, del 10% al 30% de la imágenes con los coeficientes de correlación más bajos fueron excluidas.

4.5.3 Pseudocódigo para la primera opción

Se tienen las mismas condiciones que para el programa secuencial: I imágenes referencia, J imágenes a alinear.

La estrategia de paralelización responde a un modelo *maestro–hijo*. Se tiene un proceso "maestro" que coordina a Nproc procesos "hijos", y cada uno de ellos es responsable de $n = I/\text{Nproc}$ reproyecciones. El pseudocódigo para el proceso "maestro" queda:

```
Proceso "MAESTRO"
    Read y send a cada proceso n imágenes de referencia
            y el identificador de cada imagen.
    Para cada imagen a alinear:
        Send la imagen a cada proceso
        Receive de cada proceso el resultado:
            image_id[Nproc];        /* Identificador de las imágenes */
            c[Nproc];               /* Coeficientes de correlación */
            psi[Nproc];             /* Ángulos φ */
            flag[Nproc];            /* Bandera por si es el coef. reflejado */
            coef = −∞;
            for 0 ≤ i < Nproc do
                if c[i] ≥ coef then
                    coef = c[i];
                    index = i;
                endif
            endfor
            if flag[index] = 0 then
                similar = image_id[index];
            else
                similar = image_id[index] + I;
            endif
            print resultados:
                1.- j
                2.- similar
                3.- c[index] ó coef
                4.- psi[index]
```

End proceso "**MAESTRO**"

Y cada proceso "hijo" deberá realizar:

Proceso "**HIJO**"
Receive $nref$ imágenes de referencia y sus identificadores.
Para cada imagen:
— Normalizarla
— Muestrearla a coordenadas polares
— Realizar la FFT de los anillos
— Almacenarla en memoria
While receive la imagen a alinear: im_a **do**
for $0 \leq i < nref$ **do**
$c_i = \mathrm{CCF}(\mathrm{polar\_IM}_i, \mathrm{polar\_IM}_a)$
$cf_i = \mathrm{CCF}(\mathrm{polar\_IM}_i^*, \mathrm{polar\_IM}_a)$
endfor
/** Buscar el máximo coeficiente de correlación **/
$coef = -\infty;$
for $0 \leq i < nref$ **do**
if $c_i \geq coef$ AND $c_i > cf_i$ **then**
$coef = c_i;$
$idi = i;$
$flag = 0;$
else if $cf_i \geq coef$ **then**
$coef = cf_i;$
$idi = i;$
$flag = 1;$
endif
endfor
Send al proceso MAESTRO:
— Identificador de la imagen: $image\_id[idi]$
— El máximo coefiente de correlación: $coef$
— El ángulo ψ correspondiente
— El valor de la variable $flag$
End while
End Proceso "**HIJO**"

4.5.4 Pseudocódigo para la segunda opción

Las condiciones de entrada para la paralelización de los programas *voea*, *proj3d* y *align3d* son las siguientes:

Se supone que se tiene un proceso "maestro" y Nproc procesos "hijos". Los datos de entrada necesarios son:

- $d\varphi$, el incremento del ángulo de inclinación, para el muestreo equiespaciado en las direcciones de proyección.
- *r1*, *r2*, *skip*: que son los mismos valores usados por el programa *align3d*.
- *volume_name*: el nombre de la estructura tridimensional inicial.
- *input.sel*: el nombre del archivo que contiene el nombre de todas las projecciones a alinear.

De esta forma el programa en pseudocógico para el proceso "maestro" queda como:

```
Proceso "MAESTRO"
    Read la estructura 3D.
    Send a cada proceso "hijo" la estructura 3D.
    Send a los procesos hijos los valores de r1, r2 y skip
    Calcular los ángulos (θ, φ) y send a cada proceso hijo
        ternas de valores (id, θ, φ)

    Para cada imagen a alinear:
        Send la imagen a cada proceso
        Receive de cada proceso el resultado:
            image_id[Nproc];        /* Identificador de las imágenes */
            c[Nproc];               /* Coeficientes de correlación */
            psi[Nproc];             /* Ángulos φ */
            flag[Nproc];            /* Bandera por si es el coef. reflejado */
            coef = −∞;
            for 0 ≤ i < Nproc do
                if c[i] ≥ coef then
                    coef = c[i];
                    index = i;
                endif
            endfor
            if flag[index] = 0 then
                similar = image_id[index];
            else
                similar = image_id[index] + I;
            endif
            print resultados:
                1.- j
                2.- similar
                3.- c[index] ó coef
                4.- psi[index]
End proceso "MAESTRO"
```

Y cada proceso "hijo" deberá realizar:

```
Proceso "HIJO"
    Receive la estructura 3D.
    Receive los valores de r1, r2 y skip
    Receive ternas de valores (id, θ, φ), n número de ellas.

    for 0 ≤ i < n do
        image = retroproyectar la estructura 3D a (θ_i, φ_i)
        — Normalizar image
        — Muestrearla a coordenadas polares
        — Realizar la FFT de los anillos
        — Almacenarla en memoria: polar_IM_i
    endfor

    While receive imagen a alinear: im_a do
        — Normalizar image
        — Muestrearla a coordenadas polares
        — Realizar la FFT de los anillos
        — Almacenarla en memoria: polar_IM_i
        for 0 ≤ i < n do
            c_i = CCF(polar_IM_i, polar_IM_a)
            cf_i = CCF(polar_IM*_i, polar_IM_a)
        endfor
        /** Buscar el máximo coeficiente de correlación **/
        coef = −∞;
        for 0 ≤ i < n do
            if c_i ≥ coef AND c_i > cf_i then
                coef = c_i;
                idi = i;
                flag = 0;
            else if cf_i ≥ coef then
                coef = cf_i;
                idi = i;
                flag = 1;
            endif
        endfor
        Send al proceso MAESTRO:
            — Identificador de la imagen: image_id[idi]
            — El máximo coefiente de correlación: coef
            — El ángulo ψ correspondiente
            — El valor de la variable flag
    End while
End proceso "HIJO"
```

4.6 Descripción del Programa Paralelo

Se decidió que la mejor opción de paralelización era la segunda: Los programas *proj3d* y *align3d* se combinan en el nuevo programa de ejecución en paralelo **smaster3d** que se describirá a continuación. Además, el programa *vomd* se ha desdoblado en tres programas que permiten el seguimiento del proceso de la ejecución de cada iteración: *sreadangs, vomdtheshold, vomdok*, también estos programas se describirán en detalle a continuación.

Los programas *alignxy y voea*, se siguen ejecutando en modo secuencial durante la ejecución de una iteración del *método de comparación directa de las proyecciones*

Como entrada a una iteración tenemos las mismas condiciones que para una ejecución secuencial: dos conjuntos de datos; primero, un grupo de N imágenes a las cuales ya se han asignado los valores iniciales en los ángulos de Euler de acuerdo a los procedimientos habituales para el método de colección de datos para la geometría de inclinación cónica aleatoria [89]; segundo, tenemos un primer volumen reconstruido con el mismo grupo de imágenes. Los nombres de los archivos para cada imagen están almacenados en un archivo al que llamaremos *input.sel*.

Programa **sreadvals**:
Sintaxis: sreadangsy input.sel > angles0.txt
Uso: Guarda en el archivo *angles0.txt* la información de los tres ángulos de Euler que se encuentran en el cabecero de cada una de las imágenes especificadas en el archivo *input.sel*. Este archivo (angles0.txt) es el punto inicial para el chequeo de la condición de parada de todo el procedimiento (ver la descripción del programa *vomdcheck* en esta sección).

Programa **smaster3d**:
Sintaxis: smaster3d NP theta1 theta2 phi1 phi2 delta
vol_name input.sel r1 r2 skip > out_align_file
Uso Realiza el refinamiento de la orientación en función de los tres ángulos de Euler. *NP* indica el número de procesos paralelos en los que se ejecutará el programa, debe ser mayor de dos. Los parámetros *theta1 theta2 phi1 phi2 y delta* son los mismos que los usados anteriormente en el programa *voea*. vol_name es el nombre del volumen de referencia. *r1* y *r2* son los radios de los anillos; sólo los datos en el intervalod de *r1-r2* serán analizados. Si *skip* es igual a 1, entonces se tomará cada anillo entre *r1* y *r2*; para *skip*= 2 cada dos, y así sucesivamente. *out_align_file* es el archivo resultante que contiene cuatro campos con el formato "*Id n distancia* ψ":

1. Id.- Identificador, número secuencial de cada campo. El primero es 1 (uno).
2. n.- número de la proyección de referencia más similar.
3. distancia.- Puede usarse como una medida de similitud.
4. ψ.- Ángulo de rotación "en el plano".

Programa **vomdok**:
Sintaxis: vomdok angles_file out_align_file > angles_n.txt
Uso: Convertir el archivo de salida del programa *smaster3d*, "out_align_file", a un archivo que contenga la información de los tres ángulos de Euler, esto es, con el mismo formato que el archivo "angles_file" de salida del programa *voea*.

Programa **vomdthreshold**:
Sintaxis: vomdthreshold out_align_file name_output_file
Uso: Ordena los nombres de las imágenes contenidas en *out_align_file* según su "distancia" (la medida de similitud). Genera una gráfica llamada "graph.eps" con la que se puede visualizar las "distancias" y escoger un *umbral* para tener sólo las imágenes con una alta medida de similitud.

Programa **vomdcheck**:
Sintaxis: vomdcheck angles_$[n]$.txt angles_$[n+1]$.txt
Uso: Calcula las diferencias entre los datos generados durante una iteración del proceso. La condición de paro viene expresada como:

$$\frac{1}{N}\sum_{i=1}^{N}|\varphi_i - \widetilde{\varphi}_i| < \text{Umbral}$$

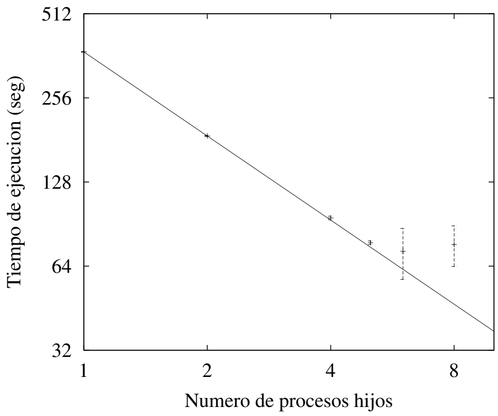
o hasta que los valores no cambien más. Como considera Frank ([50] p. 243), sólo en el ángulo de inclinación, φ. Los otros dos ángulos de Euler, θ y ψ son responsables de rotaciones de la imagen alrededor de un eje y no son de interés aquí.

4.7 Resultados

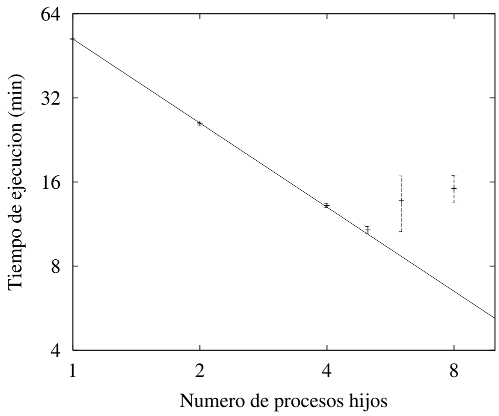
Los resultados de la paralelización pueden verse en la Fig. 4.2. El programa fue ejecutado en una máquina Power Challenger (de Silicon Graphics) con seis microprocesadores R10000.

En la Fig. 4.2(a) se visualiza la gráfica (log-log, como se sugiere en [22]) del tiempo de ejecución necesario para realizar 5266 retroproyecciones versus el número de procesos hijos. Corresponde al trabajo realizado en el primer ciclo *for* en el pseudocódigo del proceso hijo (en la subsec. 4.5.4, lineas 1-11). En la Fig. 4.2(b) se presenta el tiempo de ejecución, se presenta la media +/− la desviación estandar de varias mediciones (> 5), de comparar 1000 proyecciones contra las 5266 retroproyecciones de referencia, y que corresponde al ciclo *while* en el pseudocódigo del proceso hijo (de las líneas 13-40). En la tabla 4.2 se presentan los datos de la Fig. 4.2(a) pero representados ahora como *aceleración* (speedup) y *eficiencia*. De manera semejante la tabla 4.3 presenta los datos de aceleración y eficiencia para la Fig. 4.2(b).

La medición del tiempo de ejecución correspondiente a un proceso hijo de la Fig. 4.2 se realizó de dos maneras: (1) ejecutando el programa paralelo con un sólo proceso hijo y (2) con una versión secuencial del programa. Los tiempo de ejecución son semejantes, sin diferencias apreciables, entre (1) y (2), lo que indica que la carga debida a las comunicaciones en el programa paralelo es mínima.



(a) Retroproyección



(b) Comparación

Figura 4.2: Tiempos de ejecución del *Método de Refinamiento por Comparación Directa de las Proyecciones*. Se presenta la media $+/-$ la desviación estandar de varias mediciones (> 5). Detalles en el texto.

La caída en el rendimiento teniendo más de cinco procesos hijos se explicara a continuación. En nuestro escenario tenemos: i) el proceso maestro, ii) los procesos hijos, iii) los procesos del demonio de PVM para el ambiente de paso de mensajes y iv) los demás procesos que incluyen los del sistema operativo y los de otros usuarios (el uso de la máquina es compartida por varios grupos de trabajo). La máxima eficiencia se logra con cinco procesos hijos más el proceso maestro, esto es, un proceso por procesador (recordemos que la plataforma tiene seis microprocesadores). Pero la carga del proceso maestro es mínima, solo hace la colecta de los resultados de los procesos hijos, por lo que aún tendriamos que tener un buen resultado con seis procesos hijos. Sin embargo hay que tener en cuenta la carga debido a iii) y especialmente iv). Con iii) y iv) en nuestro escenario, la máxima eficiencia se logra con cinco procesos hijos, como queda demostrado en la gráfica de la Fig. 4.2, realizada bajo condiciones de carga "baja" en la plataforma (las mediciones fueron realizadas fuera del horario laborable).

Merece destacarse en este apartado que se estudió la posibilidad de incorporar a la aplicación paralela un control explícito del balanceo de la carga. Se pretendía de esta forma anticiparse a situaciones reales de uso de un máquina multiprocesador en un entorno de acceso y ejecución no exclusivo. Sin embargo, se observó que la propia gestión de los procesos que realiza el sistema operativo de la máquina (Irix 6.3, [17] p. 335) era bastante eficiente. En concreto, el sistema operativo "ve" todo el hardware como una sola máquina, esto es, los procesos no quedan residentes en un solo procesador, sino que la cola del despachador reasigna dinámicamente cada proceso al procesador disponible. Este hecho se verificó estudiando exhaustivamente el patrón de ejecución del programa paralelo. Con menos de siete procesos hijos, se mantiene una rebanada del proceso por procesador; arriba de seis procesos hijos, los multiplexa en los procesadores. Así, como no se tiene control directo del usuario sobre los procesadores, lo que se buscó finalmente fue equidistribuir la carga. En la estrategia de paralelización tratada en este capítulo, esta equidistribución se logró fácilmente haciendo que cada proceso hijo mantuviese un número igual de reproyecciones (ver pseudocódigos de los procesos hijos).

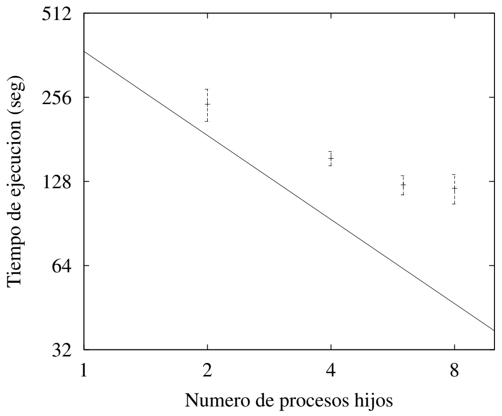
Las gráficas de la Fig. 4.2 se obtuvieron bajo una situación de carga "baja" sobre la máquina. Es de suponerse que los tiempos de ejecución aumentarán, aún más allá de la desviación estandar, si la carga de trabajo es "pesada". La Fig. 4.3 muestra la media $+/-$ la desviación estandar de los tiempos de ejecución del programa paralelo bajo una situación de carga "pesada" (el sistema operativo de la máquina designa menos rebanadas de tiempo a nuestros procesos), se observa que los tiempos de ejecución aumentan ostensiblemente.

No. procesos hijos	Aceleración	Eficiencia
1	1.00	1.00
2	2.00	1.00
4	3.93	0.98
5	4.82	0.96
6	5.17	0.86
8	4.89	0.61

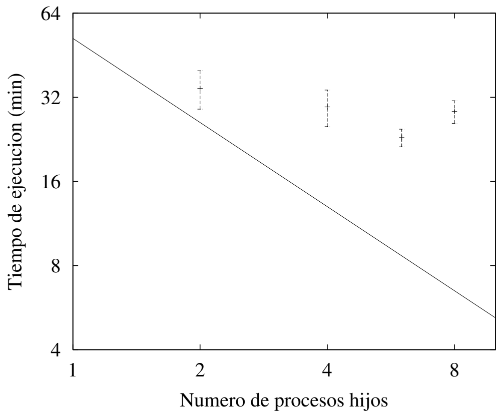
Tabla 4.2: La aceleración y la eficiencia, según el número de procesos hijos para la parte de la reproyección de la Fig. 4.2(a).

No. procesos hijos	Aceleración	Eficiencia
1	1.00	1.00
2	2.01	1.01
4	3.94	0.98
5	4.82	0.96
6	3.79	0.63
8	3.43	0.43

Tabla 4.3: La aceleración y la eficiencia, según el número de procesos hijos para la parte de la comparación de la Fig. 4.2(b).



(a) Reproyección



(b) Comparación

Figura 4.3: Tiempos de ejecución del *Método de Refinamiento por Comparación Directa de las Proyecciones* para una condición de carga "pesada" en la plataforma. Se presenta la media $+/-$ la desviación estandar de varias mediciones (> 7).

Capítulo 5

Método de Refinamiento de la Geometría vía Transformadas de Radon

5.1 Introducción

Este método ha sido desarrollado por Radermacher [91]. El objetivo que buscó este investigador fue obtener una técnica fiable que permitiese la determinación de los ángulos de proyecciones aisladas de partículas preparadas en agua vitrificada.

Como ya se ha indicado anteriormente en esta memoria, las imágenes de crio-microscopía electrónica tienen una relación señal-ruido menor que 1. Bajo estas condiciones no es factible encontrar una única linea común entre dos proyecciones, como en el caso del método de reconstitución angular propuesto por van Heel [117] (ver sec. 2.8.1), puesto que la correlación cruzada podría ser dominada por el ruido presente.

En principio, la base matemática del método propuesto por Radermacher podría ser lo suficientemente robusta frente al ruido como para resolver tanto el problema de refinar la geometría de un conjunto de imágenes partiendo de un volumen previo de referencia, como para tratar el problema de la asignación directa la geometría a partir de un grupo de proyecciones con direcciones desconocidas, sin ninguna información *a priori*. Sin embargo, la realidad actual del método sólo lo hace aplicable al problema del *refinamiento*, quedando la asignación directa de una geometría aún pendiente de ciertos desarrollos matemáticos (Radermacher 1997, comunicación personal).

Al igual que el *método de comparación directa de las proyecciones*, tratado en el capítulo anterior, el *método de la transformada de Radon* también permite incluir los datos de proyecciones, adicionales en el proceso; así como también llegar a una reconstrucción 3D de alta resolución (ver sec. 4.1).

De esta manera, el *método de la transformada de Radon* permite conseguir un mejora de la resolución de la reconstrucción tridimensional debido a una determinación más aproximada de todos los ángulos o al aumento en el número de las las proyecciones experimentales usadas para la reconstrucción.

El método está basado en la correlación-cruzada [101] de transformadas bi- y tri-dimensionales de Radon. Primero, partiendo de un subgrupo de imágenes que muestren una orientación preferente identificable, se obtiene una reconstrucción tridimensional usando el esquema cónico aleatorio [89], el cual puede combinarse ya sea con la técnica de Proyección Sobre Conjuntos Convexos (POCS, por sus siglas en inglés) [106, 105, 14, 12] o con una reconstrucción multi-cono [16, 83, 6, 25]. Ambos tienen el propósito de producir una reconstrucción virtualmente sin cono perdido. En este punto se calcula la la transformada de Radon de la reconstrucción tridimensional. La transformada de Radon tridimensional provee esencialmente una representación compacta de todas la proyecciones posibles del volumen. A continuación se calcula la transformada bidimensional de Radon de cada proyección experimental cuya orientación va a ser encontrada, y se calcula la correlación cruzada tridimensional como una función de los tres ángulos de Euler α, β, γ. Las ángulos para los cuales esta función de correlación-cruzada tridimensional asume su máximo define la orientación de la proyección relativa al volumen. De una manera similar a como se determina la orientación, la translación de la proyección puede determinarse también simultáneamente calculando lo que viene a ser una función de correlación cruzada en cinco dimensiones.

Mientras el uso de las transformadas de Radon provee la herramienta matemática para el alineamiento, el método puede también describirse en términos más simples. Esencialmente, todas las proyecciones posibles de un volumen son calculadas y usadas como referencia para el alineamiento de la proyección experimental usando técnicas de correlación cruzada. Los ángulos de la proyección experimental son aquellos ángulos de la referencia calculada que resultan en el pico más alto de la correlación cruzada. Así mismo, una transformada de Radon tridimensional contiene la misma información que un grupo completo de proyecciones bidimensionales, pero de una forma mucho más compacta, y permite que el alineamiento translacional y rotacional pueda efectuarse simultáneamente.

Como la orientación es encontrada por correlación cruzada, es posible plantear una formulación equivalente de este método como un ajuste por mínimos cuadrados de la transformada de Fourier de la proyección en la transformada de Fourier tridimensional del objeto, con los tres ángulos de Euler y un vector de corrimiento bidimensional como parámetros. Un método basado en un ajuste explícito por mínimos cuadrados ha sido planteado previamente por Provencher y Vogel [88, 123]. Sin embargo su método fue desarrollado para el refinamiento de ángulos preespeficados, mientras que el método propuesto por Radermacher permite ambas cosas: una búsqueda completa en el espacio angular, y un refinamiento de los ángulos alrededor de valores especificados. Una limitación de este método, como cabe presuponerse, es el hecho de que es necesario contar con un modelo en baja resolución del objeto.

5.2 Teoría

Una descripción detallada sobre las transformadas de Radon puede encontrarse, por ejemplo, en Deans [30], cuya terminología es seguida aquí.

Dada una función tridimensional $f(\mathbf{r})$, $\mathbf{r} = (x, y, z)$, su transformada de Radon queda definida como:

$$\hat{f}(p, \xi) = \int f(\mathbf{r})\delta(p - \xi \cdot \mathbf{r})\mathrm{d}\mathbf{r} \tag{5.1}$$

donde $\delta(p - \xi \cdot \mathbf{r})$ especifica un plano sobre el cual se lleva a cabo la integración, y ξ es el vector unitario que determina la dirección de este plano:

$$\xi = (\cos\phi \operatorname{sen}\theta, \cos\theta, \operatorname{sen}\phi \operatorname{sen}\theta) \tag{5.2}$$

Nótese que los ángulos en la Ec. (5.2) podrían ser escogidos de manera diferente; la convención usada aquí refleja una realización específica del método.

La transformada de Radon bidimensional puede ser definida de forma idéntica a la Ec. (5.1):

$$\hat{g}(p, \zeta) = \int f(\mathbf{r})\delta(p - \zeta \cdot \mathbf{r})\mathrm{d}\mathbf{r} \tag{5.3}$$

con ζ siendo de nuevo el vector unitario ortogonal a la dirección de las rayos sobre los cuales se realiza la integración:

$$\zeta = (\operatorname{sen}\epsilon, \cos\epsilon) \tag{5.4}$$

De nuevo, los ángulos en la Ec. (5.4) podrían ser escogidos de manera diferente.

En la literatura sobre reconstrucción tridimensional a partir de proyecciones, la transformada de Radon bidimensional es referida frecuentemente como "senograma" [118].

La transformada de Radon bidimensional puede ser calculada mediante el cálculo de todas las proyecciones unidimensionales (con un incremento angular pequeño) de una imagen bidimensional.

Una transformada de Radon tridimensional puede ser calculada en dos pasos, primero calculando todas las proyecciones bidimensionales a lo largo de las direcciones $(\cos\phi, 0, \operatorname{sen}\phi)$, p.e. en el plano x-z, seguido por una transformada de Radon bidimensional de todas las proyecciones bidimensionales. El primer paso crea lo que se conoce en microscopía electrónica como una serie de inclinación simple alrededor del eje-y (ver sec. 2.7.1).

El resultado de la transformada de Radon tridimensional es un conjunto de líneas que pueden identificarse por los ángulos ϕ_i de la proyección bidimensional y un segundo ángulo θ_i que define la dirección de la proyección dentro del plano al ángulo ϕ_i. La transformada de Radon es almacenada como la función tridimensional $\hat{f}(p, \theta, \phi)$.

De forma análoga, la transformada bidimensional de la proyección de referencia es calculada como el conjunto de integrales sobre las líneas a ángulos ϵ_i. La transformada de Radon bidimensional es almacenada como la función $\hat{g}(p, \epsilon)$.

La transformada de Radon bidimensional $\hat{g}(p, \epsilon)$ de una proyección a los ángulos de Euler α, β, γ puede ser extraída de la transformada de Radon tridimensional como se mostrará a continuación:

El vector unitario ζ en el plano de una transformada de Radon bidimensional de una proyección del volumen puede ser escrito como:

$$\zeta = (\operatorname{sen}\epsilon, \cos\epsilon, 0) = \begin{pmatrix} \cos\epsilon & \operatorname{sen}\epsilon & 0 \\ -\operatorname{sen}\epsilon & \cos\epsilon & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{5.5}$$

$$= D_\epsilon \mathbf{n}_y^p \tag{5.6}$$

siendo $D_\epsilon \mathbf{n}_y^p$ el vector unitario a lo largo del eje-y en la proyección.

Supongamos que el objeto es rotado por los ángulos de Euler α, β, γ. Sea $\mathbf{r}'$ un vector en el sistema de coordenadas del volumen rotado y sea $\mathbf{r}$ un vector en el objeto sin rotar. Entonces:

$$\mathbf{r}' = D_\gamma D_\beta D_\alpha \mathbf{r} \tag{5.7}$$

$$= \begin{pmatrix} \cos\gamma & \operatorname{sen}\gamma & 0 \\ -\operatorname{sen}\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \cos\beta & 0 & -\operatorname{sen}\beta \\ 0 & 1 & 0 \\ \operatorname{sen}\beta & 0 & \cos\beta \end{pmatrix} \times \begin{pmatrix} \cos\alpha & \operatorname{sen}\alpha & 0 \\ -\operatorname{sen}\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{r} \tag{5.8}$$

ζ expresada en el sistema de coordenadas del objeto rotado es

$$\zeta' = (D_\epsilon \mathbf{n}_y^p)' = D_\gamma D_\beta D_\alpha D_\epsilon \mathbf{n}_y^p \tag{5.9}$$

Como las rotaciones en α y ϵ son alrededor del mismo eje, pueden combinarse fácilmente haciendo $\alpha' = \alpha + \epsilon$, y la Ec. (5.9) queda

$$\zeta' = D_\gamma D_\beta D_{\alpha'} \mathbf{n}_y^p \tag{5.10}$$

$$= \begin{pmatrix} \cos\alpha' \operatorname{sen}\gamma + \operatorname{sen}\alpha' \cos\beta\cos\gamma \\ \cos\alpha' \cos\gamma - \operatorname{sen}\alpha' \cos\beta \operatorname{sen}\gamma \\ \operatorname{sen}\alpha' \operatorname{sen}\beta \end{pmatrix} = \begin{pmatrix} \zeta_1' \\ \zeta_2' \\ \zeta_3' \end{pmatrix} \tag{5.11}$$

El vector unitario que determina la orientación del plano de integración en la transformada de Radon tridimensional es:

$$\xi = \begin{pmatrix} \cos\phi \operatorname{sen}\theta \\ \cos\theta \\ \operatorname{sen}\phi \operatorname{sen}\theta \end{pmatrix} \tag{5.12}$$

De la condición $\zeta'(\alpha, \beta, \gamma) = \xi(\phi, \theta)$ se deduce:

$$\theta = \arccos(\zeta_2') \tag{5.13}$$

y entonces, para $\theta \neq 0$:

$$\cos\phi = \zeta_1' / \operatorname{sen}\theta, \tag{5.14}$$

$$\operatorname{sen}\phi = \zeta_3' / \operatorname{sen}\theta \tag{5.15}$$

Para $\theta = 0$ el ángulo ϕ está indeterminado y puede ser puesto a $\pi/2$.

Los ángulos θ, ϕ en la transformada de Radon tridimensional corresponden a la línea con el ángulo ϵ en la transformada de Radon de la proyección a ángulos α, β, γ. Con θ, ϕ conocidos, una línea de la transformada de Radon bidimensional correspondiente puede ser extraída de la transformada de Radon tridimensional. Ya que $\alpha' = \alpha + \epsilon$ varía con la línea en la transformada de Radon bidimensional, la transformada es extraída a $\alpha = 0$ con el intervalo de ϵ comprendido entre $-90°$ y $+(90 - \Delta\epsilon)°$. Las transformadas de Radon correspondientes a la proyección rotada por un ángulo adicional α, pueden ser obtenidas por permutaciones e invirtiendo las líneas en la transformada extraída. Si $l(\epsilon)$ es una línea en la transformada de Radon bidimensional, entonces

$$l(\epsilon + \phi) = \mathrm{mirror}(l(\epsilon)) \tag{5.16}$$

La orientación de una proyección es determinada por correlación cruzada: Sea $\hat{r}^2_{\beta,\gamma}(p, \alpha')$ la transformada de Radon bidimensional extraída de la transformada de Radon tridimensional, y sea $\hat{g}(p, \epsilon)$ la transformada de Radon bidimensional de una proyección. Entonces se calcula la función de correlación cruzada definida como:

$$c_{\beta,\gamma}(\alpha) = \int\int \hat{r}^2_{\beta,\gamma}(p, \epsilon)\hat{g}(p, \epsilon + \alpha)\mathrm{d}p\,\mathrm{d}\epsilon \tag{5.17}$$

La orientación está determinada por el máximo de c. Los ángulos $\alpha_0, \beta_0, \gamma_0$ a los cuales se encuentra el máximo de c, describen la orientación de la proyección g relativa al volumen f.

Además de la orientación de la proyección, es también posible encontrar la relación translacional de la proyección con respecto al volumen. Para encontrar la relación translacional es necesario hacer uso de la propiedad de desplazamiento de la transformada de Radon, la cual declara:

$$\mathbf{R}(f(\mathbf{r} - \mathbf{a})) = \int f(\mathbf{r} - \mathbf{a})\delta(p - \xi \cdot \mathbf{r})\mathrm{d}\mathbf{r} \tag{5.18}$$

$$= \int f(\mathbf{s})\delta((p - \xi \cdot \mathbf{a}) - \xi \cdot \mathbf{s})\mathrm{d}\mathbf{s} \tag{5.19}$$

La función de correlación cruzada de la Ec. (5.17) es calculada como:

$$c_{\beta,\gamma}(\alpha) = \int\int \hat{r}^2_{\beta,\gamma}(p, \epsilon)\hat{g}(p - \zeta \cdot \mathbf{a}, \epsilon + \alpha)\mathrm{d}p\,\mathrm{d}\epsilon \tag{5.20}$$

con $\zeta = (\operatorname{sen}\epsilon, \cos\epsilon)$ y $\mathbf{a} = r(\cos\eta, \operatorname{sen}\eta)$:

$$\zeta \cdot \mathbf{a} = r\operatorname{sen}(\epsilon + \eta) \tag{5.21}$$

Reemplazando p por $p - r\,\mathrm{sen}(\epsilon + \eta)$ en $\hat{g}$ en la Ec. (5.17) se obtiene la función de correlación cruzada dependiendo de los tres ángulos de Euler α, β, γ, y un vector de desplazamiento $r(\cos\eta, \mathrm{sen}\,\eta)$:

$$c(\alpha, \beta, \gamma, r, \eta) = \int\int \hat{r}^2_{\beta,\gamma}(p, \epsilon)\hat{g}(p - r\,\mathrm{sen}(\epsilon + \eta), \epsilon + \alpha)\,\mathrm{d}p\,\mathrm{d}\epsilon \qquad (5.22)$$

La Ec. (5.17) es la misma que la Ec. (5.22) para el caso especial cuando $r = 0$.

5.3 Descripción del Programa Secuencial

La realización práctica del método de refinamiento de la geometría usando transformadas de Radon, en modo secuencial, consta de la ejecución de *cinco* programas: **doradon2d, fftra2d, doradon3d, fftra3d y ccf_3d**. El programa principal, el corazón del método, que calcula la función de correlación cruzada de la Ec. (5.17) ó de la Ec. (5.22), es **ccf_3d**. Estos programas están incluidos dentro del paquete de procesamiento de imágenes de microscopía electrónica XMIPP [73] desarrollado por nuestro grupo. A continuación se describirán en detalle cada uno de ellos.

Como entrada a una iteración del proceso de refinamiento, tenemos:

1. Un grupo de N imágenes 2D de proyección a las cuales ya se han asignado los valores iniciales en los ángulos de Euler de acuerdo a los procedimientos habituales para el método de colección de datos para la geometría de inclinación cónica aleatoria [89].

2. Un primer volumen de referencia reconstruido con el mismo grupo de imágenes del punto anterior.

Programa doradon2d

Uso: Realiza la transformada de Radon (TR) de una imagen bidimensional. La imagen de entrada es rotada un ángulo ϵ y cada línea de la imagen de salida corresponde a la suma de las líneas (sobre el eje-x) de la imagen rotada.

Sintaxis: doradon2d input_image output_image radius samples
xoff yoff Astart Aend Ainc mode

donde:

input_image:
Nombre de la imagen bidimensional de entrada en formato SPIDER [51].

output_image:
Nombre de la imagen de salida, equivalente a una imagen tipo SPIDER.

radius: Radio de la máscara circular. Sólo los valores de la imagen dentro de ésta máscara serán usados en el cálculo de la transformada de Radon.

samples:
Dimensiones de la línea de la transformada de Radon.

xoff yoff:
El centro de rotación para el cálculo de la TR es $(N-1)/2, (N-1)/2$, pero puede tener un corrimiento dando como entrada estos datos.

Astart Aend Ainc:
Amplitud y muestreo angular de la transformada. Estos datos son almacenados en el cabecero de la imagen de salida.

mode: Si mode=0, ningún cambio es aplicado a la imagen de entrada. Si mode=1, substrae el mínimo valor a toda la imagen. Si mode=2, substrae el promedio de la imagen y entonces pone todos los valores negativos a 0 (cero).

Programa fftra2d

Uso: Realiza la transformada de Fourier de una transformada 2D de Radon.

Sintaxis: fftra2d input_image output_image filter [-a epsilon]

donde:

input_image:
Nombre de la imagen, una TR bidimensional, de entrada

output_image:
Nombre de la imagen de salida

filter: Puede ser "n", no; o "y", de yes. Si filter="y", entonces se le aplicará un filtraje a la TR y serán necesarios los datos: **type number cutoff[s] [temp]**

type: Un número que indica el tipo del filtro. Puede ser (3)Gauss Low Pass, (4)Gauss High Pass, (5)Fermi Low Pass ó (6)Fermi High Pass.

number:
number=1, para un filtro simple. number=2 para un filtro múltiple (Band-Pass ó Cut-Off).

cutoff[s]:
Radio(s) de el(los) filtro(s).

[temp]: Opcional. El valor de la temperatura para los filtros tipo Fermi.

[-a epsilon] :
Opcional. "-a" activa la bandera para aplicar una normalización de los coeficientes de Fourier por su propio valor. Un filtro tipo Wiener es aplicado. Si F_N es el n-ésimo coeficiente nuevo de Fourier y F_O es el valor antiguo, la fórmula para la normalización es: $F_N = F_O/(|F_O| + \text{epsilon})$.

Programa doradon3d

Uso: Realiza la transformada 3D de Radon de un volumen. La transformada tridimensional de Radon es almacenada como $\hat{r}^3_{i,j,k}$, donde i es el índice de la línea, correspondiente a p en la Ec. (5.1), el índice j se refiere al ángulo θ, y k se refiere al ángulo ϕ. Entonces, cada sección en la TR 3D corresponde a una TR 2D de una proyección en un serie de inclinación de eje único.

Sintaxis: doradon2d spider_volume output_volume
theta1 theta2 phi1 phi2 delta
radius samples mode

donde:

spider_volume:
Nombre del volumen de entrada en formato SPIDER.

output_volume:
Nombre del archivo de salida. El formato del archivo de salida es equivalente a un volumen en formato SPIDER.

theta1 theta2:
Valores inicial y final de la amplitud angular para el ángulo de rotación.

phi1 phi2:
Valores inicial y final de la amplitud angular para el ángulo de inclinación.

delta: valor del muestreo angular.

radius: Radio de la esfera. Sólo los valores del volumen dentro de la esfera serán usados el cálculo de la transformada 3D de Radon.

samples:
Dimensiones de la línea de la transformada 2D de Radon.

mode: Igual que el programa **doradon2d**

Programa fftra3d

Uso: Realiza la transformada de Fourier de una transformada 3D de Radon.

Sintaxis: fftra3d input_volume output_volume filter [-a epsilon]

donde:

input_volume:
Nombre del volume, una TR tridimensional, de entrada

output_volume:
Nombre del volumen de salida

Todos los demás parámetros son iguales que para el programa **fftra2d**.

Programa ccf_3d

Uso: Realiza el alineamiento rotacional y/o translacional de las transformadas de Fourier de transformadas de Radon. Este programa se describirá en detalle en la subsección 5.3.1.

Sintaxis: ccf_3d name_in name_sel beta1 beta2 gamma1 gamma2
psi1 psi2 inc shift search_type

donde:

name_in:
Nombre de la transformada de Fourier 3D de la TR 3D del volumen de referencia.

name_sel:
Nombre del archivo que contiene todos los nombres de las imágenes a alinear. El formato de cada línea es: *nombre_imagen flag*; si *flag*> 0 la imagen será considerada, si flag < 0 la imagen ha sido rechazada en alguna parte del proceso.

beta1 beta2:
Amplitud de la búsqueda en el ángulo de inclinación.

gamma1 gamma2:
Amplitud de la búsqueda en el ángulo de rotación (primer ángulo de Euler).

psi1 psi2:
Amplitud de búsqueda en el ángulo de rotación (tercer ángulo de Euler, rotación "en el plano").

inc: Incremento angular para la búsqueda.

shift: Longitud, en pixels, para la búsqueda en el desplazamiento x-y.

search_type:
puede ser "f" (full), búsqueda completa, ó "s" (subsearch).

Valores típicos para cada parámetro:

beta1 beta2:
$-90°, 90°$

gamma1 gamma2:
$-180°, 180°$

psi1 psi2:
$-90°, 90°$

inc: 2°. Este valor es el indicado para una esfera de radio máximo de 29 voxels.

shift: Si shift=0, sólo se calculará el alineamiento rotacional. shift=x indica que se hará el alineamiento translacional dentro de un corrimiento máximo de x pixels; para este caso todos los valores en los parámetros de los ángulos deben ser igual a cero. Si se intenta fijar muchos parámetros al mismo tiempo con datos con mucho ruido, el proceso puede volverse inestable, por esta razón se sugiere realizar el alineamiento translacional solo (con todos los valores en las amplitudes de búsqueda de los ángulos betas's, gamma's y psi's, iguales a cero).

5.3.1 Pseudocódigo del programa secuencial

A continuación se presenta el pseudocódigo para el programa **ccf_3d** en forma secuencial. Este programa realiza el alineamiento rotacional y/o translacional de las transformadas de Fourier de transformadas de Radon y es preciso entenderlo como paso previo a la paralelización del mismo.

Sintaxis: ccf_3d name_in name_sel beta1 beta2 gamma1 gamma2
psi1 psi2 inc shift search_type

Programa ccf_3d.

1. **read** name_in; /** TF 3D de la TR 3D del volumen **/
2. irange = no. de valores en ψ's = $\frac{\psi_{\text{end}} - \psi_{\text{start}}}{\Delta angles} + 1$;
3. $\Delta\beta = \Delta angles$;
4. **for each** imagen en name_sel **do**
   (a) **read** image$_k$;
   (b) **for** $\beta_{\text{start}} \leq \beta_i \leq \beta_{\text{end}}, \quad \beta_{i+1} = \beta_i + \Delta\beta,$ **do**
      i. $\Delta\gamma = \Delta\beta / \operatorname{sen}|\beta|$;
      ii. **for** $\gamma_{\text{start}} \leq \gamma_j \leq \gamma_{\text{end}}, \quad \gamma_{j+1} = \gamma_j + \Delta\gamma,$ **do**
         A. reference_img = do_image (name_in, β_i, γ_j);
         B. **ccf_radons**(reference_img, image$_k$, irange, shift);
            /** Función de correlación cruzada entre la imagen de referencia y la imagen leída **/
         C. **find** el máximo de la función de correlación cruzada;
      iii. **endfor**
   (c) **endfor**
5. **endfor**

Como ha sido realizado, el programa contiene cuatro lazos principales. El primero (que comienza en el paso 4 en el pseudocódigo anterior) aplica la búsqueda de ángulos a cada imagen de entrada. Los otros tres lazos actúan sobre la búsqueda angular: primero sobre β_i (en el paso 4.b), segundo en γ_j (en 4(b).ii) y por último sobre ϵ_n (en la función *ccf_radons* en 4(b).ii.B). La amplitud de ϵ es siempre la amplitud completa de la transformada de Radon bidimensional.

La función *do_image()* (en 4(b).ii.A) calcula la imagen de referencia bidimensional, usando la Ec. (5.9) para $\alpha = 0$ y las Ecs. (5.12)–(5.15), a los ángulos dados (β_i, γ_j), a partir de la transformada tridimensional de Radon del volumen.

La función *ccf_radons()* (que aparece en el paso 4(b).ii.B del mismo pseudocódigo anterior) se describirá a continuación. Esta función es la que realiza en sí misma la correlación cruzada entre las transformadas de Radon bidimensionales correspondiente a la Ec. (5.22).

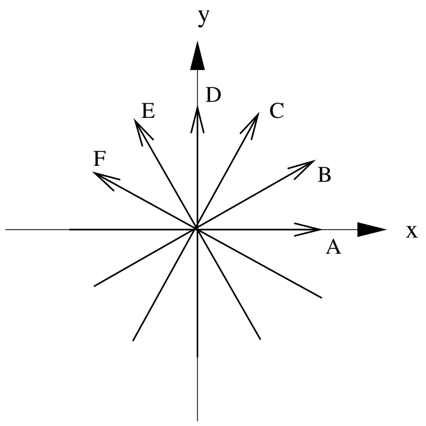


Figura 5.1: Representación esquemática de una transformada de Radon muestreada a $\alpha = 30°$.

Las entradas a la función *ccf_radons()*, recordemos, son la transformada de Fourier de la transformada de Radon bidimensional de la imagen de referencia y la imagen leída. La primera línea, (1) en el pseudocódigo, indica que tenemos nsam-2 muestras reales, debido a las propiedades de la transformada de Fourier de una señal real, que es siempre una función Hermítica [80, 9].

Función ccf_radons(A[rows][nsam], B[rows][nsam], irange, shift)

1. rsam = nsam−2; /** Muestras en el espacio real **/
2. radius=0; phase=0;
3. **for** $0 \leq Kp \leq$ irange /** Ciclo sobre ψ **/
   (a) **for** $0 \leq Kl \leq$ rows /** Ciclo sobre las filas de las TR's **/
      i. ccf[Kl][K] = $\mathrm{A}_{Ka}\mathrm{B}^*_{Kl}$; /** K es el numero de muestras en A y B **/
      ii. INV_FT (ccf[Kl], rsam);
   (b) **endfor**
   (c) ffmax = $\frac{1}{\sigma_A\sigma_B}\sum_{i=0}^{rows-1}$ ccf[i][$rsam/2$]; /** Calcula sólo el alineamiento rotacional **/
   (d) **if** shift> 0 **then** /** Calcula el alineamiento translacional **/
   /** Precisión: 1 pixel.
   Las coordenadas del corrimiento son determinadas en coordenadas polares: Amplitud angular = $n * 8$ intervalos, a n pixels de desplazamiento **/
      i. Δarg = π/rows
      ii. **for** $1 \leq n \leq$ shift **do**
         - intervals= $8 \cdot n$;
         - Δphase = 2π/intervals;
         - **for** $0 \leq j \leq$ intervals **do**
           – $\phi = j \cdot \Delta$phase;
           – ff = $\frac{1}{\sigma_A\sigma_B}\sum_{i=0}^{rows-1}$ ccf[i] [$rsam/2 + n \cdot \cos(\phi + \Delta\mathrm{arg} \cdot i)$];
           – **if** ff > ffmax **then** /** Resultados **/
             ∗ ffmax = ff;
             ∗ radius = n;
             ∗ phase = ϕ;
           – **endif**
         - **endfor** /** End sobre ciclo de fase **/
      iii. **endfor** /** End sobre ciclo de la amplitud **/
   (e) **endif** /** End sobre el bloque de corrimiento **/
4. **endfor** /** End ciclo sobre ψ **/
5. /** Resultados en: ffmax/rows, radius, phase **/

El índice Ka en la línea 3(a).i indica una nueva transformada de Radon rotada un ángulo α. Esta es una propiedad de las transformadas de Radon ya indicada en la sección 5.2: las transformadas de Radon correspondientes a una proyección rotada por un ángulo adicional α pueden ser obtenidas por permutaciones e inversiones de las líneas en la transformada extraída. Esta propiedad puede entenderse viendo el esquema de la Fig. 5.1, las líneas A,B, ... ,F, representan la transformada de Radon de una imagen muestreadas a $\alpha = 30°$, A está a 0°, B a 30° y así sucesivamente hasta F que está a 150°. La línea a 180°, $l(180°)$, es la reflejada a 0°, esto es, igual a mirror($l(0°)$) (Ec. 5.16). Si se calculara la correlación entre esta imagen, tomada como referencia, y otra imagen con índices A',B', ... , F'; para calcular $C(0, 0°)$ (ver la Ec. 5.24 un poco más adelante) se calcula la correlación cruzada entre las líneas AA', BB', ... , FF'. De la misma forma, para calcular $C(0, 30°)$, se tomaría la correlación cruzada entre las líneas BA', CB', DC', ED', FE' y mirror(A)F'. Y así sucesivamente.

Si $\hat{g}(p, \epsilon)$ es la imagen leída y $\hat{r}^2_{i,j}(p, \epsilon)$ la reproyección extraída del volumen (A[][] y B[][], respectivamente, en el pseudocódigo) entonces primero se calcula la correlación normalizada entre las correspondientes líneas de la transformadas como:

$$c_{i,j,\alpha_k}(p', \epsilon) = \frac{1}{\sigma_g \sigma_r} \int \hat{r}^2_{i,j}(p, \epsilon)\hat{g}(p + p', \epsilon + \alpha_k)\,\mathrm{d}p \tag{5.23}$$

para cada α dentro de una amplitud especificada, σ_g y σ_r son la raíz cuadrada de la varianza de $\hat{r}$ y $\hat{g}$, respectivamente; segundo, si únicamente se busca el alineamiento rotacional, entonces se promedian en $p' = 0$, resultando:

$$C_{i,j,\alpha_k} = \frac{1}{N} \sum_{l=1}^{N} c_{i,j,\alpha_k}(0, \epsilon_l) \tag{5.24}$$

Si se realiza además el alineamiento translacional, la suma sobre todas las líneas $c_{i,j,\alpha_k}(p', \epsilon_l)$ se hace para $p' = r\,\mathrm{sen}(\epsilon_l + \eta)$, para un intervalo específico de r y para toda η en el intervalo $-\pi \leq \eta < \pi$, resultando

$$C_{i,j,\alpha_k,n,m} = \frac{1}{N} \sum_{l=1}^{N} c_{i,j,\alpha_k}(r_n\,\mathrm{sen}(\epsilon_l + \eta_m), \epsilon_l) \tag{5.25}$$

En la práctica, el alineamiento se realiza a una precisión fija en r de un pixel y η dividido en $r \cdot 8$ intervalos. Para un corrimiento de un pixel se calculan 8 coeficientes de correlación cruzada a $0°, 45°, 90°, 135°, 180°, 225°, 270°$ y $315°$.

La Ec. (5.23) está especificada en el pseudocódigo en las líneas 3.a–3.b. La Ec. (5.24) esta en la línea 3.c y la Ec. (5.25) en el bloque **if-then** de la línea 3.d. La división entre N de las ecuaciones (5.24) y (5.25) es realizada al final. En vez de almacenar todos los resultados, lo cual podría conducir a una lista en cinco dimensiones de coeficientes de correlación cruzada, solamente el coeficiente más grande es mantenido, junto con los parámetros de r_n (radius, en el pseudocódigo) y η_m (phase).

σ_g y σ_r, que aseguran que el valor del coeficiente de correlación cruzada esté normalizado entre -1 y 1, son calculadas en el dominio recíproco. σ_g es calculada cuando se realiza la transformada de Fourier y es almacenada en el cabecero de le imagen. σ_r debe calcularse cada vez que se extrae del volumen. La varianza se calcula en el dominio de Fourier aplicando la relación de Parseval ([80] p. 125) para la Transformada Discreta de Fourier, lo cual es muy simple, sencillamente se coge la suma sobre los coeficientes de Fourier al cuadrado excluyendo $F(0)$ (el cual es la media de la imagen) y dividirlos entre $n * (n-1)$, n siendo el orden más grande. La ventaja de hacerlo así es que puede calcularse la varianza después de haber aplicado algún filtraje.

5.4 Análisis Previo a la Paralelización

Una revisión del programa secuencial, a primera instancia, no da una guía obvia sobre como efectuar la paralelización. Nos encontramos, pues, ante un caso en que la estrategia de paralelización no es trivial y en el que es necesario realizar un análisis profundo de las dependencias entre las variables.

Desde el punto de vista del almacenamiento de datos, la cantidad de memoria requerida es relativamente pequeña: el volumen y dos imágenes bidimensionales. Si suponemos que tenemos un volumen de $64 \times 64 \times 64$ voxels y las imágenes de 64×64 pixels, tomando 60 muestras en la transformada de Radon a un muestreo angular de $2°$; la cantidad de memoria requerida para almacenar una transformada de Radon será $60 \times (180°/2°) \times 4 = 21,600$ bytes ó, aproximando, 21KB. La transformada de Fourier requerirá $(60+2) \times (180°/2°) \times 4 = 22,320\text{B} \approx 22\text{KB}$. Y la transformada de Radon del volumen requerirá $(180°/2°) \times 21600\text{B}$, esto es 1.85MB y su transformada de Fourier 1.92MB. En total, menos de 2MB en datos. Por tanto, no parece que el tema de almacenamiento de datos sea importante en este problema.

A continuación pasamos a estudiar los requerimientos de cálculo del proceso. Para ello realizamos un "profiling", o "perfilado", del programa secuencial. En los párrafos siguientes se muestra el contenido del archivo resultante del *perfilado*; la medición es sobre la ejecución de una iteración en el ciclo γ (4(b).ii en el programa principal de la subsec. 5.3.1), realizando sólo la asignación angular y con la especificación de los datos de entrada bajo las mismas condiciones anteriormente supuestas. La búsqueda en el ángulo ψ se realizó en el intervalo $0° \leq \psi \leq 178°$, a un incremento angular de $2°$.

```
 Prof run at: Thu Jun 19 13:20:46 1997
Command line: prof -gprof -pixie ccf_3d


          77819225: Total number of cycles
          0.39704s: Total execution time
          94989403: Total number of instructions executed
             0.819: Ratio of cycles / instruction
               196: Clock rate in MHz
            R10000: Target processor modelled
```

```
-----------------------------------------------------------
Procedures sorted in descending order of cycles executed.
Unexecuted procedures are not listed. Procedures
beginning with *DF* are dummy functions and represent
init, fini and stub sections.
-----------------------------------------------------------

      cycles(%)  cum %     secs    instrns    calls procedure(dso:file)

64289700(82.61)  82.61     0.33   78740100  8100 MRFFT
                                                  (ccf_3d:~/radon/mrfft.f)
 6592042( 8.47)  91.09     0.03    8792706     1 ccf_radons
                                                  (ccf_3d:~/radon/ccf_3d.c)
 3871800( 4.98)  96.06     0.02    4365900  8100 MRFOUR1
                                                  (ccf_3d:~/radon/mrfour1.f)
 2430000( 3.12)  99.18     0.01    2381400 64800 __sinf
                                                  (/usr/lib64/libm.so:fsin.c)
  430515( 0.55)  99.74     0.00     459898   311 memcpy
                                                  (/usr/lib64/libc.so.1:bcopy.s)
   42488( 0.05)  99.79     0.00      50854    94 imgalloc
                                                  (ccf_3d:~/tgc/Lib/tgc.c)
   36535( 0.05)  99.84     0.00      29308     1 do_image
                                                  (ccf_3d:~/radon/ccf_3d.c)
```

Nótese que las funciones que realizan la transformada de Fourier están codificadas en FORTRAN. Este resultado del perfilado es usando las opciones de optimización del compilador y sustituyendo en el código FORTRAN todas las potenciaciones por multiplicaciones simples. Como puede observarse, las primeras cuatro funciones ocupan el 99.18% del tiempo total de ejecución: *MRFFT, ccf_radons, MRFOUR1* y __sinf; *ccf_radons* es la misma que la especificada en la subsección 5.3.1, las otras tres están asociadas a la transformada de Fourier inversa (en 3(a).ii del pseudocódigo de la función ccf_3d). También puede verse que la carga en la ejecución de la función *do_image* es despreciable, en comparación.

El trabajo principal de la función *ccf_radons* es la multiplicación compleja de los coeficientes de Fourier de las líneas de las transformadas de Radon. La carga en la ejecución de la transformada inversa de Fourier es grande debido al número de veces que se realiza la FFT multiradix y a la misma FFT multradix. El número de veces que se realiza la FFT es igual al número ciclos en ψ multiplicado por el número de filas del muestreo angular de la transformada de Radon, como se puede ver en el pseudocódigo de la función *ccf_radons* (en la pág. 62), que equivale al producto irange∗rows; en nuestra prueba es igual a $[(178^\circ - 0^\circ)/2^\circ + 1] * (180^\circ/2^\circ) = 8100$, que es el mismo valor de llamadas a la función que aparece en el resultado del perfilado. La FFT multiradix se realiza por descomposición en factores primos por el algoritmo de Singleton [109], que tiene la ventaja de que las secuencias no tienen que estar fijas a un tamaño que sea una potencia de 2, pero tiene el inconveniente de un mayor tiempo de ejecución.

Así, con estos datos, se decidió centrarse en paralelizar la función *ccf_radons*, distribuyendo las operaciones de cada transformada de Radon entre los procesadores disponibles. No se intentó paralelizar ninguna otra función, al ser *ccf_radons*, claramente, "el cuello de botella" en la ejecución de la aplicación en su totalidad. El objetivo de la paralelización es la reducción del tiempo total de ejecución. En la siguiente sección de tratará en detalle la paralelización, usando la estrategia basada en el paso de mensajes.

5.5 Descripción del Programa Paralelo

El programa que se ejecutará en paralelo es el **ccf_3d**, que como hemos visto anteriormente es el que conlleva mayor tiempo de ejecución. El nombre del programa principal paralelo se llamará *masradon*.

Las condiciones de entrada al programa son las mismas que las usadas para el programa secuencial:

Programa masradon

Uso: Realiza en paralelo el alineamiento rotacional y/o translacional de las transformadas de Fourier de transformadas de Radon.

Sintaxis: masradon NP name_in name_sel beta1 beta2 gamma1 gamma2
psi1 psi2 inc shift search_type

donde:

NP: Número de los procesos hijos en que se ejecutará el programa en paralelo.

Las especificaciones para todos los demás parámetros son iguales que para el programa *ccf_3d* (ver 5.3)

5.5.1 Pseudocódigo del programa paralelo

Como veíamos anteriormente, después del análisis de la sección 5.4, se concluyó que debía paralelizarse la función *ccf_radons* del programa secuencial. También indicamos anteriormente que el método de paralelización iba a ser basado en la estrategia maestro–hijo. De esta forma, las condiciones de entrada para la paralelización del programa secuencial son: se tiene un proceso "maestro" y Nproc procesos "hijos". Este esquema encaja muy bien con la estrategia del código por lo siguiente: si se observa el segundo bucle **for** en 3.a—3.b del pseudocódigo de la función *ccf_radons*, cada operación sobre par de líneas Ka,Kl de las transformadas de Radon es independiente una de otra, por lo que se puede distribuir entre Nproc procesos hijos que realicen una porción rows/Nproc del cálculo.

Según la idea expuesta en el párrafo anterior, el programa en pseudocódigo para el proceso "maestro" queda:

Programa: MAESTRO masradon

1. **read** name_in; /** FT 3D de la RT 3D del volumen **/
2. irange = no. de valores en ψ's = $\frac{\psi_{\text{end}}-\psi_{\text{start}}}{\Delta angles}+1$;
3. **send** datos a los NProc HIJOS: rows, nsam, irange, shift, etc;
4. $\Delta\beta = \Delta angles$;
5. **for each** imagen en name_sel **do**
   (a) **read** image$_k$;
   (b) **send** image$_k$ a los NProc HIJOS;
   (c) **for** $\beta_{\text{start}} \leq \beta_i \leq \beta_{\text{end}}, \quad \beta_{i+1} = \beta_i + \Delta\beta,$ **do**
      i. $\Delta\gamma = \Delta\beta / \operatorname{sen}|\beta|$;
      ii. **for** $\gamma_{\text{start}} \leq \gamma_j \leq \gamma_{\text{end}}, \quad \gamma_{j+1} = \gamma_j + \Delta\gamma,$ **do**
         A. reference_img = do_image (name_in, β_i, γ_j);
         B. **send** reference_img a los NProc HIJOS;
         C. **ccf_radons_parallel**(irange, shift);
            /** Función de correlación cruzada entre la imagen
            de referencia y la imagen leída **/
         D. **find** el máximo de la función de correlación cruzada;
      iii. **endfor**
   (d) **endfor**
6. **endfor**

El proceso "maestro" lee el volumen, manda a los procesos hijos los datos necesarios, lee la imagen a alinear y la envía a los procesos hijos y, para cada par de ángulos β_i, γ_j, calcula la reproyección a partir del volumen y la envía también a los hijos. Así la función para calcular la correlación entre transformadas de Radon bidimensionales pasa a realizarse en paralelo y se denomina ahora *ccf_radons_parallel*.

La función *ccf_radons_parallel()*, en la parte 5(c).ii.C del pseudocódigo del programa MAESTRO masradon (arriba, en esta misma sección), se describe a continuación:

Función MAESTRO ccf_radons_parallel(irange, shift)

1. **if** shift = 0 **then**
   (a) **receive** ff[irange]; /** de cualquier HIJO **/

(b) **for** $1 \leq k <$ Nproc **do**

i. **receive** ff_buff[irange]; /** Los otros HIJOS **/

ii. **for** $0 \leq j <$ irange ff[j] = ff[j] + ff_buf[j];

(c) **endfor**

(d) **for** $0 \leq k <$ irange ff[j] = ff[k]/($\sigma_A \sigma_B$);

2. **else**

(a) **for** $0 \leq k <$ irange **receive** ccf[k][rows][nsam];

(b) **for** $0 \leq k <$ irange **do**

i. ffmax = ff; Δarg = π/rows;

ii. Δarg = π/rows;

iii. **for** $1 \leq n \leq$ shift **do**

- intervals= $8 \cdot n$;
- Δphase = 2π/intervals;
- **for** $0 \leq j \leq$ intervals **do**
  - $\phi = j \cdot \Delta$phase;
  - ff = $\frac{1}{\sigma_A \sigma_B} \sum_{i=0}^{rows-1}$ ccf[i] [$rsam/2 + n \cdot \cos(\phi + \Delta\text{arg} \cdot i$];
  - **if** ff > ffmax **then** /** Resultados **/
    * ffmax = ff;
    * radius = n;
    * phase = ϕ;
  - **endif**
- **endfor** /** End sobre ciclo de fase **/

iv. **endfor** /** End sobre ciclo de la amplitud **/

(c) **endfor**

3. **endif** /** End sobre el bloque de corrimiento **/

La función *ccf_radons_parallel* en el "maestro" realiza, en general, la recolecta de los resultados realizados por los "hijos". Si sólo se realiza el alineamiento rotacional, en el bloque **if—then** en 1, se reciben de los hijos los resultados parciales de C_{i,j,α_k} [Ec. (5.24)]. Si lo que se realiza es el alineamiento translacional, es necesario recibir de los hijos todas las matrices con los resultados de correlación cruzada, correspondientes a cada α_k, como se ve en la segunda parte del bloque **if—then**. Y, por consiguiente, cada proceso "hijo" deberá realizar:

Programa: HIJO masradon

1. **receive** datos del MASTER: rows, nsam, irange, shift, etc;

2. **ccf_3d_parallel**(rows, nsam, irange, shift);

Función HIJO ccf_radons_parallel(rows, nsam, irange, shift)

1. rsam = nsam - 2;
2. **while receive** image **do** /** imagen B[rows][nsam] **/ /** Ciclo sobre ψ **/
   (a) **while receive** imagen de referencia **do** /** imagen A[rows][nsam] **/
      i. **for** $0 \leq Kp \leq$ irange **do** /** Ciclo sobre ψ **/
         - **for** $0 \leq Ki \leq$ merows **do**
           - Kl = meindex + Ki;
           - ccf[Ki][K] = $\mathrm{A}_{Ka}\mathrm{B}^{*}_{Kl}$; /** K es el numero de muestras en A y B **/
           - INV_FT (ccf[Ki], rsam);
           - ff[Kp] = ff[Kp] + ccf[Ki][rsam/2];
         - **endfor**
         - **if** shift $>$ 0 **then**
           send al MAESTRO kp, my_id, ccf[merows][nsam];
         - **endif**

      ii. **endfor**

      iii. **if** shift = 0 **then**
         send al MAESTRO ff[irange];

      iv. **endif**

   (b) **endwhile**
3. **endwhile**

Cada "hijo" realiza la correlación cruzada de sus Ki respectivos renglones.

5.6 Resultados

Los programas secuencial y paralelo se codificaron en lenguaje C. Como ambiente de paso de mensajes se utilizó PVM [111]. Los programas se ejecutaron en una máquina *Power Challenge*, de Silicon Graphics, con seis procesadores R10000. Los resultados pueden verse en la Fig. 5.2.

Las condiciones para el experimento son las mismas que las usadas en la sección 5.4: un volumen de $64 \times 64 \times 64$ voxels y para *una sola* imágen de 64×64 pixels, tomando 60 muestras en las transformadas 3D y 2D de Radon a un muestreo angular de 2°. Los parámetros de entrada al programa fueron: 0°—90° en beta1—beta2, 0°—180° en gamma1—gamma2 y $-90°$—90° en psi1—psi2. Se utilizó un muestreo

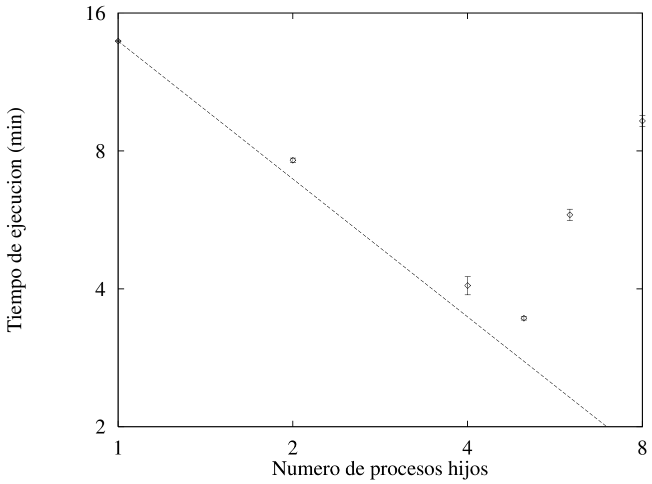


Figura 5.2: Resultados de la medición de tiempos de ejecución vs. el número de procesos hijos de los programas secuencial y paralelo. La línea recta indica la aproximación ideal para la ejecución del programa paralelo. Se presenta la media $+/-$ la desviación estándar de varias mediciones (> 5). Detalles en el texto.

angular de 2°, se consideró shift igual a 0 y, finalmente, se llevó a cabo una búsqueda exhaustiva del mejor coeficiente de correlación entre todas las orientaciones (opción "f").

En la Fig. 5.2 se presenta la media y desviación estándar en el eje-y de varias mediciones del tiempo de ejecución. Se presenta el tiempo total de ejecución, como se sugiere en [22], que involucra también los tiempos en Entrada/Salida, paginado, inicialización del sistema PVM, etc. Las mediciones fueron realizadas bajo una condición de carga "baja" en la plataforma. En la tabla 5.1 se presentan la *aceleración* (speedup) y la *eficiencia* con los datos medios de la Fig. 5.2.

Claramente se puede observar en la Fig. 5.2 que se ha obtenido una reducción muy buena de el tiempo ejecución del programa paralelo hasta tener 5 procesos hijos. La perdida del mejoramiento con seis o más procesos hijos se explica enseguida. Como ya se describió en la sección 4.7 (en el capítulo anterior) nuestro escenario consta de: i) el proceso maestro, ii) los procesos hijos, iii) los procesos del demonio de PVM para el ambiente de paso de mensajes y iv) los demás procesos que incluyen los del sistema operativo y los de otros usuarios (el uso de la máquina es compartida por varios grupos de trabajo). Además, que recordemos que aqui i) es un proceso "maestro" que realiza trabajo importante, entonces tenemos Nproc+1 procesos ejecutándose (los hijos más el maestro) y como estamos corriendolos sobre una plataforma con seis microprocesadores, se deduce, por la tanto, que tenemos el máxima rendimiento en el uso de la plataforma cuando tenemos cinco procesos hijos, o más fácilmente, un proceso de nuestra aplicación por cada procesador disponible. Sin olvidar que al mismo tiempo un microprocesador debe ocuparse también de iii) y/o iv).

No. procesos hijos	Aceleración	Eficiencia
1	1.00	1.00
2	1.82	0.91
4	3.41	0.85
5	4.03	0.81
6	2.39	0.41
8	1.49	0.19

Tabla 5.1: La aceleración y la eficiencia, según el número de procesos hijos, para los resultados de los tiempos de ejecución de la Fig. 5.2. Estos datos de han calculado en base a los tiempos medios.

— Les diré de qué estoy hablando —contestó—: La mayor parte de las distintas clases de poder exigen un gran sacrificio por parte de quien quiera tener ese poder. Hay un aprendizaje, una disciplina que dura años. Cualquiera que sea la clase de poder que se busque. Presidente de la compañía. Cinturón negro de karate. Gurú espiritual. Atleta profesional. Sea lo que sea lo que se persiga, hay que ponerlo en el tiempo, en la práctica, en el esfuerzo. Hay que sacrificar muchas cosas para lograrlo. Tiene que ser muy importante para uno. Y una vez que se alcanza, es el poder de uno mismo; no se puede delegar: reside en uno. Es, literalmente, resultado de nuestra disciplina.

Ahora bien: lo interesante de este proceso es que, en el momento en que alguien adquirió la capacidad de matar con sus manos, también maduró hasta el punto en que sabía como utilizar ese poder. No lo utilizaría de manera imprudente. Así que esa clase de poder lleva una clase de poder incorporado: la disciplina de conseguir el poder cambia a la persona, de manera que esa persona no hace mal uso de su poder.

Pero el poder científico es como la riqueza heredada: se obtiene sin disciplina. Una persona lee lo que otras hicieron, y da el paso siguiente. Puede darlo muy joven. Se puede progresar muy de prisa. No hay una disciplina que dure muchas décadas. No hay una enseñanza impartida por unos maestros: se pasa por alto a los viejos científicos. No hay humildad ante la Naturaleza. Sólo existe la filosofía de hacerse-rico-pronto, hacerse-un-hombre-rápido. Engañar, mentir, falsificar, no importa. Ni para uno ni para sus colegas. Nadie nos critica: nadie tiene pautas. Todos intentan hacer lo mismo: hacer algo grande, y hacerlo rápido.

Y, como uno se puede levantar sobre los hombros de los gigantes, se puede lograr algo con rapidez. Uno ni siquiera sabe con exactitud qué ha hecho, pero ya informó sobre ello, lo patentó y lo vendió. Y el comprador tendrá aun menos disciplina que el científico: el comprador simplemente adquiere el poder, como si fuera cualquier bien de consumo. El comprador ni siquiera concibe que pueda ser necesaria disciplina alguna.

Ian Malcolm
Quinta Iteración. Control.

Capítulo 6

Métodos de Refinamiento No Basados en Prototipo: Nuevas Propuestas

6.1 Estimación del Centroide de las Proyecciones Mediante Segmentación de la Masa

Comenzamos este capítulo recordando que necesitamos dos condiciones básicas para efectuar el proceso de reconstrucción tridimensional de un volumen a partir de un conjunto de imágenes de proyección experimentales: (1) Que las orientaciones de cada una de las proyecciones estén en un mismo marco de referencia y (2) Que todas las proyecciones compartan un mismo origen de coordenadas, que es nuestro caso, se sitúa en el centro de masa del volumen. En esta primera sección nos centraremos en la determinación de (2). Aquí suponemos que (1), los ángulos de inclinación y rotación para las proyecciones, asignados por el método tradicional de la geometría cónica, son correctas. El objetivo de esta sección es, por consiguiente, resolver (tanto como se pueda) el problema de los desplazamientos en el plano. De esta forma, se identificará en la imagen de proyección la localización sobre la que se proyecta el centroide del volumen, puramente a partir de la misma imagen de proyección bidimensional. Se usarán procedimientos clásicos del procesamiento digital de imágenes que, como demostraremos, pueden ser aplicadas a imágenes típicas obtenidas en el microscopio electrónico de transmisión con una razón señal-a-ruido menor que 1. Este trabajo tambin puede verse en De la Fraga, *et al*, (1997) [26].

6.1.1 Descripción del Método

El objetivo del método es identificar en la proyección aquellos pixels que contienen señal (esto es, aquellos pixels para los cuales la correspondiente integral de línea del volumen podrían no ser cero, aún en el caso libre de ruido). Se usan solamente estos pixels para estimar la localización del centroide del volumen (el cual es, en el caso sin ruido, el centroide de la proyección). Como paso final, se checa la consistencia de una proyección particular con el resto, de manera que se rechacen aquellas proyecciones (previo a la reconstrucción) que contienen demasiado ruido para que esta propuesta funcione.

Dado un grupo de imágenes que representan proyecciones del volumen (p.e., Fig. 6.2(a)), todos los pasos siguientes son aplicados a cada imagen.

1. Se inicia aplicando a la imagen original $f(x,y)$ el suavizado:

$$g_1(x,y) = \frac{1}{9}\sum_{i=-1}^{1}\sum_{j=-1}^{1} f(x+i, y+j) \qquad \text{para} \quad 0 < x < N-1, \quad 0 < y < N-1 \tag{6.1}$$

   donde N es el tamaño del lado de las imágenes de forma cuadrada que se han usado (ver la Fig. 6.2(b)).

2. Se calcula sobre la imagen original $f(x,y)$ la media y la desviación estándar de la región de fondo, considerada como ruido, m_b, σ_b, y de la región del objeto m_b, σ_b; como se puede observar en la Fig. 6.1, la zona del objeto está entre los radios r_1 y r_2, y la zona de fondo esta en el borde, como se especifica con el parámetro l. Entonces se encuentra un valor m tal que

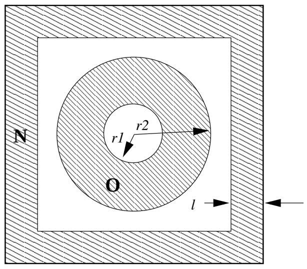


Figura 6.1: Regiones en la imagen

$$\frac{m_o - m}{\sigma_o} = \frac{m - m_b}{\sigma_b} \tag{6.2}$$

   Y se genera una nueva imagen $g_2(x,y)$ como sigue:

$$g_2(x,y) = \begin{cases} 0 & \text{si} \quad g_1(x,y) \geq m_o, \\ 1 & \text{si} \quad \frac{m_o+m}{2} \leq g_1(x,y) < m_o, \\ 2 & \text{si} \quad m \leq g_1(x,y) < \frac{m_o+m}{2}, \\ 3 & \text{si} \quad \frac{m_b+m}{2} \leq g_1(x,y) < m, \\ 4 & \text{si} \quad m_b \leq g_1(x,y) < \frac{m_b+m}{2}, \\ 5 & \text{si} \quad g_1(x,y) < m_b. \end{cases} \tag{6.3}$$

   Un ejemplo de ésta nueva imagen de seis tonos se puede ver en la Fig. 6.2(c).

   En este paso se puede entender la esencia de este método: necesitamos sólo las estimaciones de lo que representa señal y de lo que representa ruido, no la definición precisa de ellas. En otras palabras, nuestras zonas de lo que se considera "señal" ó "ruido" no deben delimitar exactamente al objeto o al fondo, sino sólo una parte representativa de ellos, la suficiente para tener una "estimación" de los mismos. Cabe puntualizar aqui también que esto es lo que nos permite luchar con la gran cantidad de ruido presente en las imágenes típicas de criomicroscopía.

3. Se binariza la imagen $g_2(x, y)$ de la Ec. (6.3) aplicando la siguiente subrutina. Como entrada a la subrutina tenemos la imagen $g_2(x, y)$. Como salida tendremos una nueva imagen $g_3(x, y)$.

   for $1 \leq s \leq 5$ **do**

   (a) **if** $g_2(x, y) = 0$ or 5 **then** **set** $g_3(x, y) = g_2(x, y)$;

   (b) **else**
   Calcular el promedio, $\text{ave}(x, y)$, de los cuatro pixels más cercanos a x, y

   $$\text{ave}(x, y) = \frac{1}{5}\left[\sum_{i=-1}^{1} g_2(x+i, y) + g_2(x, y-1) + g_2(x, y+1)\right] \qquad (6.4)$$

      i. **if** $\text{ave}(x, y) \leq 0.5s$ **then** **set** $g_3(x, y) = 0$;
      ii. **else if** $\text{ave}(x, y) \geq 5 - 0.5s$ **then** **set** $g_3(x, y) = 5$;
      iii. **else** **set** $g_3(x, y) = g_2(x, y)$;
      iv. **endif**

   (c) **endif**

   (d) **if** $s \neq 5$ **then** **Copy** imagen $g_3(x, y)$ a la imagen $g_2(x, y)$

   (e) **endif**

   endfor

   Al finalizar la subrutina tenemos la imagen binaria $g_3(x, y)$ que, como un ejemplo, podemos verla en la Fig. 6.2(d).

4. Una máscara se extrae como se explica a continuación. Se busca un pixel en la zona comprendida entre r_1 y r_2 (ver la Fig. 6.1) para el cual g_3 tenga un valor de 0. Si la búsqueda tiene éxito, entonces se usa una modificación del algoritmo *"Filling by connectivity"* [82] para encontrar el componente de pixels con valor 0 en g_3 que contiene el pixel encontrado. Si este componente está sobre el borde la imagen, entonces se considera que no hay máscara. De otro modo, se asignan a los pixels dentro de la máscara el valor de 1 y todos los demás pixels son puestos a 0. La imagen resultante puede ser vista en la Fig. 6.2(e).

5. Se dilata la máscara agregando a esta todos los puntos que están originalmente fuera de ella, pero adyacentes-4 a ellos (los cuatro pixels vecinos, horizontal y verticalmente). El resultado de la dilatación es ilustrado en la Fig. 6.2(f).

6. Finalmente, se calcula el centroide de la máscara resultante [97]

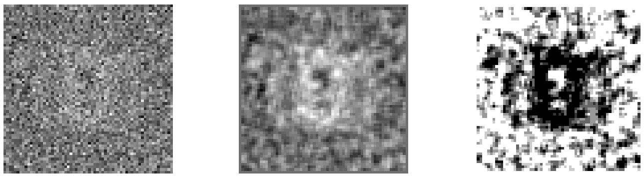
$$\text{C.G.}_{\cdot x} = \frac{\sum_i g_1(x_i, y_i) \cdot x_i}{\sum_i g_1(x_i, y_i)} \qquad \text{C.G.}_{\cdot y} = \frac{\sum_i g_1(x_i, y_i) \cdot y_i}{\sum_i g_1(x_i, y_i)} \tag{6.5}$$

   donde g_1 es la imagen suavizada y x_i, y_i corresponden a las posiciones donde los valores de los pixels en la imagen final, g_f, son iguales a 1.

7. La clasificación de todas las máscaras. Se usan dos medidas para realizarla: El Factor de Forma (F.F.) y el Diámetro Equivalente (D.E.), definidas como

$$\text{F.F.} = \sqrt{\frac{1}{Area}\left[\sum_i (x_i - \text{C.G.}_{\cdot x})^2 + \sum_i (y_i - \text{C.G.}_{\cdot y})^2\right]} \qquad \text{E.D.} = \sqrt{\frac{4}{\pi} Area} \tag{6.6}$$

   Y se escoge *la mayor agrupación* que aparece en un mapa representando F.F. versus D.E. para cada una de las imágenes.



(a) Imagen original (b) Suavizada (c) Seis tonos

(d) Refinada (e) Extraída (f) Dilatada

Figura 6.2: Un ejemplo de la aplicación de todo el proceso. Detalles en el texto.

6.1.2 Experimento

Se comienza con tres volúmenes "fantasmas". Un *fantasma* es un volumen sintético de prueba, con formas geométricas (cilindros o elipsoides) de una densidad conocida (igual a 1, en nuestro caso).

1. CCT, mostrado en la Fig. 6.3(a), es un modelo sintético que está inspirado en el aspecto de las imágenes de proyección experimentales de la proteína CCT [74].

2. BCI, en la Fig. 6.3(b), que es un modelo arbitrario de una proteína gruesa, y

3. fBCI, en la Fig. 6.3(c), que es una versión delgada del fantasma BCI.

El fantasma CCT presenta una simetría cilíndrica de orden 8. BCI y fBCI presentan una simetría cilíndrica de orden 4. Ésta característica será útil posteriormente para obtener una mejor reconstrucción.

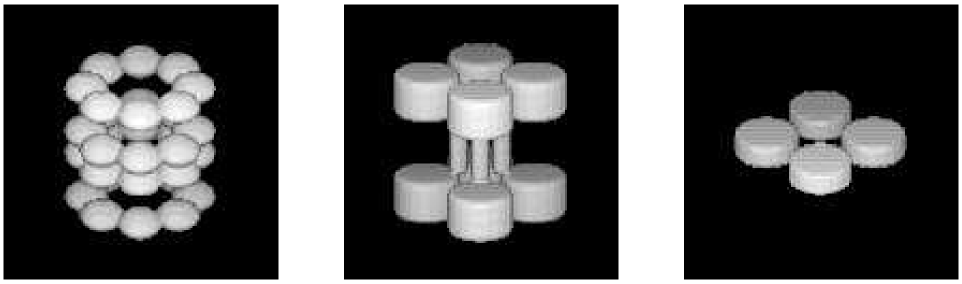
Por cada fantasma se tomaron 65 reproyecciones a 55° en el ángulo de inclinación y $n \cdot 2\pi/65$, $0 \leq n < 65$, en el ángulo azimutal (de acuerdo a una geometría de inclinación cónica). Se sumó a cada imagen reproyectada un ruido Gausiano con objeto de obtener un nivel fraccional de ruido (nfr) de 0.80 [41]: El nivel de ruido adicionado a cada proyección está expresado como la razón entre la desviación estándar del ruido sumado y el promedio de las amplitudes en la intensidad para todas las imágenes simuladas. El nfr=0.80 corresponde aproximadamente a una relación señal-ruido de 0.30. Una muestra de las reproyecciones sin inclinar, sin ruido, para cada fantasma pueden verse en la primera fila de la Fig. 6.4; las reproyecciones inclinadas están en la segunda fila de la misma figura 6.4. Estas imágenes más el ruido adicionado se muestran en la tercera fila de la Fig. 6.4. La Fig. 6.2(a) es la misma que la segunda imagen de la tercera fila de la Fig. 6.4.

A cada grupo de 65 imágenes inclinadas con ruido, se aplicó el método descrito en la subsección 6.1.1.
Finalmente, y como una prueba última de la consistencia de todo el procedimiento, con cada grupo de imágenes centradas en las coordenadas halladas para el centroide, se realizó una reconstrucción tridimensional usando el algoritmo "weighted back-projection" [89].

El método aquí propuesto fue programado usando el lenguaje de programación C-ANSI. EL paquete de software XMIPP [73] fue usado para realizar los fantasmas, las reproyecciones y las reconstrucciones. El paquete WEB [51] fue usado para hacer las visualizaciones de superficie en las Figs. 6.3 y 6.5.

6.1.3 Resultados

Para examinar la bondad de los resultados obtenidos por el método de alineamiento translacional expuesto en este capítulo, se compararon estos nuevos resultados con aquellos obtenidos por el método del alineamiento *estiramiento coseno*, usado habitualmente para alinear las imágenes inclinadas a partir de las imágenes sin inclinar. Los errores fueron calculados usando la Ec. (6.7), esto es, el error cuadrático medio entre la localización correcta del centroide y aquellas coordenadas calculadas por



(a) CCT (b) BCI (c) fBCI

Figura 6.3: Visualización de superficie de los fantasmas

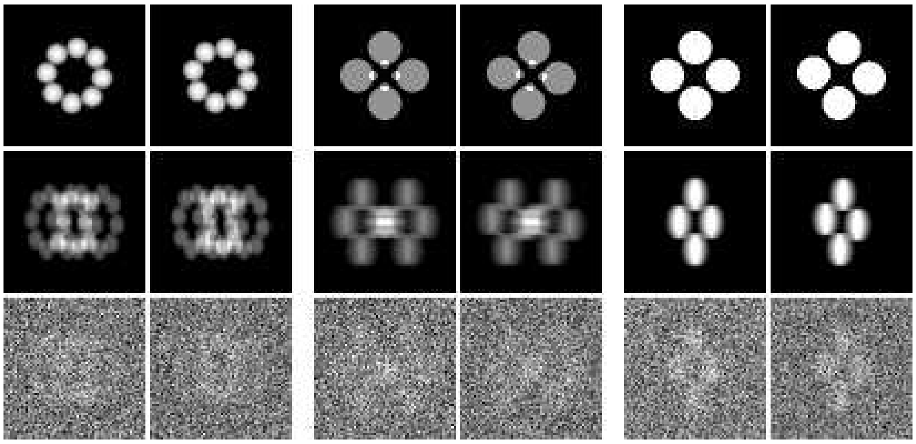
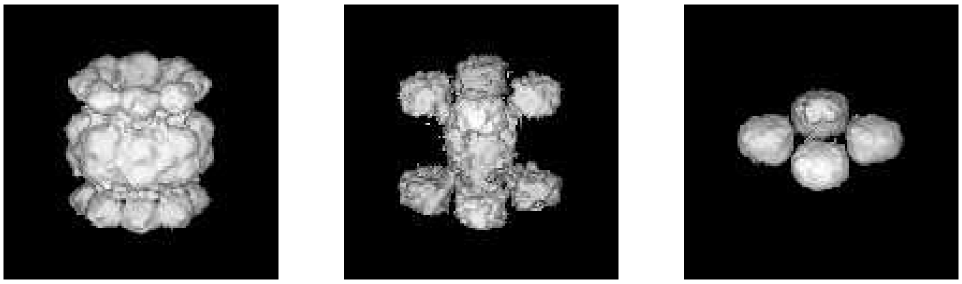
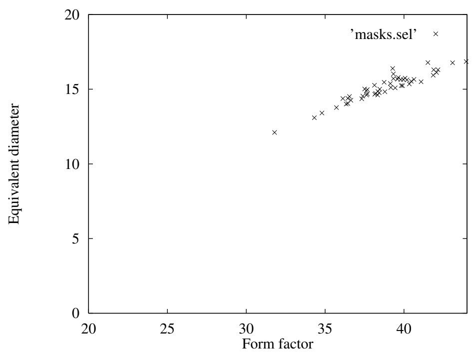


Figura 6.4: Muestras de imágenes de proyección obtenidas a partir de los fantasmas representados en la Fig. 6.3. La primera fila muestra unas imágenes sin inclinar y sin ruido, la segunda fila imágenes inclinadas a 55° sin ruido, y la tercera fila resulta de adicionar a las imágenes de la segunda fila un ruido Gaussiano a un nfr=0.80.

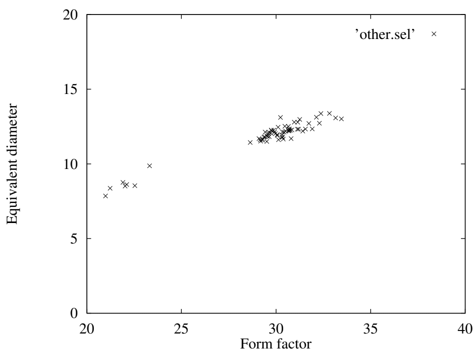


(a) CCT (b) BCI (c) fBCI

Figura 6.5: Reconstrucciones 3D de los fantasmas



(a) caso de la CCT



(b) caso de la fBCI

Figura 6.6: Resultados de la clasificación

cada uno de los métodos.

$$\text{error} = \frac{(val_{ok} - val_{new})^2}{N} \tag{6.7}$$

Un cuadro resumen de estas medidas se error se muestra en la Tabla 6.1 es el número final de imágenes en el proceso. La clasificación de las máscaras solo rechazó imágenes en en el caso de la fBCI. En la Fig. 6.6(a) se pude ver que todas las imágenes forman una sola agrupación. Esto fue también cierto para el caso del fantasma BCI. Sin embargo, para la fBCI (Fig. 6.6(b)) se ven claramente dos agrupaciones. En este caso, se escoge como agrupación "correcta" la mayor de ellas (el situado arriba a la derecha).

Espécimen	Error con máscaras de correlación	Error con nuestro método	N
CCT	$dx = 42.215$ $dy = 2.809$	$dx = 1.687$ $dy = 0.594$	51
BCI	$dx = 236.735$ $dy = 5.173$	$dx = 1.197$ $dy = 0.875$	57
fBCI	$dx = 1.825$ $dy = 0.881$	$dx = 0.240$ $dy = 0.478$	57

Tabla 6.1: Resumen de errores en el centrado de imágenes con ruido obtenidos por el método tradicional del estiramiento coseno y el propuesto en esta memoria.

La Tabla 6.1 muestra que las estimaciones de los centroides del volumen en proyecciones muy ruidosas, que se obtienen por este nuevo método, son claramente superiores a aquellas obtenidas por el procedimiento del estiramiento coseno.

Concluyendo, el nuevo método de alineamiento translacional desarrollado aquí, nos permite generar un volumen (si al menos tenemos aproximados los orientaciones de las proyecciones experimentales) que puede ser usado como un prototipo por algún método de refinamiento de la geometría como los expuestos en el capítulo 3. Se ha demostrado que el método trabaja para imágenes con muy baja relación señal-ruido.

6.2 Refinamiento de los Ángulos de Inclinación y de Rotación

6.2.1 Proceso de Refinamiento

Problema.- Supongamos que hay un grupo de imágenes con φ como el ángulo de inclinación. Hay J rotaciones, definidas por θ_j, para el grupo. Esto nos definen los parámetros de una geometría cónica aleatoria.

Objetivo.- Para el grupo completo de imágenes, identificar el φ y encontrar un subgrupo consistente de θ_j "correctas", esto es, si los valores ha encontrar de las θ_j no son exactamente los correctos, deben ser lo más cercano posible a ellos.

El problema podemos especificarlo en otras palabras: si tenemos un grupo de imágenes de proyección experimentales que forman una geometría de inclinación cónica aleatoria y cuyos valores de los ángulos asignados están cerca de los valores correctos, queremos un procedimiento que nos minimize el error actual en los valores de los ángulos.

Para la solución del *problema*, hacemos uso de las siguientes consideraciones:

1. La amplitud del ángulo de inclinación es: $0 < \varphi < \frac{\pi}{2}$
2. La amplitud de los ángulos de rotación es: $0 < \theta < 2\pi$
3. El ángulo donde se encuentra la línea común (ver la Sec. 2.8.1) entre la imagen con θ_j y la imagen con θ_k (para $1 \leq j \leq J,\ j < k \leq J,\ j \neq k$) es:
$$\rho(\varphi, \theta_j, \theta_k)$$
ρ queda definido en $-\frac{\pi}{2} < \rho \leq \frac{\pi}{2}$.
ρ es el ángulo donde la proyección de la imagen j es igual a la proyección de la imagen k al mismo valor del ángulo ρ, pero negativo. La demostración de esta aseveración se tratará en la subsección 6.2.2.
4. Para los cálculos consideramos los ángulos discretizados con una resolución $\delta = \pi/2048 \approx 0.1°$. Así, $10°$ es igual a $10°(2048/180°) = 114$ *unidades delta* (ud).

Solución:

1. Calcular ρ_{jk}, esto es, el ángulo donde se halla la línea común entre las imágenes j y k con ángulos de rotación θ_j y $theta_k$, respectivamente.
$$\rho_{jk} = \rho(\varphi, \theta_j, \theta_k), \quad 1 \leq j < J,\ j < j \leq J \tag{6.8}$$
Las ρ's también cumplen la condición:
$$\rho_{kj} = -\rho_{jk} \tag{6.9}$$

2. Crear una tabla (una matriz triangular superior de tamaño $\frac{J(J-1)}{2}$). Definiendo:

$$\mathrm{D}(\rho, j, k) = \sum_{r \text{ rayos}} [\mathrm{p}_j(\rho - \pi/2, r) - \mathrm{p}_k(-\rho - \pi/2, r)]^2 \tag{6.10}$$

como la distancia cuadrática entre las proyecciones de las imágenes j y k. El mínimo de la función D de la Ec. (6.10) corresponde al lugar donde se halla la línea común entre las imágenes j y k. La función D es de suponerse que sea bastante suave en su forma (debido a nuestro muestreo), al menos para un intervalo alrededor del sitio exacto donde se encuentra la línea común. Para hallar el mńimo de la Ec. (6.10) se hace una búsqueda alredor del lugar donde se supone (pero no con certeza) que se encuentra la línea común. Aqui suponemos que una busqueda alrededor de ± 64 ud ($= 5.625°$) es suficiente para corregir las desviaciones angulares [1]. Para hallar el mńimo de la función D de la Ec. (6.10), primero realizamos lo siguiente:

Para cada par $j, k \quad (1 \leq j < J,\ j < k \leq J)$, se calcula:

$$D(\rho_{jk} - 64\delta, j, k), D(\rho_{jk} - 32\delta, j, k), D(\rho_{jk}, j, k), D(\rho_{jk} + 32\delta, j, k), D(\rho_{jk} + 64\delta, j, k) \tag{6.11}$$

si el valor más pequeño de estos cinco es una de las posiciones extremas, hacemos:

$$D_{jk} = \mathrm{MARCA}$$

esto es, ponemos en la localidad j, k de la tabla una “marca” que indica que para ese par de imágenes no puede calcularse (no existe) el valor correspondiente a donde se halla la línea común.

3. Remover la imagen con más marcas en la tabla de las D’s y remover el renglón o columna correspondiente. Si hay igual número de marcas en algún renglón y columna, entonces remover la columna.
Actualizar el valor de J.

4. Encontrar el valor más pequeño de D_{jk} (dentro del intervalo especificado en el paso 2), a este valor corresponde $\mathring{\rho}_{jk}$. Este valor representa el lugar “exacto”, dentro de nuestra resolución, donde se halla la línea común. Poner este valor en otra tabla. Calcular

$$M = \frac{2}{N(N-1)} \sum_{j=1}^{N-1} \sum_{k=j+1}^{N} \left(\rho_{jk} - \mathring{\rho}_{jk}\right)^2 \tag{6.12}$$

[1] Aunque las desviaciones angulares pueden ser de hasta 20°, aqui lo que nos interesa es tener “las mejores” partículas que generan una geometría cónica aleatoria, y generar con ellas la mejor reconstrucción 3D inicial. Las imágenes con una desviación angular mayor pueden agregarse al modelo inicial usando alguno de los métodos de refinamiento con prototipo explicados en el capítulo 3.

A partir de ahora ya no se utilizarán los valores de la tabla de D's, por lo tanto se borra.

La búsqueda del valor mínimo de D no se realiza con una búsqueda exaustiva. Se realiza de la manera siguiente: de los cinco puntos de D calculados en el paso 2, se toma el mínimo (si no es un punto extremo y por la tanto se haya desechado) y sus dos extremos, se calculan otros dos valores a $\pm 16\delta$ del punto mínimo para tener de nuevo cinco valores. Se busca de nuevo el valor mínimo de estos nuevos cinco valores y se repite el procedimiento hasta la resolución que tenemos (de 1 ud). Esto tiene la ventaja de que no necesitamos calcular todos los valores (dentro del intervalo) de la función D. Claro que esto funciona bajo la consideración de que la forma de la función D es suave.

5. Estimar los valores de φ para cada par j, k y $\mathring{\rho}_{jk}$

$$\phi_{jk} = \phi(\mathring{\rho}_{jk}, \theta_j, \theta_k) \tag{6.13}$$

6. Encontrar las tres imágenes más típicas (más representativas) de todo el conjunto, eliminando cada vez la imagen más atípica, como sigue: Eliminar la imagen l que maximize:

$$\left| \frac{2}{N(N-1)} \sum_{j \neq k} \varphi_{jk} - \frac{2}{(N-1)(N-2)} \sum_{\substack{j \neq k \\ j \neq l \\ k \neq l}} \varphi_{jk} \right| \tag{6.14}$$

Iniciando con $N = J$, $1 \leq j < N$, $j < k \leq N$, se repite el proceso para $N \leftarrow N - 1$ hasta que queden sólo tres imágenes ($N = 3$).

Dentro de este mismo paso, las imágenes se van ordenando por "tipicalidad". Al terminar el proceso, la imagen más atípica es la correspondiente a $j = J$.

Actualizar el valor de φ a:

$$\mathring{\varphi} = \varphi_{12} \quad \text{ó} \quad \varphi_{13} \quad \text{ó} \quad \varphi_{23} \tag{6.15}$$

En este punto, las tres imágenes más típicas y los $\rho(j,k)$ asociados (hay tres de ellos) determinan inequívocamente a φ, $\theta_2 - \theta_1$, $\theta_2 - \theta_1$. Finalmente, insistiendo en que θ_1, θ_2 y θ_3 concuerdan con los valores inicialmente estimados, tenemos los valores de φ, θ_1, θ_2 y θ_3. Aquí esta el fondo de este procedimiento: *tres orientaciones pueden siempre ser descritas como tres rotaciones alrededor de un solo eje de inclinación.* En otras palabras, si las tres imágenes más típicas tienen los valores de la θ's correctos, entonces cualquier estimación de ϕ nos debe dar el valor correcto del ángulo de inclinación.

7. Ahora supongamos que tenemos un grupo estimado de $\varphi,\ \theta_1,\ \theta_2,\ldots,\theta_n$. Para una nueva imagen $k=n+1$, recalcular ρ (como en el paso 1). Si para algún j, sucede que:

$$\left|\rho_{jk}-\mathring{\rho}_{jk}\right|\geq 64\delta \tag{6.16}$$

entonces:

PARAR el proceso.

64δ es el intervalo máximo de búsqueda en el valor angular, como se especificó anteriormente en al paso 2. de otro modo calcular M' (sobre j) de manera semejante a la Ec. (6.12) del paso 4. Si se cumple que M' es mayor que M:

PARAR el proceso.

de otro modo calcular

$$\overline{\theta}=\frac{1}{n}\sum_{j=1}^{n}\theta_{n+1}(\mathring{\varphi},\mathring{\rho}_{jk},\theta_j) \tag{6.17}$$

y actualizar el valor de $\mathring{\varphi}$

$$\mathring{\varphi}_{n+1}=\frac{m_n\mathring{\varphi}_n+\sum_{j=1}^{n}\varphi_{jk}}{m_n+n},\qquad \text{para } n>3 \tag{6.18}$$

donde

$$m_n=\frac{(n-1)(n-2)}{2}$$

Y la $\mathring{\varphi}_3$ inicial es la calculada en la Ec. (6.15).

Ecuaciones:

La ecuación (6.8) es calculada como:

$$\rho=\arctan\left(\frac{\cos\varphi(1-\cos\theta)}{\operatorname{sen}\theta}\right) \tag{6.19}$$

Para esta ecuación: $\theta_j=0,\ \theta=\theta_k$.

La ecuación (6.13) es calculada despejando φ de la ecuación (6.19):

$$\varphi=\arccos\left(\frac{\cos\varphi(1-\cos\theta)}{\operatorname{sen}\theta}\right) \tag{6.20}$$

Y la ecuación (6.17) es calculada haciendo $\cos\theta=\sqrt{1-\operatorname{sen}^2\theta}$ en la ecuación (6.19):

$$\theta=\arcsin\left(\frac{2a}{1+a^2}\right)\qquad \text{donde}\quad a=\frac{\tan\rho}{\cos\varphi} \tag{6.21}$$

6.2.2 Demostración de las Ecuaciones

En esta sección se desarrollará la teoría para la demostración de las ecuaciones de la sección anterior. Por convención, las ecuaciones se derivarán usando el sistema de coordenadas en 3D de *mano derecha*. Aunque puede demostrarse fácilmente que las ecuaciones derivadas son válidas también para un sistema de coordenadas 3D de mano izquierda.

El vector unitario η

En la Fig. 6.7 podemos ver la situación del vector unitario η en el sistema de coordenadas 3D. Este queda determinado por la ecuación:

$$\vec{\eta}(\varphi, \theta) = (\eta_x, \eta_y, \eta_z) \tag{6.22}$$

donde el valor de los componentes vienen expresados por:

$$\begin{aligned}
\eta_x &= \cos\theta \ \operatorname{sen}\varphi \\
\eta_y &= \operatorname{sen}\theta \ \operatorname{sen}\varphi \\
\eta_z &= \cos\varphi
\end{aligned}$$

Sustituyendo los valores de los componentes en la Ec. (6.22), obtenemos la ecuación que describe al vector $\vec{\eta}$:

$$\vec{\eta}(\varphi, \theta) = (\cos\theta \operatorname{sen}\varphi,\ \operatorname{sen}\theta \operatorname{sen}\varphi,\ \cos\varphi) \tag{6.23}$$

El plano P

La figura 6.8 representa un plano, llamado **P**, en el sistema de coordenadas 3D. La ecuación que representa al plano, en forma vectorial, es:

$$\mathbf{P}(\varphi, \theta) = u \cdot \vec{x}(\varphi, \theta) + v \cdot \vec{y}(\varphi, \theta) \tag{6.24}$$

Los vectores unitarios $\vec{x}$ y $\vec{y}$ pueden calcularse como sigue (observe la Fig. 6.8):

$$\begin{array}{ll}
\vec{x}(0,0) = (1,\ 0,\ 0) & \vec{y}(0,0) = (0,\ 1,\ 0) \\
\vec{x}(0,\theta) = (\cos\theta,\ \operatorname{sen}\theta,\ 0) & \vec{y}(0,\theta) = (-\operatorname{sen}\theta,\ \cos\theta,\ 0) \\
\vec{x}(\varphi,\theta) = (\cos\varphi\cos\theta,\ \cos\varphi\operatorname{sen}\theta,\ -\operatorname{sen}\varphi) & \vec{y}(\varphi,\theta) = (-\operatorname{sen}\theta,\ \cos\theta,\ 0)
\end{array}$$

y sustituyendo las ecuaciones anteriores en la Ec. (6.24) obtenemos la ecuación que describe al plano **P**.

$$\mathbf{P}(\varphi, \theta) = (u\cos\varphi\cos\theta - v\operatorname{sen}\theta, u\cos\varphi\operatorname{sen}\theta + v\cos\theta, -u\operatorname{sen}\varphi) \tag{6.25}$$

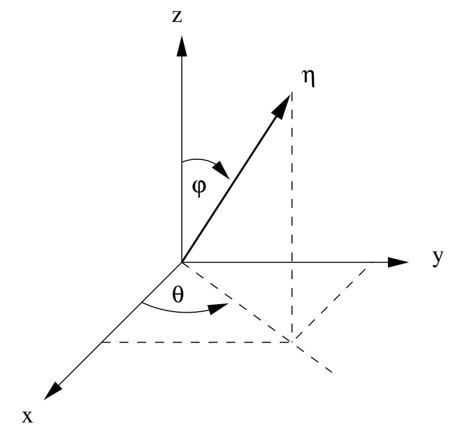


Figura 6.7: El vector unitario η

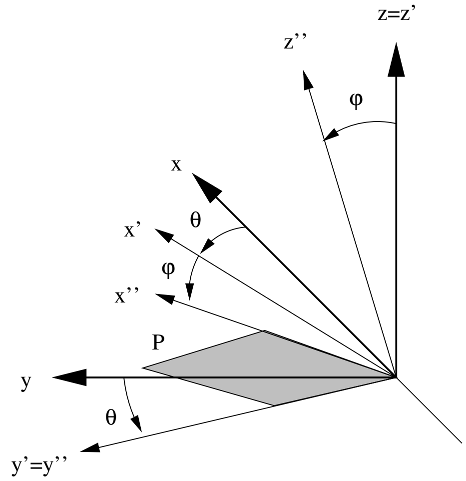


Figura 6.8: El plano **P**

La intersección de dos planos

Cuando se intersectan dos planos hay sólo una línea común entre ellos y sólo una. Si dos planos tuviesen más de una línea común, ambos planos serían, en realidad, uno solo. De esta manera, el vector normal al plano $\mathbf{P}_1$, $\vec{\eta}_1$, debe ser ortogonal al plano $\mathbf{P}_2$ a lo largo de la línea común. Así mismo, el vector normal al plano $\mathbf{P}_2$ debe ser ortogonal al plano $\mathbf{P}_1$ a lo largo de la línea común. De lo anterior se deducen las siguientes ecuaciones:

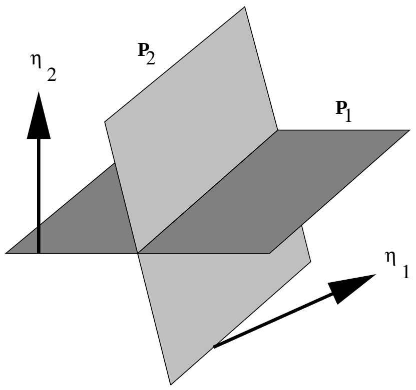


Figura 6.9: La intersección de dos planos y los vectores normales a ellos

$$\mathbf{P}_1(\varphi, \theta_1) \cdot \vec{\eta}_2(\varphi, \theta_2) = 0 \tag{6.26}$$

$$\mathbf{P}_2(\varphi, \theta_2) \cdot \vec{\eta}_1(\varphi, \theta_1) = 0 \tag{6.27}$$

Sin perder generalidad, se puede hacer $\theta_1 = 0$ y $\theta_2 = \theta$ en las ecuaciones (6.26) y (6.27). Obteniendo entonces:

$$\mathbf{P}_1(\varphi, 0) \cdot \vec{\eta}_2(\varphi, \theta) = 0 \tag{6.28}$$

$$\mathbf{P}_2(\varphi, \theta) \cdot \vec{\eta}_1(\varphi, 0) = 0 \tag{6.29}$$

Ahora, sustituyendo los valores de $\theta_1 = 0$ y $\theta_2 = \theta$ en las Ecs. (6.23) y (6.25), para obtener los valores de $\mathbf{P}_1$ y $\vec{\eta}_2$ de la Ec. (6.28), tenemos:

$$\mathbf{P}_1(\varphi, 0) = (u_1 \cos\varphi,\ v_1,\ -u_1 \operatorname{sen}\varphi)$$

$$\vec{\eta}_2(\varphi, \theta) = (\cos\theta \operatorname{sen}\varphi,\ \operatorname{sen}\theta \operatorname{sen}\varphi,\ \cos\varphi)$$

y sustituyendo las dos ecuaciones anteriores en la Ec. (6.28) obtenemos:

$$
\begin{gathered}
(u_1 \cos\varphi,\ v_1,\ -u_1 \operatorname{sen}\varphi) \cdot (\cos\theta \operatorname{sen}\varphi,\ \operatorname{sen}\theta \operatorname{sen}\varphi,\ \cos\varphi) = 0 \\
u_1 \cos\varphi \cos\theta \operatorname{sen}\varphi + v_1 \operatorname{sen}\theta \operatorname{sen}\varphi - u_1 \operatorname{sen}\varphi \cos\varphi) = 0 \\
u_1 \cos\varphi(\cos\theta - 1) + v_1 \operatorname{sen}\theta = 0
\end{gathered}
$$

De esta forma llegamos a:

$$
\frac{v_1}{u_1} = \frac{\cos\varphi(1 - \cos\theta)}{\operatorname{sen}\theta} \tag{6.30}
$$

De manera similar, sustituyendo los mismos valores $\theta_1 = 0$ y $\theta_2 = \theta$ en las ecuaciones (6.23) y (6.25) para obtener los valores de $\mathbf{P}_2$ y $\vec{\eta}_1$ de la Ec. (6.29), obtenemos:

$$
\begin{aligned}
\mathbf{P}_2(\varphi, \theta) = (&u_2 \cos\varphi \cos\theta - v_2 \operatorname{sen}\theta, \\
&u_2 \cos\varphi \operatorname{sen}\theta + v_2 \cos\theta, \\
&- u_2 \operatorname{sen}\varphi) \\
\vec{\eta}_1(\varphi, 0) = (&\operatorname{sen}\varphi,\ 0,\ \cos\varphi)
\end{aligned}
$$

y sustituyendo las dos ecuaciones anteriores en la Ec. (6.29) obtenemos:

$$
\begin{gathered}
\operatorname{sen}\varphi(u_2 \cos\varphi \cos\theta - v_2 \operatorname{sen}\theta) - u_2 \cos\varphi \operatorname{sen}\varphi = 0 \\
u_2 \cos\varphi(\cos\theta - 1) - v_2 \operatorname{sen}\theta = 0
\end{gathered}
$$

Finalmente obtenemos:

$$
\frac{v_2}{u_2} = \frac{\cos\varphi(\cos\theta - 1)}{\operatorname{sen}\theta} \tag{6.31}
$$

Llegamos finalmente a la relación de las ecuaciones (6.30) y (6.31), la cual es:

$$
\frac{v_1}{u_1} = -\frac{v_2}{u_2} \tag{6.32}
$$

La Ec. (6.32) nos dice que la pendiente de la línea común sobre el plano $\mathbf{P}_2$ es el valor negativo de la pendiente de la línea común sobre al plano $\mathbf{P}_1$. Por lo tanto, se puede reescribir la Ec. (6.32) como:

$$
\begin{gathered}
\arctan\left(\frac{v_1}{u_1}\right) = -\arctan\left(\frac{v_2}{u_2}\right) \\
\rho(\varphi, \theta_1, \theta_2) = -\rho(\varphi, \theta_2, \theta_1)
\end{gathered} \tag{6.33}
$$

La Ec. (6.33) es exactamente la misma que la Ec. (6.9) en la pag. 82.

6.2.3 Estabilidad del Proceso

En las primeras pruebas del proceso descrito en la sección 6.2.1 detectamos un error en la función *acos()* proporcionada como libreria matemática por el compilador de lenguaje C. A continuación describiremos cuando ocurría ese error, así como la solución al mismo. Invariablemente este error anunciaba alguna inestabilidad del proceso que debía de ser corregida.

El error ocurría en el paso 5 del proceso (en la pag. 84) cuando se estimaban los valores de φ para algún par de imágenes. Para explicarlo, se escogerá un caso específico; tengamos dos imágenes con valores en los ángulos:

$$\begin{aligned} &\theta_1 = 78.750^\circ \\ &\theta_2 = 84.375^\circ, \qquad \text{y} \\ &\varphi = 55^\circ \end{aligned}$$

Podemos ver que $\theta_2 - \theta_1 = \theta = 5.625^\circ$. Para estos valores, la línea común está en el ángulo $\mathring{\rho} = 1.614^\circ$ de la imagen 1 (correspondiente a θ_1).

Ahora, estudiemos la influencia de sumar un cierto *ruido* al valor de los ángulos que determinan la orientación de estas imágenes, pasando a tener:

$$\begin{aligned} &\theta_1 = 79.603^\circ \\ &\theta_2 = 82.378^\circ, \qquad \text{y,} \\ &\varphi = 60^\circ \\ &\theta_2 - \theta_1 = \theta = 2.775^\circ \end{aligned}$$

para estos valores la línea común se encuentra en el ángulo $\rho = 0.694^\circ$. Aún con estos valores con ruido, el proceso es capaz de encontrar el valor correcto para ρ, esto es 1.614° en la imagen 1. El problema se plantea al calcular φ.

Si recordamos la ecuación (6.20) para realizar el cálculo de la estimación de φ, podemos ver que esa misma ecuación se puede reescribir como:

$$\begin{aligned} \cos\varphi &= \tan\rho \frac{\operatorname{sen}\theta}{1-\cos\theta} \\ \cos\varphi &= \tan\rho\, g(\theta) \end{aligned} \tag{6.34}$$

En la ecuación (6.34) se puede ver que el producto de las funciones $\tan\rho$ y $g(\theta)$ debe tener la misma restricción que la función coseno, esto es, $|\cos x| \leq 1$.

En el paso 5 (pag. 84) del proceso se estiman nuevos valores de φ para cada par de imágenes. φ depende de la localización exacta de ρ y de los valores con ruido de las θ's. Es en esta situación donde ocurre el error en nuestro ejemplo. En efecto, si aplicamos directamente la Ec. (6.34), obtenemos:

$$\begin{aligned} \cos\varphi_{12} &= (\tan 1.614^\circ) g(2.775^\circ) \\ &= (0.028177)(41.286183) = 1.1633 \end{aligned}$$

Lo cual es *imposible*, puesto que $|\cos\varphi| \leq 1$.

La explicación de este comportamiento la podemos ver en la Fig. 6.10. Para valores de x cerca de 0° ó 360°, la función $g(x)$ es asintótica hacia $+\infty$ y $-\infty$, respectivamente.

Además, tenemos otra inestabilidad cuando sucede lo siguiente: con los primeros valores con ruido se estima la localizacilon de la línea común (esto es, el valor del ángulo ρ). Después se encuentra el lugar exacto de la línea común (el valor del ángulo $\mathring{\rho}$). Entonces, se tiene el problema adicional cuando la diferencia entre las θ's es pequeño ($< 10°$) o grande ($> 350°$) y el lugar exacto de la línea común cumple $\mathring{\rho} > \rho$ (para $\theta < 10°$) ó $\mathring{\rho} < \rho$ (para $\theta > 350°$). Para visualizar este compartamiento pensemos en el mismo ejemplo anterior pero en el que suponemos que el lugar exacto de la línea común está en $\mathring{\rho} = 0.500°$, la estimación de φ resulta:

$$\begin{aligned}\cos\varphi_{12} &= (\tan 0.500°)g(2.775°)\\ &= (0.008727)(41.286183)\\ &= 0.3603\end{aligned}$$

Y $\varphi_{12} = 68.881$. Y φ existe pero el valor esta muy alejado del correcto.

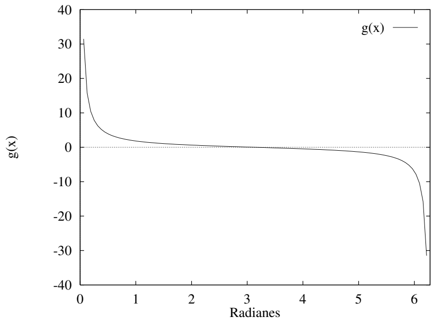


Figura 6.10: Gráfica de la función $g(x)$

Corrección al Proceso

Para corregir las inestabilidades en el proceso descrito en la sección 6.2.1, se modifica el paso 5 de la siguiente forma:

Estimar los valores de φ para cada par j, k y $\mathring{\rho}_{jk}$

$$\varphi_{jk} = \varphi(\mathring{\rho}_{jk}, \theta_j, \theta_k)$$

Si para algún par, φ no puede calcularse, hacer:

$\varphi_{jk} = $ INDEFINIDA

esto es, poner una *marca* en el lugar correspondiente.

Si para algún nuevo valor de φ_{jk} sucede que

$$\left|\mathring{\varphi} - \varphi_{jk}\right| > \Phi$$

hacer:

$\varphi_{jk} = $ INDEFINIDA

El paso 6 debe modificarse para hacer las sumatorias de las φ's [en la Ec. (6.14)] sólo sobre los valores que estén *definidos*.

El paso 7 se modifica al final, como sigue:

> Suponiendo que tenemos un grupo estimado de $\varphi,\ \theta_1,\ \theta_2,\ldots,\theta_n$. Para una nueva imagen $k = n+1$, recalcular ρ (como en el paso 1). Si para algún j, sucede que:
>
> $$\left|\rho_{jk} - \mathring{\rho}_{jk}\right| \geq 64\delta$$
>
> entonces:
>
> **PARAR** el proceso.
>
> de otro modo calcular M' (sobre j) de manera semejante a la Ec. (6.12) del paso 4. Si se cumple que M' es mayor que M, entonces:
>
> **PARAR** el proceso.
>
> de otro modo calcular
>
> $$\overline{\theta} = \frac{1}{n}\sum_{j=1}^{n}\theta_{n+1}(\mathring{\varphi}, \mathring{\rho}_{jk}, \theta_j)$$
>
> Calcular los nuevos valores de φ_{jk}
> Si para algún par de imágenes, no se puede calcular φ_{jk} hacer:
>
> $\varphi_{jk} =$ INDEFINIDA
>
> Si para algún nuevo valor de φ_{jk} sucede que
>
> $\left|\mathring{\varphi} - \varphi_{jk}\right| > \Phi$
>
> hacer:
>
> $\varphi_{jk} =$ INDEFINIDA
>
> y actualizar el valor de $\mathring{\varphi}$
>
> $$\mathring{\varphi}_{n+1} = \frac{m_n\mathring{\varphi}_n + \sum_{j=1}^{n}\varphi_{jk}}{m_n + n_{def}}, \qquad \text{para } n > 3 \tag{6.35}$$
>
> donde
>
> $$m_n = \text{al número de valores definidos en n}$$
>
> y la sumatoria se hace sobre los valores definidos de φ_{jk}, el número de estos es n_{def}. Para el siguiente $n+1$, $m_{n+1} = m_n + n_{def}$.

El valor experimental para Φ es de 6 (seis) grados.

Sobre el paso 7, en la consideración de PARO $M' > M$, se modifica a lo siguiente:

> El cambio es simple: si la condición $M' > M$ es violada, NO parar todo el proceso, sino mandar la imagen que produce esta violación al final, cogiéndola ahora como la más atípica, y continuar el proceso con la siguiente imagen en el orden de tipicalidad. Al final esa misma imagen que produjo la violación de la condición se tratará de agrupar. Este cambio permite agrupar todas las imágenes sin violar la condición $M' > M$. Claro, que si todas las imágenes que siguen, en el orden de tipicalidad también violaran la condición, el proceso terminará cuando se haya "dado la vuelta" en el orden de tipicalidad (ver Fig. 6.11).

6.2.4 Diagrama de Flujo del Proceso

En el diagrama de la Fig. 6.11 podemos ver esquemáticamente los pasos que se siguen en el proceso. Dos comentarios más, aparte de lo dicho en las secciones anteriores, son necesarios para entender el diagrama:

Primero, los ángulos son discretizados a *unidades delta*, donde cada unidad tiene el valor de $\pi/2048$. Esto explica la segunda condición de paro: $|\rho_{jk} - \mathring{\rho}_{jk}| \geq 64$. El 64 esta expresado en unidades delta y esta relacionado con la amplitud de búsqueda de la línea común del paso del paso 2 de la sección 6.2.1 (pág. 83).

Segundo, más relacionado con la realización del proceso en algún lenguaje de programación: la estructura de datos idónea para el almacenamiento de las imágenes es una *cola*. Así, el paso 7 dibujado en el diagrama se refiere a ello.

6.2.5 Experimento y Resultados

Como prueba para el proceso se realizó el siguiente experimento:

Condiciones: Se usaron 65 imágenes de proyección de un volumen sintético (un fantasma). Una visualización de superficie del volumen puede verse en la Fig. 6.12. Las imágenes fueron retroproyectadas a partir del volumen con $\varphi = 55°$ y $\theta = i \times 360°/65$, $0 \leq i < 65$.

Condiciones de ruido:

- $\pm 3°$ y $\pm 1°$ en el valor de φ,
- Se sumó a los valores de las θ's un un valor aleatorio (ruido) con distribución gaussiana de media cero y desviación estándar de $1°$.
- Se hicieron cinco grupos de imágenes, cada uno con 65 imágenes. A los pixels de las imágenes, para cada grupo, se les adicionó un ruido gausiano de media cero y con un valor en la desviación estandar acorde para obtener un *nivel fraccional de ruido (nfr)* [41] de: 0.00, 0.05, 0.10, 0.20 y 0.40, respectivamente para cada grupo.

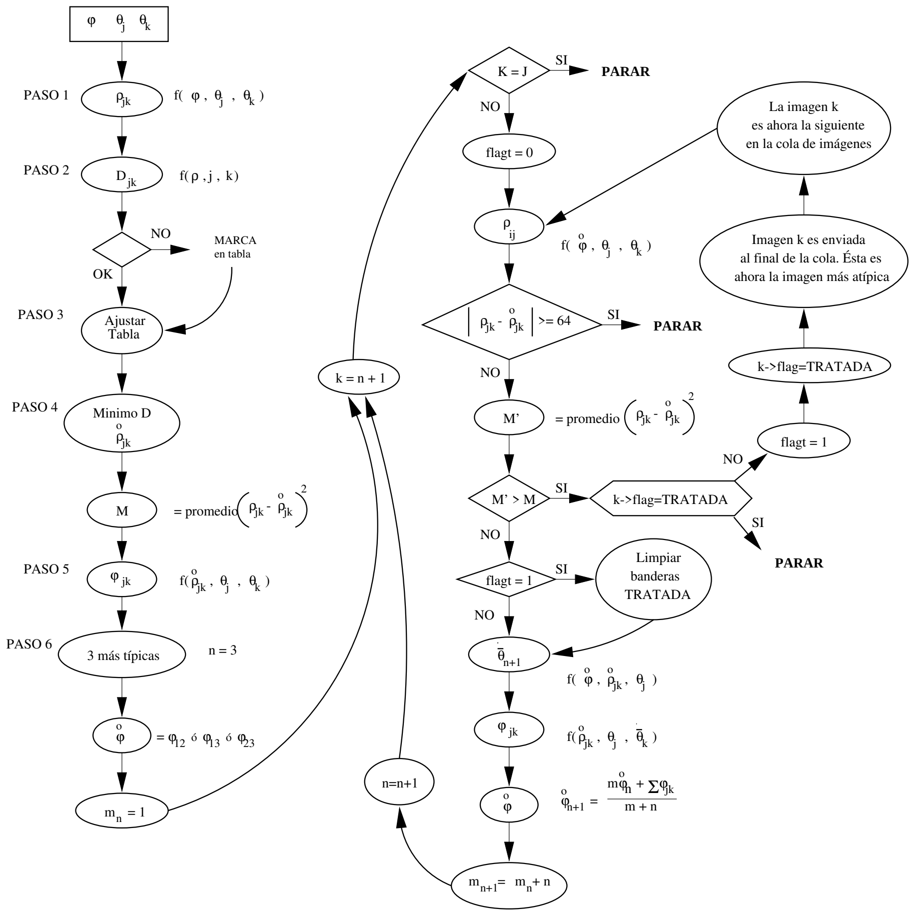


Figura 6.11: Diagrama de flujo del proceso

Se realizaron entonces 20 ejecuciones del proceso: hay 4 condiciones de ruido en φ y 5 condiciones de ruido correspondientes a cada nfr, lo que hacen las 20 ejecuciones. Al finalizar cada corrida se midió:

1. El error en φ como:

$$error_{\varphi} = \mathring{\varphi}_{final} - 626ud$$

   donde 626 unidades delta (ud) es igual a 55°

2. Dos medidas del error final en las θ's:

$$error_{1-\theta} = \frac{\sum \left|\overline{\theta}_i - \theta_i\right|}{K}, \qquad error_{2-\theta} = \sqrt{\frac{\sum \left(\overline{\theta}_i - \theta_i\right)^2}{K}}$$

   donde:
   $\overline{\theta}_i$ es el valor *final* del ángulo de rotación para la imagen i.

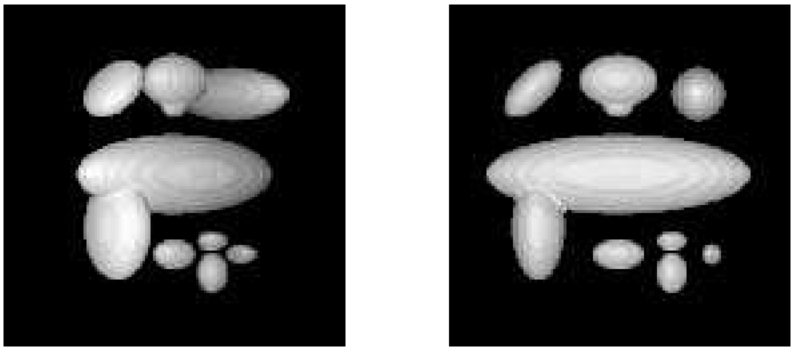


Figura 6.12: Visualización de superficie del volumen usado en los experimentos

θ_i es el valor *correcto* del ángulo de rotación para la imagen i.
K es el número de imágenes.

3. El número final de imágenes.

El nivel fraccional de ruido adicionado a las imágenes se puede entender como sigue; supongamos que $f(i,j)$ es una imagen de K imágenes en el grupo con nfr = 0.00; se genera otro grupo de imágenes a un nuevo nfr como:

$$\sigma_n = \mathrm{nfr}_{nuevo} \frac{\sum_K [\max f(x,y) - \min f(x,y)]}{K} \tag{6.36}$$

$$g(i,j) = f(i,j) + N(0,\sigma_n) \tag{6.37}$$

donde:
$f(i,j)$ es la reproyección del volumen.
$g(i,j)$ es la nueva imagen generada.
$N(0,\sigma_n)$ es el ruido Gaussiano de media 0 y desviación estándar σ. K es el número total de imágenes.

Resultados

En las Tablas 6.2, 6.3, 6.4 y 6.5 aparecen los resultados de los experimentos. De las condiciones iniciales se conoce lo siguiente: valor correcto de $\varphi = 55° = 626$ ud; error inicial $\mathrm{error}_{1-\theta} = 70.29$ ud; error inicial $\mathrm{error}_{2-\theta} = 504.80$ ud.

En la Tabla 6.5 se muestra la relación entre imágenes aceptadas y rechazadas al final del proceso según el nivel fraccional de ruido y el error inicial en φ. Para el experimento con nfr = 0.40, 55 imágenes fueron eliminadas (por el paso 2 y 3) y las estimaciones de φ con las 10 restantes resultaron indefinidas (paso 5), resultando que todas las imágenes fueron rechazadas.

En las Figuras 6.13, 6.14 y 6.15 se visualizan los resultados obtenidos para el caso de $\varphi = 58°$.

Estos resultados nos indican que el nivel fraccional de ruido máximo en que nuestro proceso es operable es de 0.20. El nivel fraccional de ruido de 0.20 corresponde, según Farrow y Ottensmeyer [41] a las imágenes digitalizadas de micrografías tomadas en *campo obscuro* [60, 8] de macromoléculas individuales sin teñir. Las micrografías tomadas con el modo campo obscuro del Microscopio Electrónico de Transmisión (MET) presentan un mejor contraste, y menor ruido, que las tomadas en el modo *campo brillante* del MET, sin embargo el modo campo obscuro produce imágenes a menor resolución.

Támbien este proceso desarrollado podria aplicarse a imágenes procedentes del Microscopio de Transmisión con Barrido (STEM) [24] o a imágenes procedentes del Microscopio Electrónico con Filtración de Energía [85]. Ambos microscopíos nos dan imágenes con una razón señal-a-ruido más grande que las imágenes obtenidas del MET en modo normal (campo brillante).

nfr	$\varphi = 58°$ (660 ud)	$\varphi = 56°$ (637 ud)	$\varphi = 54°$ (614 ud)	$\varphi = 52°$ (592 ud)
0.00	0	-1	0	0
0.05	-1	-2	-1	-1
0.10	0	-2	-1	-1
0.20	11	-8	-11	-6
0.40	—	—	—	—

Tabla 6.2: Errores en φ, los valores se expresan en unidades delta

nfr	$\varphi = 58°$ $error_{1-\theta}$	$\varphi = 58°$ $error_{2-\theta}$	$\varphi = 56°$ $error_{1-\theta}$	$\varphi = 56°$ $error_{2-\theta}$
0.00	4.206349	4.572173	4.087719	4.232290
0.05	4.147541	4.809485	3.661017	4.588120
0.10	6.226415	7.435255	4.886364	6.264982
0.20	15.333333	19.058805	10.533334	14.454527
0.40	—	—	—	—

Tabla 6.3: Errores finales en θ's, todos valores se expresan unidades delta

nfr	$\varphi = 54°$ $error_{1-\theta}$	$\varphi = 54°$ $error_{2-\theta}$	$\varphi = 52°$ $error_{1-\theta}$	$\varphi = 52°$ $error_{2-\theta}$
0.00	1.465517	1.898275	1.587302	2.468854
0.05	5.034483	6.011483	2.870968	3.672039
0.10	5.735849	6.998652	5.830189	7.204558
0.20	11.923077	14.984608	8.625000	11.081140
0.40	—	—	—	—

Tabla 6.4: Errores finales en θ's, todos valores se expresan en unidades delta

nfr	$\varphi = 58°$	$\varphi = 56°$	$\varphi = 54°$	$\varphi = 52°$
0.00	63/2	57/8	58/7	63/2
0.05	61/4	59/6	58/7	62/3
0.10	53/12	44/21	53/12	53/12
0.20	21/44	15/50	26/39	24/41
0.40	0/65	0/65	0/65	0/65

Tabla 6.5: Número final de imágenes. Los valores representan el número de aceptadas/número de rechazadas

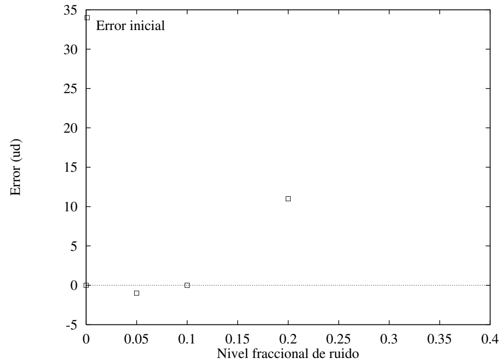
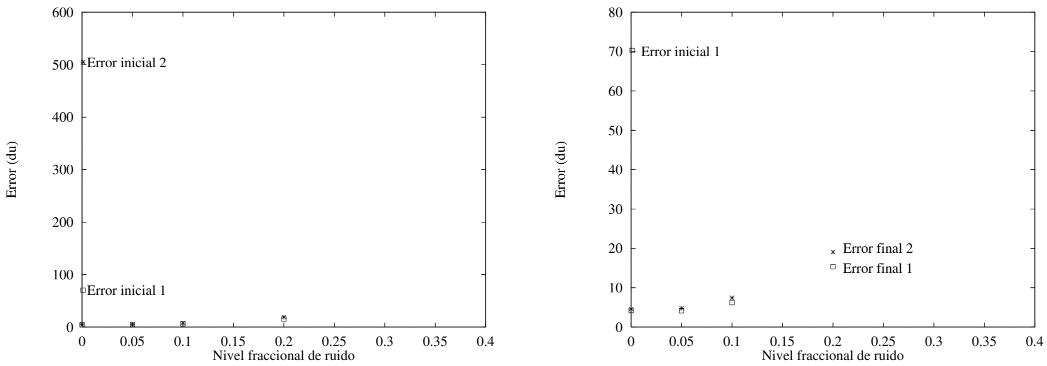


Figura 6.13: Error en φ vs. nfr.



(a) (b)

Figura 6.14: Error en θ's vs. nfr. (b) es la parte baja, expandida, de (a)

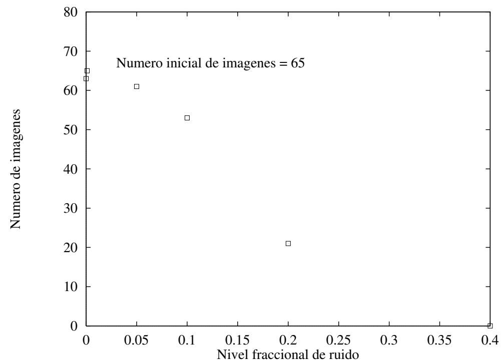


Figura 6.15: Número final de imágenes en el proceso

6.3 Estimación del Ángulo de Inclinación para Partículas con Simetría Cilíndrica

En esta sección se tratará el problema de la estimación del ángulo de inclinación de imágenes de proyección, obtenidas siguiendo una geometría de inclinación cónica, particularizado para aquellos casos en que el espécimen en estudio presente una simetría cilíndrica.

En la sección anterior se trató el tema del refinamiento de los dos ángulos involucrados en la geometría de inclinación cónica: un ángulo de inclinación para todo un grupo de partículas, y los ángulos de rotación de todas las partículas. Posteriormente a la finalización del método de la sección anterior, se midió el error que resulta de la estimación del ángulo rotacional, θ, por el procedimiento habitual del método de recolección de datos de la geometría cónica aleatoria [89], que usa un algoritmo de alineamiento con una referencia dinámica [2] [83] (basado en funciones de correlación cruzada). Resultó que el proceso de alineamiento usando una referencia dinámica es un procedimiento robusto, desde el punto de vista que da unos buenos valores del los ángulos de rotación para condiciones de fuerte ruido (aún en imágenes con un nfr de 0.80) debido a que usa funciones de correlación cruzada.

El método de la sección anterior, además, da un valor para el ángulo de inclinación para todo el grupo de imágenes involucrado. Así que se pensó en realizar un nuevo procedimiento que refinara el ángulo de inclinación *para cada imagen*, lo que nos acercaría más a la situación real donde cada imagen puede tener un ángulo de inclinación distinto debido a que el soporten donde yacen no es completamente plano.

Se particularizó el problema para imágenes con simetría cilíndrica debido a que presentan varias líneas comunes, y debido así mismo por ello, se puede luchar contra el ruido presente en la imágenes. En esta sección se tratará, por lo tanto, en crear una aproximación que nos permita mejorar la estimación inicial del ángulo de inclinación para cada imagen inclinada de una partícula que presente simetría cilíndrica.

En la Fig. 6.16 se observa los pasos involucrados en el nuevo proceso que proponemos en esta memoria para la estimación del ángulo de inclinación. Como entrada al proceso tenemos una imagen bidimensional que es una imagen de proyección de un volumen con simetría cilíndrica. En general es imagen muestra la proyección del espécimen en la zona central, siendo la periferia fondo. El primer paso es cortar la imagen inicial a un tamaño más pequeño, de manera que solo tenemos la porción central donde se encuentra la señal. En seguida es expandida para obtener una nueva imagen de tamaño impar. En esta última imagen se calcula la discrepancia entre líneas comunes. La situación de estas líneas comunes se tratará en la sección 6.3.1. La discrepancia en la(s) línea(s) comun(es) viene dada por la función $d(\varphi)$; el mínimo de esta función nos dará el valor estimado del ángulo de inclinación. Este

[2] Originalmente el método fue llamado *alineamiento con una referencia libre.* El término *alineamiento con una referencia dinámica* parece más apropiado. M. Radermacher, comunicación personal 1996

valor mínimo es calculado ajustando la función $d(\varphi)$ a un modelo de curva conocido.

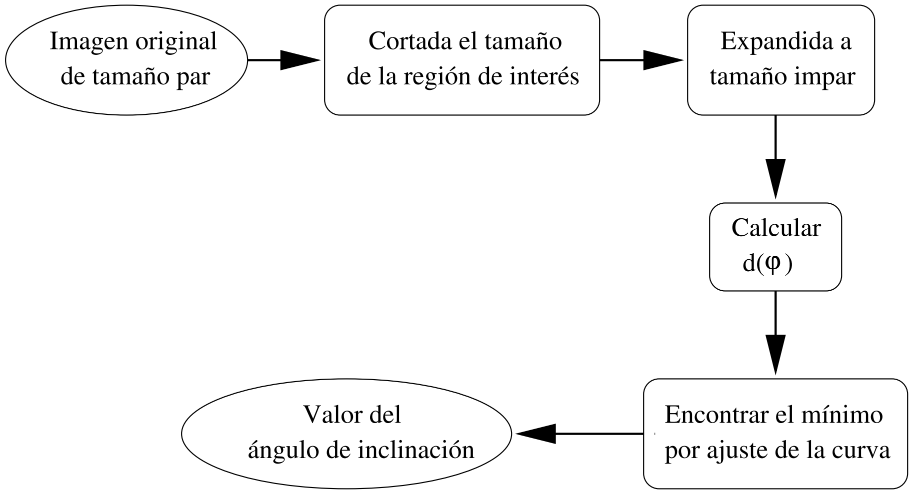


Figura 6.16: Pasos involucrados en el proceso de la estimación del ángulo de inclinación para especímenes con simetría cilíndrica

La expansión de la imagen a un tamaño impar es necesario porque se han usado algunas subrutinas del paquete SNARK93 [11]. Para calcular las proyecciones de una imagen bidimensional, SNARK93 sólo trabajo con imágenes digitalizadas de tamaño $n \times n$, donde n es un entero impar, de esta manera el origen de la imagen yace sobre el centro del pixel central. Esta parte será tratada en la sección 6.3.3.

Se han usado las subrutinas de SNARK93: POSIT, WRAY y PSEUDO; todas ellas para el caso de proyección paralela [11]. Estas subrutinas, originalmente escritas en FORTRAN, fueron traducidas al lenguaje C.

6.3.1 Análisis de Partículas con Simetrías

En lo que sigue usaremos el sistema de coordenadas cilíndricas (r, α, z).

Se trabajarán con objetos f para los cuales existe una β tal que, para cualquier (r, α, z) se cumple:

$$f(r, \alpha + 2\beta, z) = f(r, \alpha, z) \tag{6.38}$$

Consideremos ahora la situación de la Fig. 6.17. Sobre la figura, C, E y E' están en el plano que pasa a través del origen y que es ortogonal al eje z (en el ejemplo, en un plano horizontal). Tenemos que $\angle$COE $=$ $\angle$COE' $= \beta$ y $\angle$EOE' $= 2\beta$. Así mismo el segmento de línea OC' (la longitud de $\overline{\text{OC}}$ $=$ c) es ortogonal a la línea base del sistema de coordenadas cilíndrico (el cual se escoge de manera que que sea el mismo sobre el eje en el cual la partícula es inclinada) y $\overline{\text{OE}} = \overline{\text{OE'}}$. Los segmentos de línea CB, ED y E'D' son verticales (en el ejemplo, paralelos al eje z) y son todos de la misma longitud. Sigue, por lo tanto, que la línea DBD' es horizontal y

$$\angle\text{OCB} = \angle\text{OCE} = \angle\text{OBD} = \angle\text{OBD'} = \frac{\pi}{2} \tag{6.39}$$

así mismo definimos $\rho = \angle\mathrm{BOD} = \angle\mathrm{BOD'}$, $\varphi = \angle\mathrm{BOC}$, $\mathrm{b} = \overline{\mathrm{OB}}$ y $\mathrm{a} \ = \ \overline{\mathrm{BD}} = \overline{\mathrm{CE}}$.

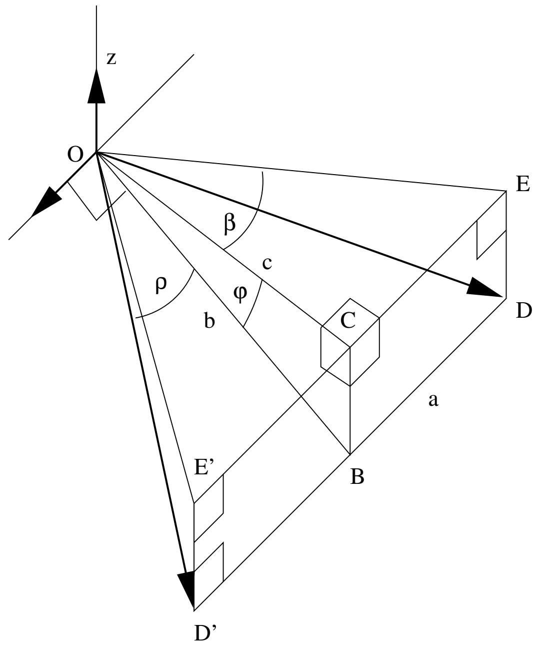


Figura 6.17: Esquema de las relaciones geométricas para el estudio de especímenes con simetría rotacional alrededor del eje z. Detalles en el texto.

De estas relaciones geométricas se deduce que:

$$\cos\varphi = \frac{\mathrm{c}}{\mathrm{b}}, \qquad \tan\beta = \frac{\mathrm{a}}{\mathrm{c}}, \qquad \tan\rho = \frac{\mathrm{a}}{\mathrm{b}} \tag{6.40}$$

Poniendo juntos los valores de la Ec. 6.40 obtenemos:

$$\cos\varphi = \frac{\tan\rho}{\tan\beta} \tag{6.41}$$

Considerando ahora los dos planos que son ortogonales a los vectores marcados con D y D', en la Fig. 6.17, y que contienen los puntos finales de estos vectores. Dada la suposición básica de la Ec. 6.38, tenemos que los valores de la función f que concuerdan con esos puntos en esos dos planos, son iguales el uno al otro y así las integrales planas de la función f sobre los dos planos son así mismo iguales la una a la otra. Ahora, si inclinamos todo este arreglo de la Fig. 6.17 por el ángulo φ alrededor del eje de inclinación (el cual es el mismo que la línea base del sistema de coordenadas cilíndrico) de manera que la línea OB (y consecuentemente las líneas OD y OD') queden horizontales. Como resultado de esta inclinación, los dos planos que están al final y que son ortogonales a los vectores D y D', respectivamente, se vuelven paralelos al eje z (en otras palabras, ambos vectores se vuelven verticales).

Por lo tanto, una vez que se han realizado las operaciones descritas párrafo anterior, los puntos O, B, D y D' yacen en el mismo plano ortogonal al eje z. Esto significa, en particular, que en una proyección vertical de este arreglo inclinado, el ángulo ρ no se encuentra escorzado. Mas aún, si se toma tal proyección vertical ortogonal a la línea OD, tenemos que ésta es igual a la integral de línea de la proyección ortogonal a la línea OD' tomadas a la misma distancia de O. Por este procedimiento, ρ puede ser estimado directamente de la proyección vertical de la partícula inclinada, calculando las integrales de línea de la proyección para pares de líneas divergentes de O y tomándolas al mismo ángulo sobre ambos lados de $\overline{OB}$. (El valor aproximado de ρ esta dado por la ecuación (6.41) y por el valor inicial que se tenga para el ángulo de inclinación φ; la búsqueda del valor exacto estará en la vecindad de este valor inicial aproximado y puede ser necesario, o nó, tener conocimiento a priori del valor de φ).

Supongamos ahora que tenemos una partícula con simetría cilíndrica s. La Ec. (6.38) se mantiene para $s-1$ valores angulares de β, donde:

$$\beta_i = \frac{\pi}{s} \cdot i, \qquad i = 1, \ldots, s-1 \tag{6.42}$$

Para encontrar los valores de ρ se necesitan pares de proyecciones (a ρ y $-\rho$, ver el párrafo previo) y, dado que φ es fija, el comportamiento de ρ es igual al comportamiento periódico de β, esto es, una función tangente, que es una función par en 0 ó π. Por lo tanto, tendremos $(s-1)/2$ valores útiles de β en la amplitud de 0 a $\pi/2$, y en el caso de que s sea un nḿero par, no se puede usar el valor de $\beta = \pi/2$ (las dos proyecciones serían una misma), así que para cualquier simetría s definimos:

$$n = \mathrm{int}\left(\frac{s-1}{2}\right) \tag{6.43}$$

donde n es el número de valores distintos de β en los que existen líneas comunes. Estos valores son:

$$\beta_i = \frac{\pi}{s} \cdot i, \qquad i = 1, \ldots, n \tag{6.44}$$

Nótese que la ecuación (6.43) será cero para $s = 2$.

Dado s, el valor de la simetría, calculamos entonces los valores de n y β_i de las Ec. (6.43) y (6.44). De esta forma podemos resolver la Ec. (6.41), reescrita como:

$$\rho_i(\varphi) = \arctan(\cos\varphi \cdot \tan\beta_i), \qquad i = 1, \ldots, n \tag{6.45}$$

Para cada valor de ρ_i definimos:

$$\mathrm{D}(\rho) = \sum_{r \text{ rayos}} \left[\mathrm{p}\left(\rho - \frac{\pi}{2}, r\right) - \mathrm{p}\left(-\rho - \frac{\pi}{2}, r\right)\right]^2 \tag{6.46}$$

como la distancia cuadrática entre las proyecciones a $\rho-\pi/2$ y $-\rho-\pi/2$. Finalmente se calcula la suma de las distancias cuadráticas, $d(\varphi)$, sobre pares de lineas comunes:

$$d(\varphi) = \sum_{i=1}^{n} \mathrm{D}(\rho_i) \tag{6.47}$$

Para la estimación del ángulo de inclinación φ, comenzamos con la primera aproximación que se tenga de este y se busca el valor exacto del ángulo en la vecindad de este valor inicial. El mínimo de la ecuación (6.47) nos da el valor correcto del ángulo de inclinación. La sumatoria en la Ec. (6.47) realiza una media que nos debe mejorar la relación señal-a-ruido, tanto mejor cuanto más grande sea el orden de simetría en nuestro espécimen.

6.3.2 La Región de Interés

Para calcular las integrales de línea sobre la imagen inclinada se va a considerar sólo la región donde se encuentra el objeto. Esto es debido a que en las imágenes típicas reales se tiene que el tamaño del objeto es menor que el tamaño total de la imagen. Por lo tanto, si decimos que n es el tamaño total de la imagen (vamos a suponer imágenes cuadradas) y r es el tamaño de *la región de interés* (r debe ser menor que n, y ambos deben de ser números enteros pares). Se crea una nueva imagen de tamaño r y se copian los valores de la imagen original.

Supongamos que $f(i,j)$ es la imagen original y $g(i,j)$ es la nueva imagen generada. Así, este paso del procesamiento puede ser expresado como:

$$g(i,j) = f(i+h, j+h), \qquad 0 \le i < r,\ 0 \le j < r \tag{6.48}$$

donde:

$$h = \frac{n-r}{2}$$

6.3.3 Expansión de la Imagen a un Tamaño Impar

)

Sea $g(I,J)$ la imagen original, la misma de la Ec. (6.48), de tamaño $n \times n$, n es un numero entero par. Es necesario expandir esta imagen a otra $g_2(i,j)$, de tamaño $m \times m$, donde m es un número entero impar. Esto es debido a que han usado algunas subrutinas del paquete SNARK93 [11] para calcular las proyecciones de las imágenes [ver la Ec. (6.46)] . SNARK93 sólo trabaja con imágenes cuadradas de tamaño impar.

En la Fig. 6.18 se ve la imagen original, cuyos pixels les llamaremos *pixels grandes*, y sobre este imagen está dibujada la nueva imagen, cuyos pixels les llamaremos *pixels pequeños*. La relación entre tamaños es:

$$m = 2(n-2) - 1 \tag{6.49}$$

Para generar la nueva imagen, g_2, se usará una máscara circular de radio $\sqrt{1.25}$. Tenemos cuatro casos posibles en el cálculo de los *pixels pequeños* y que están presentados en la Fig. 6.19. En las siguientes subsecciones se tratar'a la derivación de los coeficientes w_0, w_1 y w_2 que aparecen en la misma figura.

Antes es necesario conocer como calcular el área de las zonas circulares. De la ecuación del circulo $y = \sqrt{r^2 - x^2}$ se llega a la siguiente ecuación:

$$\int \sqrt{r^2 - x^2}dx = \frac{x\sqrt{r^2 - x^2}}{2} + \frac{r^2}{2}\arcsin\frac{x}{r} \tag{6.50}$$

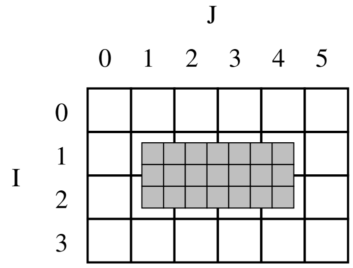


Figura 6.18: Los *pixels grandes*, de la imagen original, y los *pixels paqueños* de la nueva imagen. I y J son índices a los pixels grandes.

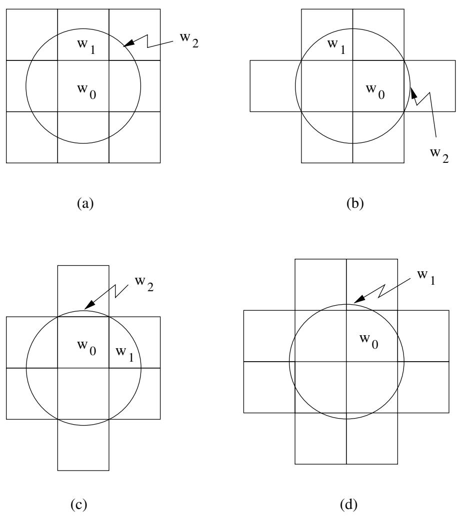


Figura 6.19: Los cuatro casos para el cálculo de los *pixels pequeños*

Pasamos a considerar cada uno de los cuatro casos que aparecen en el cálculo de los pixels pequeños.

Caso (a)

Para el caso (a) de la Fig. 6.19 tenemos el esquema de la Fig. 6.20.

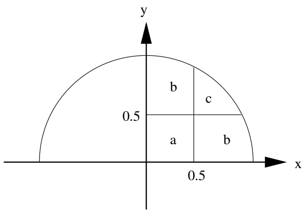


Figura 6.20: Esquema para calcular las coeficientes en el caso (a)

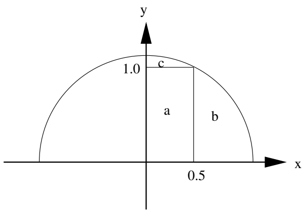


Figura 6.21: Esquema para calcular las coeficientes en el caso (b) y (c)

Viendo la Fig. 6.20, podemos deducir que:

$$\int_0^{0.5} \sqrt{r^2 - y^2}\, dx = a + b \tag{6.51}$$

$$a = (0.5)^2 \tag{6.52}$$

$$a + 2b + c = \frac{1}{4}\pi r^2 \tag{6.53}$$

y los valores de los coeficientes corresponden a:

$$w_0 = 4a = 1, \qquad w_1 = 2b, \quad \text{y} \quad w_2 = c \tag{6.54}$$

Resolviendo las ecuaciones (6.51) y (6.52) y tomando en cuenta que $r^2 = 1.25$, obtenemos los valores para los coeficientes:

$w_0 = 1.0$
$w_1 = 0.579559512$
$w_2 = 0.152188193$

Caso (b) y (c)

Podemos ver en la Fig. 6.19 que para los casos de las máscaras (b) y (c) tenemos los mismo valores para los coeficientes, así que podemos dibujar un diagrama como el de la Fig. 6.21. Observando esta misma figura podemos deducir:

$$\int_0^{0.5} \sqrt{r^2 - y^2}\, dx = a + c \tag{6.55}$$

$$a = 0.5 \tag{6.56}$$

$$a + 2b + c = \frac{1}{4}\pi r^2 \tag{6.57}$$

Los valores de los coeficientes respectivos son:

$$w_0 = 2a = 1, \qquad w_1 = 2b, \quad \text{y} \quad w_2 = 2c \tag{6.58}$$

Resolviendo las ecuaciones (6.55) y (6.57) con la Ec. (6.56) y $r^2 = 1.25$ obtenemos los valores:

$w_0 = 1.0$
$w_1 = 0.441967948$
$w_2 = 0.079559512$

Caso (d)

Para este último caso procedemos como en los dos anteriores. Observando la Fig. 6.22 deducimos:

$$\int_0^1 \sqrt{r^2 - y^2}\, dx = a + b \tag{6.59}$$

$$a + 2b = \frac{1}{4}\pi r^2 \tag{6.60}$$

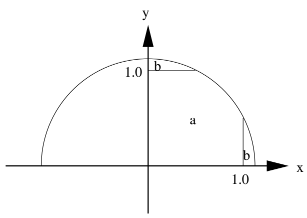


Figura 6.22: Esquema para calcular las coeficientes en el caso (d)

$$w_0 = a \quad \text{y} \quad w_1 = b \tag{6.61}$$

Resolviendo la ecuación (6.59) y sustrayendo el resultado a la Ec. (6.60), obtenemos el valor de b. Y los valores de los coeficientes resultantes quedan como:

$$w_0 = 0.902188193$$
$$w_1 = 0.03977956$$

La asignación de los pixels

¿Cómo escoger entre cada uno de las máscaras para cada caso? Si I, J son los índices para cada pixels sobre la imagen original e i, j son los índices para designar cada pixel pequeño, sobre la imagen expandida; entonces se escoge cada máscara como sigue:

```
For all i, j do                         /* 0 ≤ i < n, 0 ≤ j < n */
    I = int(i/2) + 1;
    J = int(j/2) + 1;
    if I es par then
        if J es par then
            use la máscara (a);
        else
            use la máscara (b);         /* J es impar */
        endif
    else
        if J es par then
            use la máscara (c);
        else
            use la máscara (d);         /* J es impar */
        endif
```

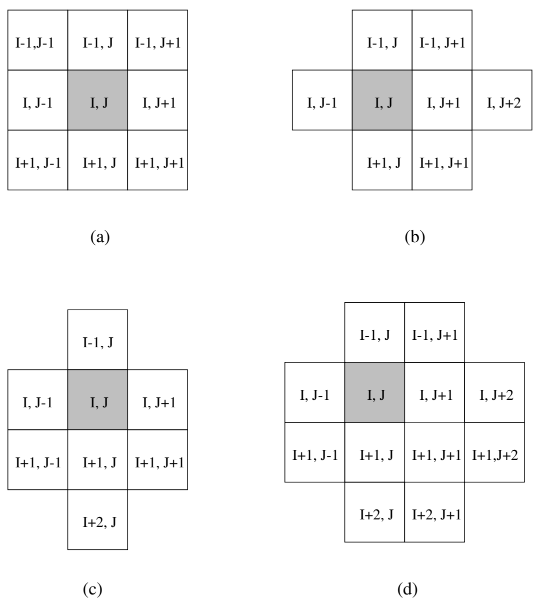


Figura 6.23: Índices de los pixels sobre los *pixels grandes*. El pixel sombreado es el pixel central.

endif
endo

Los I, J calculados arriba son las coordenadas de los pixels centrales como puede verse en la Fig. 6.23.

Finalmente, como un ejemplo, la Ec. (6.62) es la que define a un pixel pequeño en el caso de usar la máscara (a).

$$\begin{aligned} g(i,j) = \frac{1}{1.25\pi}\Big\{ & f(I,J) + \\ & + w_1 \left[f(I,J-1) + f(I,J+1) + f(I-1,J) + f(I+1,J) \right] + \\ & \quad + w_2 \big[f(I-1,J-1) + f(I-1,J+1) + \\ & \qquad + f(I+1,J-1) + f(I+1,J+1) \big] \Big\} \end{aligned} \tag{6.62}$$

de la misma manera podemos obtener las demás ecuaciones para las otras tres máscaras.

6.3.4 Modelo del Ruido

Consideremos la situación expuesta en la Fig. 6.24 que representa una imagen de proyección. Vamos a suponer que la toda la señal cae dentro del circulo, el cual está indicado aproximadamente por la unión de pixels; el diámetro del círculo es justo el tamaño de la *región de interés* (ver sección 6.3.2). Se va considerar entonces sólo las pixels que yacen dentro del círculo.

Si el ángulo ρ determina la localización de las líneas comunes $P1$ y $P2$, entonces las integrales de línea a lo largo de las correspondientes líneas ($i1$ e $i2$) perpendiculares a las líneas comunes deberían ser las mismas. Estas integrales de línea pueden ser expresadas como:

$$\sum_{j=1}^{J} a_{i,j} x_j \tag{6.63}$$

donde $a_{i,j}$ corresponde a la longitud de la intersección del i-ésimo rayo con el j-ésimo pixel, y x_j corresponde al valor del pixel. Nótese que para el pixel $j1$ intersectado por la línea $i1$ en la proyección a $P1$ de manera que el largo de la intersección es $a_{i1,j1}$, hay un pixel $j2$ intersectado por la línea $i2$ en la proyección a $P2$ de manera que el largo de la intersección es $a_{i2,j2} = a_{i1,j1}$. Ahora bien, si x_j de la Ec. 6.63 tiene un componente de ruido Gaussiano cuya media es 0 y cuya varianza es σ^2, entonces la variable aleatoria:

$$\sum_{j1=1}^{J} a_{i1,j1} x_{j1} - \sum_{j2=1}^{J} a_{i2,j2} x_{j2} \tag{6.64}$$

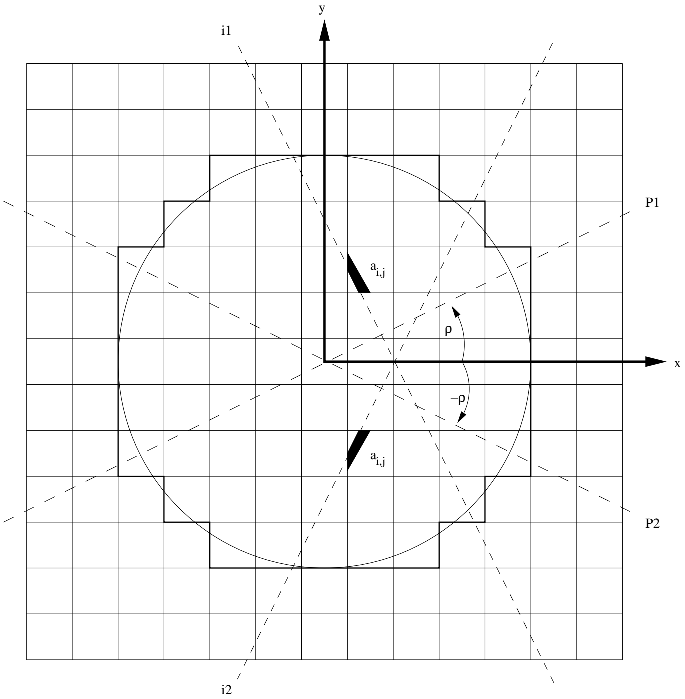


Figura 6.24: Una imagen de proyección

que es la diferencia entre las sumas de rayos correspondientes, es también una variable aleatoria de media 0 y varianza:

$$2\sigma^2 \sum_{j\in K} (a_{i,j})^2 \tag{6.65}$$

donde K es el grupo de índices de todos los pixels que caen dentro de la región de interés y no están sobre el eje de las x. Hay un punto sutil aquí, relevante a la suposición de que la imagen tiene un número impar de pixels en el tamaño. Como resultado de esto, para las líneas $i1$ e $i2$, un pixel $j1$ el cual yace sobre el eje x, es el mismo que algún pixel $j2$ el cual yace sobre el eje x, y así sucesivamente. Esto es de forma independiente de cualquier contribución del ruido, ya que los términos $a_{i1,j1}x_{j1}$ y $a_{i2,j2}x_{j2}$ se cancelan mutuamente en la Ec. (6.64) para los pixels que caen sobre el eje x.

Es fácil comprobar que, para cualquier variable aleatoria $\mathbf{x}$ de media cero, la media de $\mathbf{x}^2$ es la varianza de $\mathbf{x}$ [81]. De este hecho sigue inmediatamente que, para las correspondientes líneas de integración $i1$ e $i2$,

$$\frac{\left(\sum_{j1=1}^{J} a_{i1,j1}x_{j1} - \sum_{j2=1}^{J} a_{i2,j2}x_{j2}\right)^2}{2\sum_{j\in K}\left(a_{i,j}\right)^2} \tag{6.66}$$

es una variable aleatoria de media σ^2.

Para cualquier ángulo ρ, sea I el grupo de aquellas líneas i para las cuales $2\sigma^2\sum_{j\in K}\left(a_{i,j}\right)^2 > \delta^2$ (donde δ es el largo del lado de un pixel) y se define

$$D(\rho) = \sum_{i\in I}\frac{\left(\sum_{j1=1}^{J} a_{i1,j1}x_{j1} - \sum_{j2=1}^{J} a_{i2,j2}x_{j2}\right)^2}{2\sum_{j\in K}\left(a_{i,j}\right)^2} \tag{6.67}$$

Como vemos, esta ecuación es la versión extendida, tomando en cuenta el ruido, del caso previo de la ecuación (6.46) (en la pag. 101).

6.3.5 Búsqueda del Mínimo de $d(\varphi)$

Como se vió al final de la sección 6.3.1, necesitamos el encontrar el mínimo de la Ec. (6.47), esto es, el mínimo de $d(\varphi)$, que nos proporciona el valor correcto del ángulo de inclinación. Conocemos que en bajo una condición sin ruido, el mínimo es cero, y también que la función d tiene valores positivos y es asimétrica alrededor del mínimo, donde la derivada de la función d es también cero. Un ejemplo de la forma típica de la función $d(\varphi)$, calculada experimentalmente con el fantasma descrito más adelante (pag. 115), se muestra en la Fig. 6.25. Aquí tenemos el problema de que el ruido superimpuesto es tan grande que no podemos identificar la localización del mínimo en el caso sin ruido, con la localización del mínimo en el caso con ruido. En particular, ya que el valor esperado del ruido es positivo, el valor esperado de su mínimo en el caso con ruido es también positivo (más que cero, al igual que el caso sin ruido).

Para superar esta dificultad vamos a considerar la siguiente aproximación para identificar la localización del punto mínimo por ajuste a una curva. Se proponen dos algoritmos para realizar esto, uno por un ajuste a un modelo lineal y otro usando un modelo no lineal.

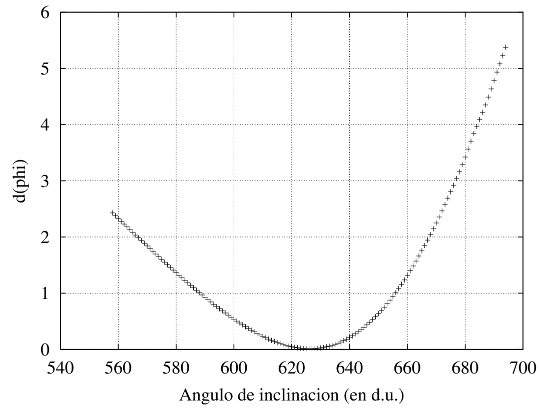


Figura 6.25: Forma típica de la función $d(\varphi)$. Los puntos fueron calculados experimentalmente usando una imagen retroproyectada a $\varphi = 55°$ de un fantasma de simetría cilíndrica de orden 8 (la especificaciones del fantasma pueden verse en la pág. 115).

Algoritmo No. 1

Aquí vamos a asumir que la forma de la curva que define la Ec. (6.47) será aproximada por un polinomio de cuarto orden:

$$\mathrm{d}(x) = \sum_{n=0}^{4} a_n x^n \tag{6.68}$$

Se escogió un polinomio de cuarto orden en base a las las características de la función $d(\varphi)$ (ver también la Fig. 6.68): todos sus valores son positivos y su forma es asimétrica alrededor del mínimo. Un polinomio de segundo orden, una parábola, serviría como primera aproximación, pero su forma no es asimétrica alrededor de su mínimo. Un polinomio de tercer grado nos daría valores positivos y negativos. El polinomio de menor grado que se ajusta a nuestro datos experimentales es el de cuarto orden, donde el coeficiente a_4 debe ser mayor que cero y el valor del coeficiente a_0 (debe ser cero en caso de no tener ruido) nos permitiría ajustar el valor esperado positivo del ruido. Para cada valor posible de φ en la amplitud sobre la cual estamos buscando, se consideran $2R + 1$ valores contiguos e incrementales valores de φ (inicialmente podría ponerse a $R = 16$). Se estiman los coeficientes de la Ec. (6.68) de manera que el valor de la función de error descrita en la Ec. (6.69) sea tan pequeño como sea posible.

$$E(\varphi) = \sum_{r=1}^{2R+1} \left[d(\varphi + r\delta) - \mathrm{d}(\varphi + r\delta)\right]^2 \tag{6.69}$$

donde δ es la unidad del valor incremental para φ. Para toda nuestra aplicación δ es igual a

$$\delta = \frac{\pi}{2048} \approx 0.088° \tag{6.70}$$

La ecuación (6.69) es justamente el error chi-cuadrado, entre los valores experimentales y nuestro modelo, el cual debe ser minimizado.

En la amplitud de nuestra búsqueda, en los puntos donde la Ec. (6.69) es minimizada, también se encuentra la raíz de la derivada de la Ec. (6.68), si el valor de su raíz, c, tiene la propiedad que

$$\varphi - \frac{R}{3}\delta \leq c \leq \varphi + \frac{R}{3}\delta \tag{6.71}$$

y la Ec. (6.69) tiene el valor más pequeño que todos los demás, c será nuestro estimador del ángulo de inclinación φ.

Ahora, supongamos que los datos experimentales (los valores de d) nos son disponibles en M puntos, los cuales denotaremos por $\varphi_0 + s, \ldots, \varphi_0 + Ms$. Supóngase también que tenemos una R fija (mucho más pequeña que M) y definimos T como el más grande entero menor que $R/3$. La realización de nuestro primer algoritmo quedaría como en el siguiente pseudocódigo.

Algoritmo No. 1 para estimar el ángulo de inclinación

Inicializar $C = \varphi_0$, Error $= \infty$;
 For $1 \leq m \leq M - 2R$ **do**
  set
   $y(r) = \mathrm{d}\,(\varphi_0 + ms + (r-1)s); \qquad 1 \leq r \leq 2R+1$

   $l = \sum_{r=R-3T}^{R-T} y(r)$

   $k = \sum_{r=R-T+1}^{R+T+1} y(r)$

   $h = \sum_{r=R+T+2}^{R+3T+2} y(r)$

  if $k \geq \min(l, h)$ **then continue with for**
  find los valores de los coeficientes en la Ec. (6.68) de manera que

   $e = \sum_{r=1}^{2R+1} [y(r) - \mathrm{y}(r)]^2$

  sea tan pequeño como sea posible.
  find la raíz real de la derivada de la eq. (6.68)
  if no es posible **then continue with for**
  if $e < E$ y
   $R - T \leq c - 1 \leq R + T$

  then set Error$=e$, $C = \varphi_0 + ms + (c-1)s$
 enddo
if $C = \varphi_0$, entonces la búsqueda a fallado y **stop**
Output C (como nuestro estimador del ángulo de inclinación)
End

Algoritmo No. 2

Para este algoritmo se asume que la forma de la curva de la función $d(\varphi)$ (Ec. 6.47), en al caso de no haber ruido, es de la forma general:

$$d(x) = a^2 \left[6x^4 + 8(b-c)x^3 + 3(b^2 - 4bc + 1)x^2 - 6(b^2+1)cx\right] + n \qquad (6.72)$$

Para entender por qué se escogío esta función, necesitamos mirar su derivada. Esta es:

$$d'(x) = 6a^2(x-c)(4x^2 + 4bx + b^2 + 1) \qquad (6.73)$$

De esta podemos ver que la función d tiene sólo un punto estacionario, el cual es un mínimo en c. A la izquierda de c la derivada será siempre positiva. La variable b puede ser usada para controlar la asimetría alrededor de c, mientras que la variable a controla al tamaño de la derivada. El valor n en la Ec. (6.73) nos permitirá ajustar el valor esperado positivo del ruido.

Como en el algoritmo 1, se consideran $2R+1$ valores incrementales de φ sobre toda la amplitud de búsqueda de φ. Y se estiman a, b, c y n en la Ec. (6.72) de manera que $E(\varphi)$ en la Ec. (6.69) sea tan pequeño como sea posible. El valor mínimo será c y debe cumplir la restricción de la Ec. (6.71). Como anteriormente se aplicó al algoritmo 1, se estudian todos los φ's en la amplitud de la búsqueda, y se selecciona aquella en donde la Ec. (6.71) sea satisfecha y para la cual el valor del error sea él más pequeño que todos los demás donde la Ec. (6.71) también es satisfecha. La estimación del minimizador de d es el valor de c asociado a tal φ (y no la φ en sí misma).

El ajuste a la curva por la Ec. (6.72) puede ser realizado por un método no lineal de ajuste por mínimos cuadrados, pero es necesario tener un conocimiento a priori de los valores de los coeficientes; esto es, es necesario inicializarlos a unos valores cercanos a los correctos.

Con las mismas suposiciones que en el algoritmo 1: se tratan M valores experimentales, R es fija y T será el más grande entero menor a R/3; se presenta el pseudocódigo para el segundo algoritmo y continuamos con un procedimiento para calcular los valores iniciales de los coeficientes de la Ec. (6.72).

Algoritmo No. 2 para estimar el ángulo de inclinación

Inicializar $C = \varphi_0$, Error $= \infty$;
For $1 \le m \le M - 2R$ **do**
set
$y(r) = \mathrm{d}\left(\varphi_0 + ms + (r-1)s\right); \qquad 1 \le r \le 2R+1$

$l = \sum_{r=R-3T}^{R-T} y(r)$

$k = \sum_{r=R-T+1}^{R+T+1} y(r)$

$h = \sum_{r=R+T+2}^{R+3T+2} y(r)$

if $k \geq \min(l, h)$ **then continue with for**
assign los valores iniciales a a, b, c y n (detalles más abajo).
if no es posible **then continue with for**
find los valores de a, b, c y n en la Ec. (6.72) de manera que

$e = \sum_{r=1}^{2R+1} [y(r) - \mathrm{y}(r)]^2$

sea tan pequeño como sea posible.
if $e < E$ y
$R - T \leq c - 1 \leq R + T$

then set Error=e, $C = \varphi_0 + ms + (c-1)s$
enddo
if $C = \varphi_0$, entonces la búsqueda ha fallado y **stop**
Output C (como nuestro estimador del ángulo de inclinación)
End

Asignamiento de los valores iniciales

El primer coeficiente es fácil: $c = R + 1$. Ahora se crean tres ecuaciones con las incógnitas a, b, n, las cuales serán:

$$l = a^2[u_l b^2 + v_l b + w_l] + n \tag{6.74}$$
$$k = a^2[u_k b^2 + v_k b + w_k] + n \tag{6.75}$$
$$h = a^2[u_h b^2 + v_h b + w_h] + n \tag{6.76}$$

El lado derecho de estas tres ecuaciones es obtenido de la ecuación (6.72) sustituyendo $c = R+1$ y $x = R-2T+1$, $x = R+1$, $x = R+2T+1$, respectivamente. La idea es que nuestros valores iniciales sean aquellos de un polinomio de cuarto orden cuyo mínimo está en $x = R+1$ y cuyos valores $x = R-2T+1$, $x = R+1$, $x = R+2T+1$ sean proveídos por los promedios resultantes del primer paso de nuestro algoritmo.

Manipulando las ecuaciones (6.74)-(6.76), obtenemos la ecuación cuadrática

$$\begin{vmatrix} l & k & h \\ u_l & u_k & u_h \\ 1 & 1 & 1 \end{vmatrix} b^2 + \begin{vmatrix} l & k & h \\ v_l & v_k & v_h \\ 1 & 1 & 1 \end{vmatrix} b + \begin{vmatrix} l & k & h \\ w_l & w_k & w_h \\ 1 & 1 & 1 \end{vmatrix} = 0 \tag{6.77}$$

donde || denota el determinante. Si esta ecuación cuadrática tiene raíces reales, entonces alguna de ellas puede ser usado como valor inicial para b. En la práctica, se escoge entre cualquiera de los dos. Si no hay raíces reales, entonces el asignamiento de los valores iniciales no es posible. Ahora, para cada una de las raíces reales, podemos estimar un valor para a usando:

$$a^2 = \frac{l-k}{(u_l b^2 + v_l b + w_l) - (u_k b^2 + v_k b + w_k)} \tag{6.78}$$

Si el término del lado derecho no es positivo para cualquiera de las raíces reales de (6.77), entonces, nuevamente, el asignamiento de los valores iniciales no es posible. Finalmente, el valor de n puede obtenerse de cualquiera de las ecuaciones (6.74)-(6.76).

6.3.6 Experimentos y Resultados

Para resumir el proceso, la pasos a seguir sobre una imagen de entrada son:

1. Cortar la imagen de entrada (la imagen original) al tamaño de la región de interés.
2. Expandir la imagen cortada a otra de tamaño impar.
3. Calcular $d(\varphi)$, Ec. (6.67).
4. Encontrar el valor mínimo de la función d por ajuste a una curva conocida.

En lo siguientes experimentos vamos a considerar un error máximo de ± 6 grados en al ángulo de inclinación, y la búsqueda se hará en pasos de $\pi/2048$; por lo tanto tendremos $2(68)+1 = 137$ puntos de datos experimentales sobre ± 68 unidades delta en el ángulo de inclinación φ.

Experimento 1

En este primer experimento se probaron los pasos 2 y 3 realizando lo siguiente:

- Se genera una imagen con sólo ruido, ruido Gaussiano de media cero y desviación estándar de 0.20. Por lo tanto, el 99% de los valores de la imagen están entre 0.6 y -0.6.
- Se asumió que tenemos una simetría de orden 8 ($s = 8$). Aplicando la Ec. (6.43) resulta $n = 3$, esto es, existen 3 líneas comunes.
- Para probar que la Ec. (6.66) es una variable aleatoria de media σ^2, se tomarán los valores promedio de la Ec. (6.67), de manera que se puede reescribir la Ec. (6.47) como:

  $$d(\varphi) = \sum_{i=1}^{3} \overline{D}(\rho_i) \tag{6.79}$$

  el valor esperado de la Ec. (6.79) debe ser $3(0.20)^2 = 0.12$.

Resultados

Se hicieron seis ensayos de la Ec. (6.79) para los casos:

1. Para una imagen de entrada de tamaño 595×595 pixels. La región de interés es toda la misma imagen. Esta imagen de entrada ya es de tamaño impar, por lo tanto no se aplicó el paso 2. Las gráficas resultantes pueden verse en la Fig. 6.27.

2. Para una imagen de entrada de tamaño 123×123; y de nuevo, no se aplicó el paso 2. Las gráficas se muestran en la Fig. 6.28.

3. Para una imagen de entrada de tamaño 64×64 y aplicando el paso 2 para expandir la imagen al tamaño impar de 123×123. La región de interés es toda ésta última imagen expandida. El resultado puede verse en las gráficas de la Fig. 6.29.

Observando las gráficas de la Fig. 6.27 se confirma que la ecuación (6.66) es una variable aleatoria de media σ^2.

Como puede observarse en las gráficas de la Fig. 6.28, la curva de la función $d(\varphi)$ varía suavemente con respecto a φ para imágenes de ruido puro. Esto en sí mismo no es una sorpresa: tenemos menos pixels sobre los cuales se calculan las sumatorias de las líneas de proyección (en la Ec. 6.45 en la 101); y además, si el cambio de φ es pequeño, los ρ_1, ρ_2 y ρ_3 resultantes cambian solo un poco y así las líneas sobre las cuales se están localizando recorren, esencialmente, los mismos pixels. De esta manera, se ha demostrado que la suavidad de $d(\varphi)$ para imágenes de ruido puro depende del tamaño del pixel y de si la máscara circular, la usada para expandir la imagen a tamaño impar (ver la Subsec. 6.3.3), ha sido aplicada o no. La máscara circular hace este suavizamiento aún más extremo, como se observa en las gráficas de la Fig. 6.29, ya que causa que el ruido entre pixels cercanos esté correlacionado.

Un comentario más se puede anexar aquí: Aunque el valor esperado de $d(\varphi)$, para las imágenes con ruido puro, es σ^2, los valores tienen una distribución exponencial. Si $\mathbf{x}$ es nuestra variable aleatoria con una distribución Gaussiana como:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{x^2}{2\sigma^2}\right] \tag{6.80}$$

y si $\mathbf{y}$ es nuestra nueva variable aleatoria, $\mathbf{y} = \mathbf{x}^2$; la densidad de $\mathbf{y}$ en términos de la densidad de $\mathbf{x}$ resulta [81]

$$f_y(y) = \frac{1}{\sigma\sqrt{2\pi y}} \exp\left[-\frac{y}{2\sigma^2}\right] U(y) \tag{6.81}$$

donde $U(y)$ es la función escalón. En el caso especial para $\sigma = 1$, la Ec. (6.81) representa una función de densidad chi-cuadrada con un grado de libertad.

Experimento 2

Para realizar este experimento se construyó un "fantasma", un modelo sintético de la proteína CCT que tiene una simetría cilíndrica de orden ocho (ver Fig. 6.26). El fantasma generado tiene dimensiones $64 \times 64 \times 64$ voxels. De este volumen tomamos 65 retroproyecciones bidimensionales de acuerdo a una simetría de inclinación cónica, a un ángulo de inclinación de 55° y a ángulos de rotación $\theta_i = i \cdot 360°/65, \quad 0 \leq i < 65$. Las proyecciones con calculadas a partir de las ecuaciones que describen cada elipsoide, y no de los voxels en sí mismos.

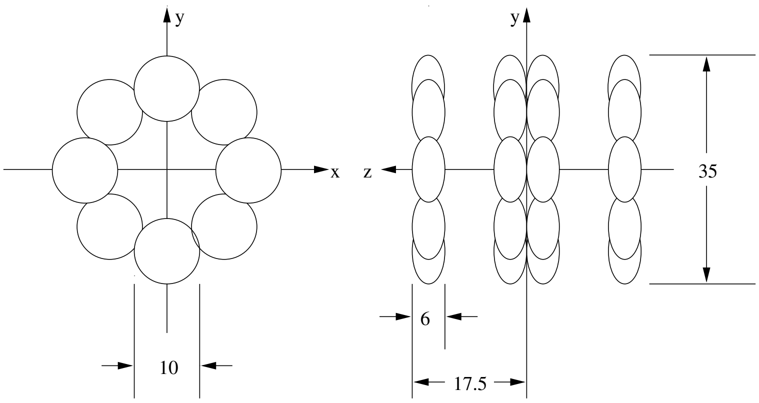


Figura 6.26: Diagrama del modelo sintético de la proteina CCT. Las unidades están en pixels

Se les sumó a las reproyecciones ruido Gaussiano de media cero y desviación estándar acorde para obtener una nivel fraccional de ruido de 0.20 (ver Ec. (6.36) sobre la pág. 94).

Se consideró una región de tamaño 48×48 pixels como la *región de interés*. Así la imagen expandida tiene de tamaño 91×91 pixels. Por lo tanto, tenemos 91 rayos para calcular la Ec. (6.67); un rayo por cada medio pixel de la imagen original.

Es importante recordar que las imágenes típicas de criomicroscopía electrónica son como las imágenes sintéticas generadas con un nfr= 0.80, desde el punto de vista subjetivo de la visualización.

Resultados

La estimación de los ángulos de inclinación, para varios casos, puede verse en la Fig. 6.30. En la misma figura, para todas las gráficas el eje y representa el ángulo de inclinación, en grados, y el eje x es el número de imágenes. Las gráficas de la primera columna, (a), (c) y (e), fueron obtenidas con el algoritmo no. 1. Las gráficas de la segunda columna, (b), (d) y (f), fueron obtenidas con el segundo algoritmo. Las gráficas (a) y (b) fueron obtenidas para el caso de las imágenes inclinadas sin ruido. Las gráficas (c) y (d) fueron obtenidas para las imágenes inclinadas con nfr= 0.20 y las (e) y (f) para la imágenes inclinadas con nfr= 0.80.

Un resultado aquí, para el caso de las imágenes sin ruido, es que no se puede estimar el ángulo de inclinación, sí para una de las imágenes, un eje de simetría cae exactamente sobre el eje x (todos los valores de la Ec. (6.67) son iguales a cero).

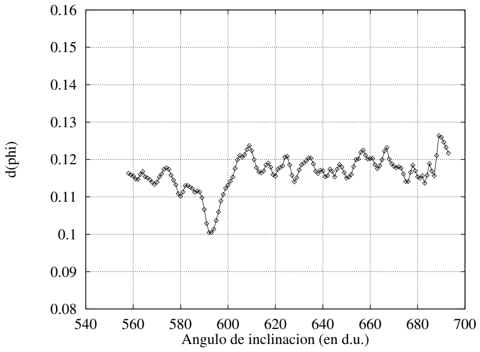
nfr	algoritmo 1	algoritmo 2
0.00	1/65	1/65
0.20	10/65	9/65
0.80	23/65	22/65

Tabla 6.6: Resultado del experimento de estimación del ángulo de inclinación. La tabla muestra las imágenes rechazadas versus el total de imágenes iniciales

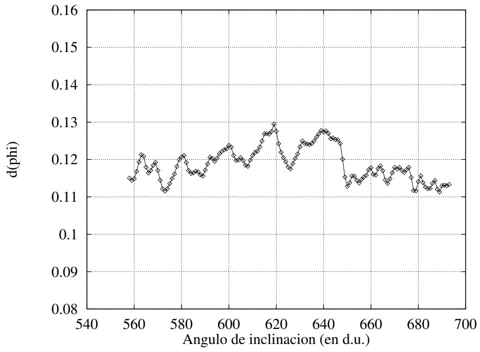
Una observación de los datos de la Tabla 6.6, de nuestros resultados para la estimación del ángulo de inclinación, en comparación con los resultados de la Tabla 6.5 (sobre la pág. 96) del método de la estimación de los ángulos de rotación y de inclinación, nos indica que hemos mejorado los resultados obtenidos con respecto al ruido presente en las imágenes. De la Tabla 6.6 se deduce que tenemos 55-56 imágenes, de un total de 65, a las que pudimos estimar el ángulo de inclinación, a un nfr de 0.20. En la Tabla 6.5 tenemos que sólo sobrevivieron 15-24 imágenes, también de un total de 65, a las que pudimos estimar el ángulo de inclinación y de rotación. Así, se demuestra el aprovechamiento de la simetría de un espécimen para mejorar la razón señal-a-ruido de nuestros datos.

Los resultados de la estimación del ángulo de inclinación de la Tabla 6.6 para un nivel fraccional de ruido de 0.80 deben tomarse con cuidado. Como se observa en la figura 6.31, todavía se puede estimar el ángulo de inclinación para muy malos valores iniciales del ángulo de inclinación. Las dos gráficas en la parte superior son para $\varphi_0 = 30°$ y las dos en la parte inferior son para $\varphi_0 = 45°$. Con esto se demuestra que los valores estimados para el caso de nfr$= 0.80$ dejan mucho que desear.

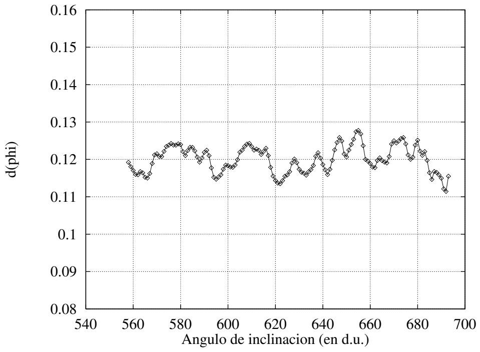
Sin embargo, al igual que para el método de la estimación de los ángulos de rotación y de inclinación, el nuevo método desarrollado en esta memoria para la estimación del ángulo de inclinación es aplicable a los datos que nos pueden proporcionar otras microscopías, tanto los procedentes del Microscopio de Transmisión con Barrido (STEM) [24] o a imágenes procedentes del Microscopio Electrónico con Filtración de Energía [85]. Estas dos microscopios nos generan imágenes con un razón señal-a-ruido más alto que las imágenes de criomicroscopía producidas por el Microscopía Electrónico de Transmisión (MET). También podrían aplicarse a imágenes tomadas en el modo *campo obscuro* [60, 8] del MET, las cuales tienen un nivel fraccional de ruido de 0.20 [41].



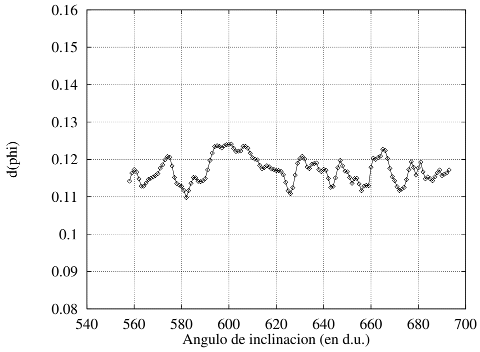
(a)



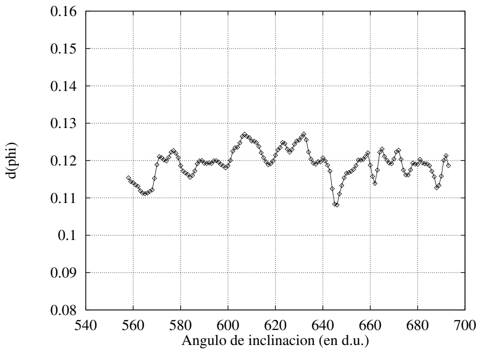
(b)



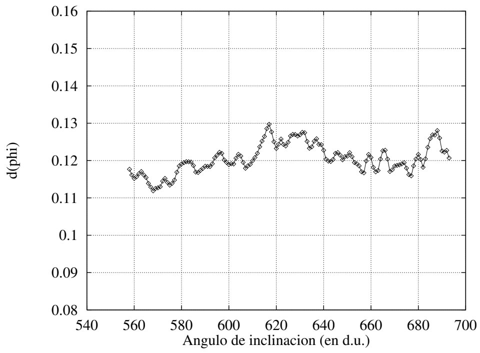
(c)



(d)

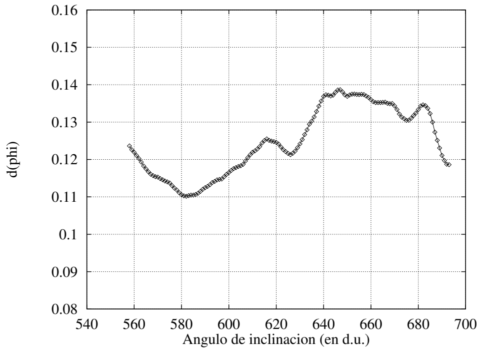


(e)

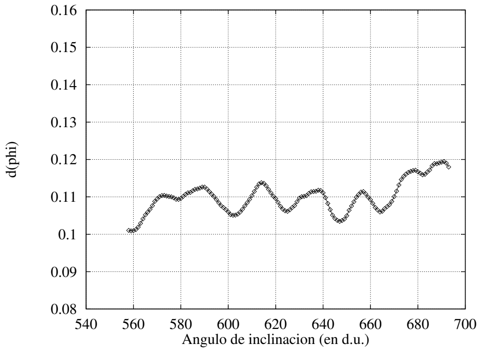


(f)

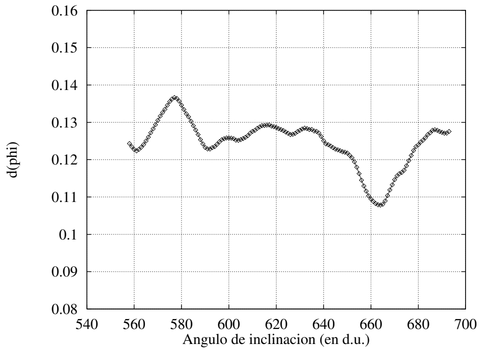
Figura 6.27: Resultados del experimento 1 para seis imágenes de 595×595, conteniendo sólo ruido, sin expandirlas



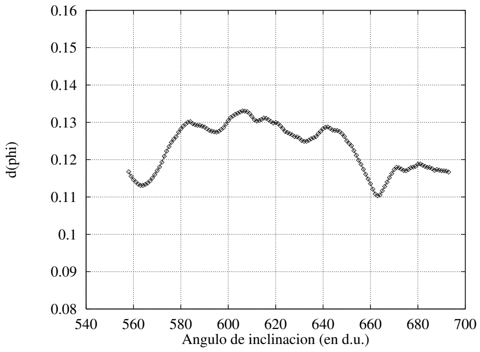
(a)



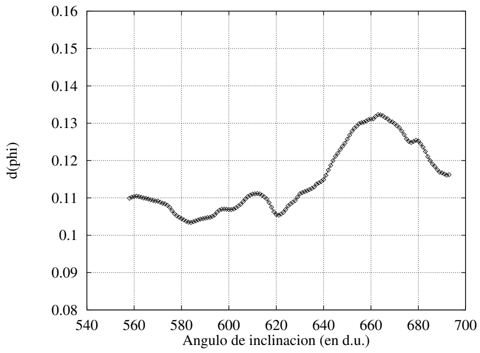
(b)



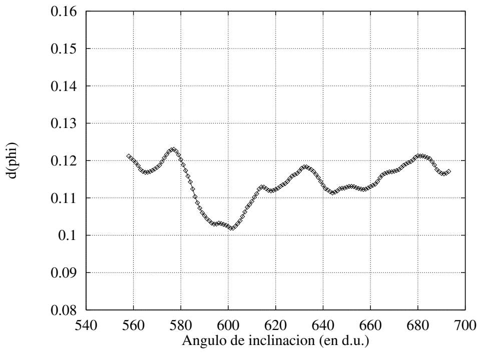
(c)



(d)

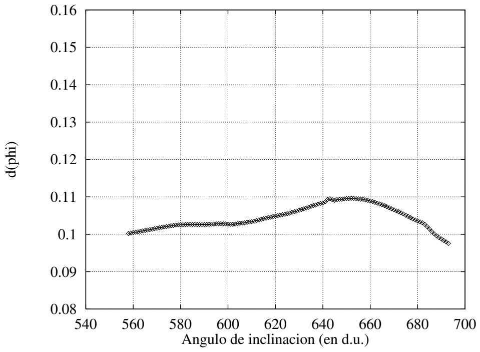


(e)

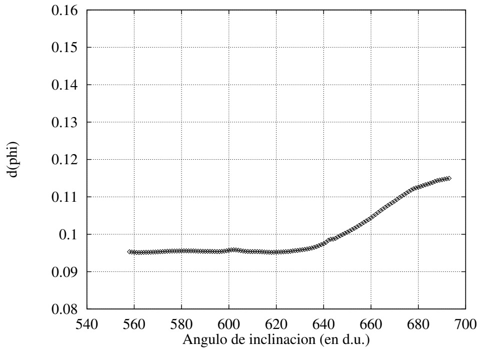


(f)

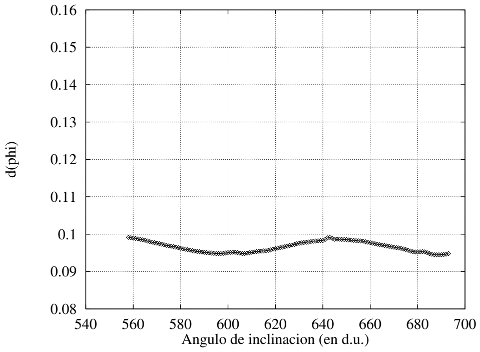
Figura 6.28: Resultados del experimento 1 para seis imágenes de 123×123 conteniendo sólo ruido, sin expandirlas



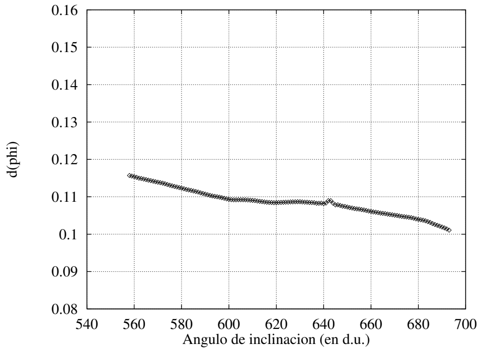
(a)



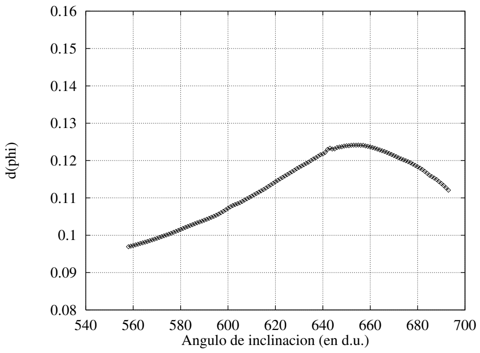
(b)



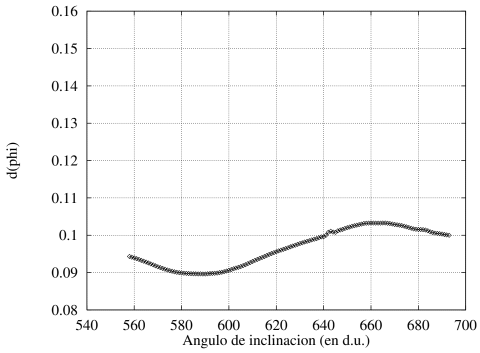
(c)



(d)



(e)



(f)

Figura 6.29: Resultados del experimento 1 para seis imágenes de 64×64, de sólo ruido, expandidas a una imagen de tamaño 123×123

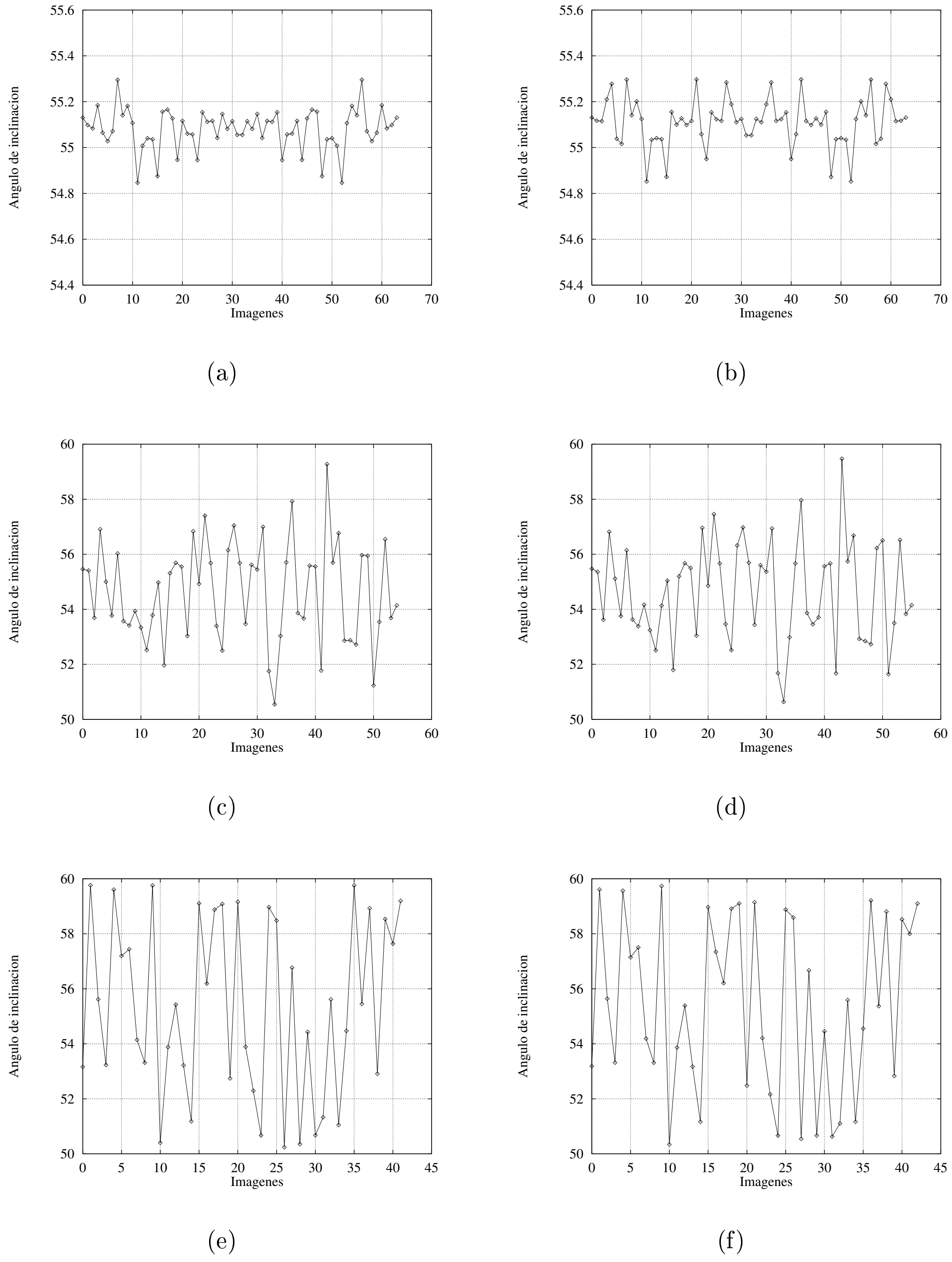
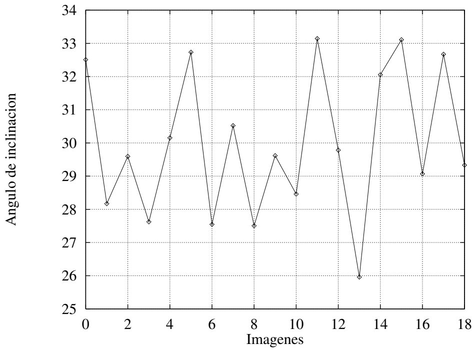
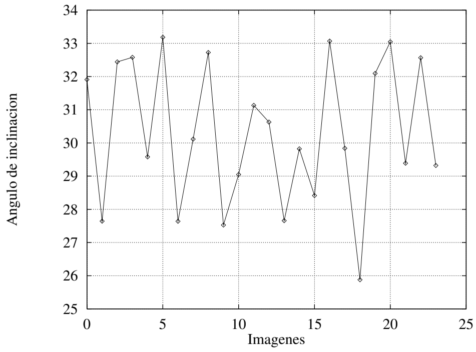


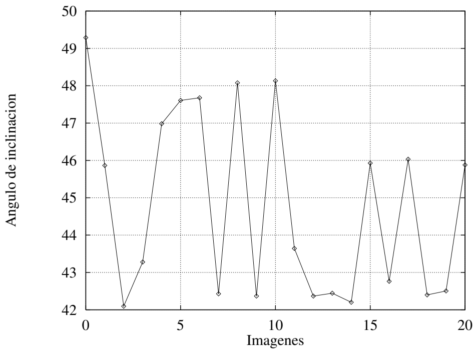
Figura 6.30: Resultados del experimento 2. Detalles en el texto



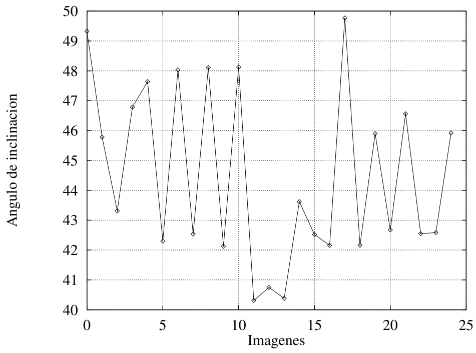
(a)



(b)



(c)



(d)

Figura 6.31: Otros resultados del experimento 2. Detalles en el texto

Capítulo 7

Experimentos con los Métodos Basados en Prototipo

En este capítulo se realizará una comparación de los dos métodos de refinamiento de la geometría basados en un prototipo: (1) El método badao en la comparación directa de las proyecciones (estudiado en detalle en el capítulo 4), y (2) El método basado en la transformada de Radon (estudiado en detalle en el capítulo 5).

Así mismo, se presenta una nueva metodología para realizar una reconstrucción tridimensional en la que se combina el procedimiento para estimar el centroide por segmentación de la masa (descrito en la sección 6.1), son el método (1) de refinamiento de la geometría. Esta metodología ha sido inspirada en los datos que conocemos sobre la proteína CCT [74] y podría aplicarse a la futura reconstrucción tridimensional de la misma [1]

7.1 El Modelo Sintético NCCT

Para los experimentos realizados en este capítulo se construyó un modelo tridimensional sintético, un fantasma, que denominaremos NCCT y que tenía de dimensiones $64 \times 64 \times 64$ voxels. Este fantasma es un modelo estilizado del fantasma CCT usado en capítulos anteriores (en la págs. 80 y 115). Un diagrama de las características geométricas del fantasma NCCT puede verse en la Fig. 7.1, este consta de 64 elipsoides y 16 esferas, todos ellos de densidad uno. En aquellas zonas donde existe la unión entre dos elipsoides de toma la función Booleana OR entre ellas, de esta manera la densidad para todo el fantasma sigue siendo igual a uno.

En la Fig. 6.3 se muestran varias vistas de la superficie del fantasma NCCT. Estas vistas fueron generadas con el paquete WEB [51].

[1] A la fecha de escritura de esta memoria, se está trabajando en la puesta a punto de la preparación y toma de micrografías por las técnica de criomicroscopía para la proteína CCT.

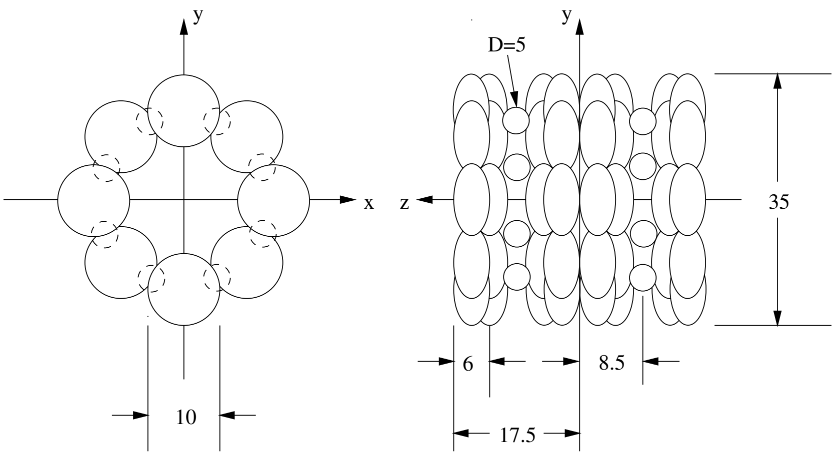
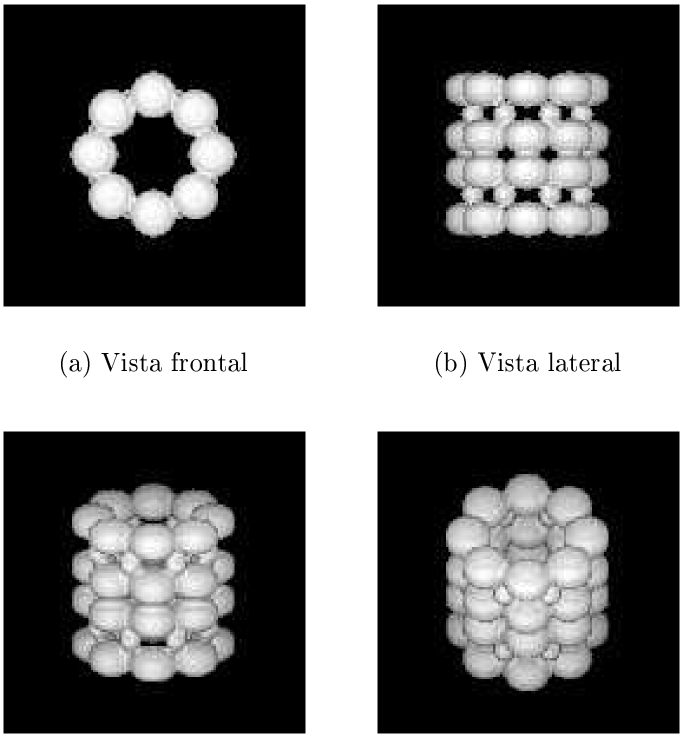


Figura 7.1: Diagrama del modelo sintético NCCT. Este modelo es una versión estilizada del modelo CCT de la Fig. 6.26 (en la pág. 115). Las unidades indican pixels.



(a) Vista frontal (b) Vista lateral

(c) Vista a $\varphi = 20°$ (d) Vista a $\varphi = 45°$

Figura 7.2: Varias vistas de la superficie del fantasma NCCT

7.2 Prueba de los Métodos Basados en Prototipo

Para probar la efectividad de los dos métodos disponibles para refinar la geometría de proyecciones a partir de un prototipo se diseño el siguiente experimento:

1. Se calcularon 173 proyecciones, de dimensiones 64×64 pixels, del fantasma NCCT a los ángulos $0° \leq \varphi \leq 90°$ y $0° \leq \theta < 45°$ y a un incremento $\Delta\varphi = 4°$ (ver Ec. 4.3). No necesitamos calcular proyeccciones en toda la amplitud del ángulo de rotación θ debido a la simetría cilíndrica de orden ocho que presenta el fantasma NCCT, de ahí que se calculen la proyecciones en la amplitud reducida del ángulo de rotación θ de $360°/8 = 45°$. Debido también a la misma simetría, no es necesario proyectar en toda la amplitud del ángulo de inclinación φ, entre $-90° \leq \varphi \leq 90°$, sino sólo en la mitad de ella. En la Fig. 7.3 se muestra el NCCT enmarcado en el sistema de coordenadas 3D, así como la zona donde se realizaron las proyecciones. Los ángulos a los que se calculan las proyecciones se generan con el programa *voea* (ver sec. 4.3) de la siguiente manera:

   ```
   voea 0.0 44.9 0.0 90.0 4.0 > ang.txt
   ```

   Y las proyecciones se calculan con el programa *proj3d* a los ángulos almacenados en el archivo *ang.txt* producido anteriormente:

   ```
   proj3d newcct.img ang.txt a0ta
   ```

   donde *newcct.img* es el nombre del archivo del volumen y *a0ta* es el patrón de los nombres con los que se almacenarán las proyecciones, desde la a0ta0000 hasta la a0ta0172 (ver la sec. 4.3).

2. Se asignan a las proyecciones calculadas en el paso anterior los valores de los ángulos que generan la aplicación de alguno de los dos métodos de refinamiento (por comparación directa de las proyecciones y por uso de la transformada de Radon). El espacio de búsqueda será realizado en $0° \leq \varphi \leq 90°$, $0° \leq \theta < 45°$, y a un $\Delta\varphi = 2°$.

7.2.1 Aplicación del método basado en la comparación directa de las proyecciones

De entrada suponemos que tenemos la lista de los nombres de las 173 proyecciones en un archivo al que llamaremos **a0t.sel**, con el formato de un archivo *sel* (ver la descripción de opciones del programa *ccf_3d* en la pág. 58). El fantasma está referido por el nombre del archivo del mismo: **newcct.img**.

La línea de comandos para la ejecución del programa paralelo **smaster3d** fue:

```
smaster3d 5 0.0 44.9 0.0 90.0 2.0 newcct.img a0t.sel 1 22 1 > out3D.txt
```

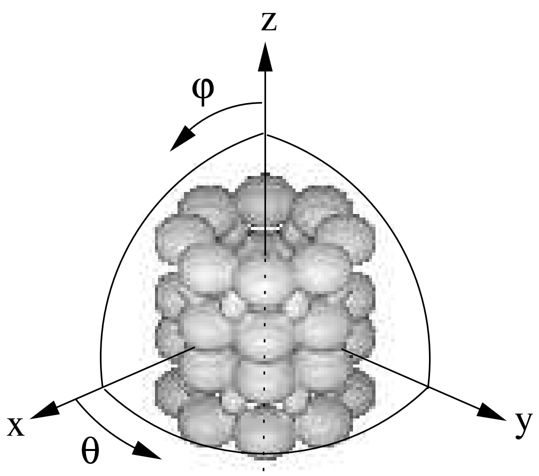


Figura 7.3: El fantasma NCCT empotrado en el sistema de coordenadas tridimensional de mano derecha. La línea punteada señala el octante donde se tomarán las proyecciones, aprovechando la simetría cilíndrica de orden 8, para $0° \leq \varphi \leq 90°$ y $0° \leq \theta < 45°$.

Como se puede apreciar en la línea de comandos, el programa paralelo se ejecutó para 5 procesos hijos. También se ve que se han tomado en las proyecciones la región entre los radios 1 y 22, a un incremento de un pixel, como la región que enmarca totalmente la señal y son los datos para muestrear la proyección a coordenadas polares (ver pseudocódigo en la pág. 4.5.4). Tal como se comentó anteriormente, el espacio de búsqueda se realiza en $0° \leq \varphi \leq 90°$, $0° \leq \theta < 45°$, a un $\Delta\varphi = 2°$.

El programa **smaster3d** se modificó para que realizara la búsqueda solamente en el intervalo señalado, *sin* realizar la búsqueda en las proyecciones reflejadas (no se realiza el cálculo de la línea 20 en el pseudocódigo del proceso hijo de la sec. 4.5.4).

7.2.2 La aplicación del método por medio de la transformada de Radon

Para la aplicación de este método es necesario realizar más operaciones que en el método anterior. En concreto, es necesario realizar la transformada de Radon de las proyecciones y del fantasma, y a su vez realizar las transformadas de Fourier de todas las transformadas de Radon.

La preparación de los datos para la aplicación del método de la transformada de Radon conllevó los siguientes pasos:

1. Realizar la transformada de Radon para cada una de las 173 proyecciones:

   `doradon2d a0ta0??? prad0??? 23 46 0.0 0.0 0 178 2 0`

   Donde "???" se debe sustituir por el número de la proyección respectiva, desde la 000 hasta la 172.

Los datos en la línea de comandos indican que se consideró como señal la zona dentro de una máscara circular de radio 23, y se consideran 46 muestras en la transformada de Radon (una por pixel). La máscara está centrada sobre la imagen en las coordenadas (0.0, 0.0) (que corresponde al centro de la imagen). Se realiza la transformada en la amplitud de 0° a 178°, inclusive, a un intervalo de muestreo de 2°. Finalmente, el ultimo cero en la línea de comandos indica que no se aplica ningún cambio a la imagen de entrada. Para más detalles ver la sección 5.3.

2. Realizar la transformada de Fourier de la transformada de Radon de las 173 proyecciones:

   fftra2d prad0??? pradfft??? n

   donde "n" indica que no se aplica filtraje alguno a la transformada de Radon.

3. Realizar la transformada de Radon del volumen, el fantasma NCCT:

   doradon3d newcct.img radncct.img -90 88 -90 88 2 23 46 0

   donde los parámetros son los comunes para realizar una transforma de Radon tridimensional al mismo muestreo con el que se realizaron las transformadas de Radon bidimensionales del paso 1.

4. Realizar la transformada de Fourier de la transformada de Radon del volumen:

   fftra3d radncct.img fftradncct.img n

Más detalles de las opciones disponibles para cada programa puede verse en la sección 5.3 de esta memoria.

Una vez realizados todos los pasos anteriores, se tienen los nombres de todas las transformadas de Fourier de las transformadas de Radon bidimensionales (la salida del paso 2) en un archivo que llamaremos "prf.sel".

La línea de comandos para la ejecución del programa paralelo **masradon** del método de refinamiento de la geometría usando transformadas de Radon fue:

masradon 5 fftradncct.img prf.sel 0.090.00.044.90.044.0 2 0 f > outradon.txt

Esto es, se usaron 5 procesos hijos y una búsqueda completa de la orientación en la amplitud especificada de los ángulos de Euler.

7.2.3 Resultados

La distribución de las orientaciones a los que se proyectó el fantasma NCCT puede verse en la Fig. 7.4(a). Recordemos, las proyecciones se realizaron a $0° \leq \varphi \leq 90°$ en el ángulo de inclinación y a $0° \leq \theta < 45°$ en el ángulo de rotación, a un muestreo angular $\Delta\varphi$ de 2°.

Los resultados de la orientación asignada a cada proyección se presentan de dos maneras: (1) en forma gráfica para visualizar la distribución espacial de las orientaciones asignadas por cada método (Figs. 7.4(b) y 7.4(c)) y (2) en forma tabular del histograma de los errores en el ángulo de inclinación (ver las Tablas 7.1 y 7.2). Aquí recordemos que la búsqueda angular se realiza a los mismos límites en que se proyectó, $0° \leq \varphi \leq 90°$ en el ángulo de inclinación y $0° \leq \theta < 45°$ en el ángulo de rotación, pero a un intervalo de muestreo de $\Delta\varphi = 2°$. Con estos datos de búsqueda se genera el archivo de ángulos de referencia con el programa **voea** de la siguiente forma:

```
voea 0.0 44.9 0.0 90.0 2.0 > angref.txt
```

Y las nuevas orientaciones son generadas a partir de este archivo *angref.txt* y los archivos de salida *out3D.txt* y *outradon.txt* de la siguiente manera:

```
vomdok angref.txt out3D.txt > angles3D.txt
```

y

```
vomdok angref.txt outradon.txt > anglesradon.txt
```

Así, los datos del archivo *angles3D.txt* son los que vemos en forma gráfica en la Fig. 7.4(b), y los datos del archivo *anglesradon.txt* en la Fig. 7.4(c).

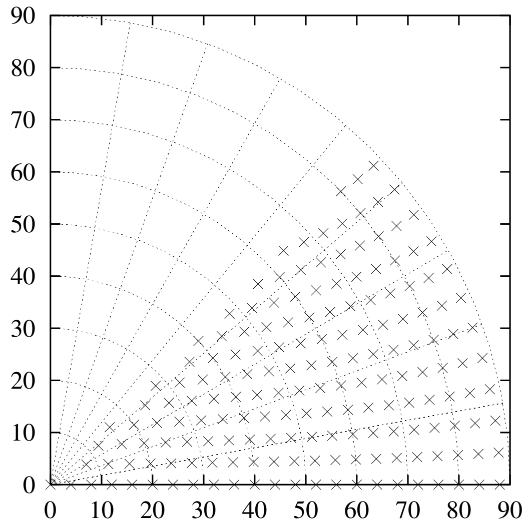
Los errores en el asignamiento angular para cada método, presentados en las Tablas 7.1 y 7.2, se hicieron a partir de los datos de los archivos *ang.txt* (ver la sec. 7.2, que contiene los valores correctos de cada (θ, φ) para cada proyección, y *angles3D.txt* y *anglesradon.txt*, que contienen los (θ, φ) asignados a cada proyección por cada método de refinamiento angular.

Error (en grados)	Número de Proyecciones	Porcentaje del Total	Porcentaje Acumulativo
0	170	98.27%	98.27%
1	0	0.00%	98.27%
2	3	1.73%	100.00%

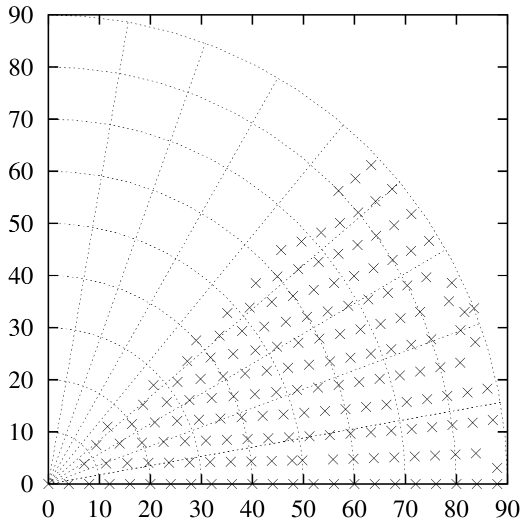
Tabla 7.1: Errores en el ángulo de inclinación asignado a las proyecciones por el *método por comparación directa de las proyecciones.*

El tiempo de ejecución total para el método de por comparación directa fue de 49 segundos. El tiempo de ejecución total para el método por transformada de Radon fue de 6 horas 16 minutos. El segundo método tarda 460 veces más que el primero.

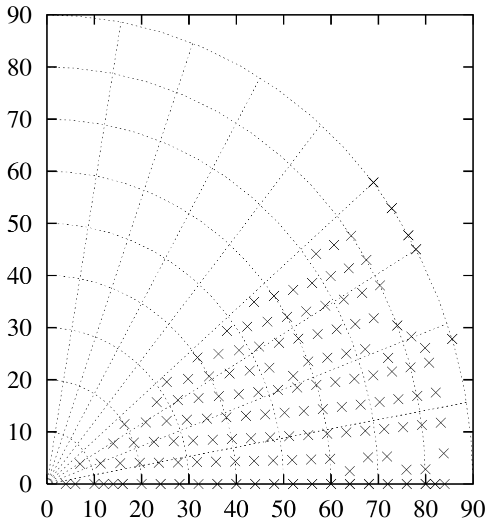
J. Frank [50] (p. 243) señala como umbral, para indicar la estabilización de la solución en el proceso iterativo del refinamiento angular, que el 80% de las partículas se muevan menos de 5° en el ángulo de inclinación. Es de pensarse que no se consideren los otros dos ángulos de Euler debido a que ellos están relacionados con rotaciones alrededor de ejes que no son de interés. El tercer ángulo de Euler, ψ,



(a) Referencia



(b) Comparación directa



(c) Transformada de Radon

Figura 7.4: Gráficas de los resultados. (a) Muestra la distribución de las orientaciones a las que se realizaron las 173 proyecciones. (b) La distribución de las orientaciones asignadas a las proyecciones por el *método de comparación directa de las proyecciones.* (c) La distribución de las orientaciones asignadas por el *método por transformadas de Radon.*

Error (en grados)	Número de Proyecciones	Porcentaje del Total	Porcentaje Acumulativo
0	151	87.28%	87.28%
1	0	0.00%	87.28%
2	18	10.40%	97.69%
3	0	0.00%	97.69%
4	1	0.58%	98.27%
5	0	0.00%	98.27%
6	3	1.73%	100.00%

Tabla 7.2: Errores en el ángulo de inclinación asignado a las proyecciones por el *método por transformadas de Radon*.

esta relacionado con la rotación “en el plano” de la partícula y puede ser inestable en su valor. El primer ángulo de Euler, θ, puede calcularse con gran precisión por la técnica de adquisición de datos de la geometría cónica aleatoria (ver sec. 2.8.2).

Bajo el criterio de J. Frank, ambos métodos de refinamiento de la geometría dan una solución correcta: El *método por comparación de las proyecciones*, en nuestro experimento, nos resulta en que el 100% de las imágenes están dentro de 2° en la desviación del ángulo de inclinación, como se observa en la Tabla 7.1; y el *método por transformadas de Radon* resulta que el 98% de las imágenes están dentro de una desviación de 5° en el ángulo de inclinación, como se ve en la Tabla 7.2.

Observando las gráficas de la Fig. 7.4, vemos que la gráfica de la Fig. 7.4(c) se parece menos a la gráfica con los ángulos correctos de proyección de la Fig. 7.4(a), que la gráfica de la Fig. 7.4(b). Concretamente la Fig. Fig. 7.4(c) no “encaja” bien a ángulos $\varphi > 80°$. Esto puede deberse a los valores discretos con que se realiza la transformada de Radon tridimensional, ya que de ella se extrae la proyección de referencia con la que se compara la proyección de nuestro experimento. El método por comparación directa de las proyecciones realiza la proyección directamente del volumen de referencia y exactamente en las direcciones donde se realizaron las proyecciones del experimento, sin pasar por el muestreo intermedio de la transformada de Radon. En cualquier caso ambos métodos nos dan resultados correctos según el criterio de estabilidad de J. Frank.

7.3 Un Método Híbrido para Obtener una Reconstrucción 3D

En esta sección se intenta dar una respuesta, mediante una simulación por ordenador, a esta pregunta: ¿Es viable y correcta una reconstrucción tridimensional tomando como prototipo un volumen sintético?. Dicho en otras palabras, ¿Es posible orientar las proyecciones experimentales sin pasar por las técnicas de adquisición de datos de la geometría cónica aleatoria o la reconstitución angular para obtener un primer prototipo?

Supongamos que ya tenemos un prototipo en baja resolución de una estructura 3D. Supongamos también que tenemos un grupo de proyecciones que queremos incorporar al volumen prototipo a fin de aumentar su resolución. De este grupo de proyecciones no conocemos ni sus parámetros de translación ni su orientación angular. ¿Cómo incorporarlas a nuestra estructura? Los dos métodos existentes de refinamiento de la estructura 3D usando un prototipo necesitan conocer primeramente la orientación angular de la proyección para efectuar la posible correción translacional de la misma: proyectan una imagen de referencia a los ángulos conocidos y corrigen la desviación translacional de la proyección dada en base a esa proyección de referencia (o reproyección). Pero contamos con un método para centrar las imágenes de proyección, el método de la *estimación del centroide mediante segmentación de la masa* presentado en la Sec. 6.1 de esta memoria, que es independiente de la orientación angular de la proyección y que estima el centro de masa de la proyección directamente de la información de los propios pixels que conforman la proyección. Para encontrar la orientación angular de las proyecciones necesitamos, invariablemente, el prototipo.

Supongamos ahora que tenemos un grupo de proyecciones con el que queremos obtener una reconstrucción 3D y no contamos con un prototipo de nuestra estructura 3D. Pero, supongamos, que contamos con dos, o varias, imágenes medias provenientes del análisis bidimensional de dos, o varias, vistas preferentes enlas que yace nuestro espécimen en la preparación para la toma de imágenes en el microscopio electrónico. Con la información estructural de estas imágenes medias, y con alguna información adicional *a priori*, como puede ser la simetría de la estructura, es posible construir un prototipo inicial que nos permita orientar angularmente el grupo de proyecciones. El centro de las proyecciones puede ser conocido aplicando el método de la *estimación del centroide mediante la segmentación de la masa*. Y así, con el grupo de proyecciones con los parámetros de translación y orientación conocidos se puede realizar una reconstrucción tridimensional, usarla como prototipo, y comenzar el proceso iterativo de refinamiento de la geometría.

En esta sección no se quiere dar una respuesta general a todos los problemas de reconstrucción tridimensional, sino un planteamiento metodológico que pueda aplicarse a algún caso concreto. Es específico, en esta sección se realizará una prueba por ordenador de (1) como incorporar nuevas proyecciones a una reconstruccion tridimensional existente, (2) la viabilidad de usar un modelo sintético como prototipo inicial, y (3) una prueba del comportamiento de los métodos de refinamiento de la geometría basados en un prototipo.

7.3.1 Condiciones

Tenemos al fantasma NCCT (ver la Fig. 7.4(c)), sobre él se realizaron las siguientes operaciones:

1. Se proyectaron 250 imágenes a los valores en los ángulos de Euler θ aleatorio

entre $0°$ y $360°$; $\varphi = 0°$, para todas las imágenes; y $\psi = 0°$ también para todas la imágenes. Una muestra de estas imágenes se puede ver en la Fig. 7.5. A las 250 imágenes proyectadas se les sumó un ruido Gaussiano acorde para obtener un nivel fraccional de ruido de 0.80, que recordemos, corresponde a las imágenes típicas de criomicroscopia obtenidas con el Microscopio Electrónico de Transmisión, desde el punto de vista subjetivo de la visualización. Estas 250 imágenes son muestras de una sola vista del fantasma NCCT, a la que se llamó "vista frontal".

2. Se proyectaron otro grupo de 250 imágenes a los valores en los ángulos de Euler θ aleatorio entre $0°$ y $360°$; $\varphi = 90°$, para todas las imágenes; y ψ también aleatorio entre $0°$ y $360°$. Varios ejemplos de estas imágenes se muestran en la Fig. 7.5. A todas las 250 imágenes proyectadas se les sumó un ruido Gaussiano acorde para obtener un nivel fraccional de ruido de 0.80. Estas 250 imágenes son muestras de una sola vista del fantasma NCCT, a la que se llamó "vista lateral".

3. A ambos grupos de imágenes se las alineo translacional y rotacionalmente con el algoritmo iterativo de *alineamiento con una referencia dinámica* [83] que esta disponible en el paquete XMIPP [73], para obtener dos imágenes medias características de cada "vista". Estas medias presentan una razón señal-a-ruido alta que nos permite observar detalles imposibles de ver en las imágenes originales con ruido. Ambas imágenes medias pueden verse en la Fig. 7.6. Las gráficas del resultado del cálculo de la resolución para ambas medias puede verse en la Fig. 7.7. Para el cálculo de la resolución de utilizó la medida de la Correlación en Anillos de Fourier con el algoritmo presentado en esta memoria en el Apéndice B.

4. Con ambas medias, y considerando "a priori" que nuestro volumen a reconstruir presenta una simetría rotacional de orden 8, se construyó un modelo tridimensional, al que se le llamó *0thmodel*. Esté esta la particularidad más interesante del experimento: aunque ya sabemos de antemano que nuestro fantasma NCCT tiene una simetría rotacional de orden 8 y conocemos, por lo tanto, el volumen al que queremos llegar con nuestra reconstrucción tridimensional, aquí suponemos que tenemos dos vistas de un volumen y a partir de ellas generamos un modelo sintético, con la consideración de la simetría. En una aplicación real, la imposición de la simetría es un dato que se suministra como información *a priori*, y que tiene que ser validado explícitamente.

   Nuestro modelo *0thmodel* se presenta en la Fig. 7.8, este consta de 32 elipsoides. Los datos de las dimensiones para cada elipsoide se tomaron directamente de las imágenes de las medias de la Fig. 7.6. El *Othmodel* fue generado usando el paquete XMIPP [73].

5. A partir del fantasma NCCT se realizan 173 proyecciones a los ángulos $0° \leq \theta < 45°$ y $0° \leq \varphi \leq 90°$ y a un incremento $\Delta\varphi = 4°$. A las proyecciones se les

adicionó un ruido Gaussiano acorde para obtener un nivel fraccional de ruido de 0.80. Estas imágenes son las que conoceremos de ahora en adelante como nuestras *proyecciones experimentales.*

7.3.2 Experimento

Nuestro objetivo fue reconstruir el fantasma NCCT a partir de las proyecciones experimentales, usando el *0thmodel* como nuestro primer prototipo, aplicando el refinamiento iterativo de la geometría. En este experimento se usó el *método de refinamiento por comparación directa de las proyecciones* debido a que es el que conlleva menor tiempo de ejecución, como se ha mostrado en la sección anterior (en la sec. 7.2.3).

Las proyecciones experimentales, de las que suponemos primeramente que no conocemos ni su centro de masa ni su orientación angular, se alinean, al inicio, translacionalmente centrándolas en su centro de masa usando el nuevo *método de la estimación del centroide mediante segmentación de la masa*, presentado en esta memoria en la sección 6.1. Las orientaciones angulares iniciales de las proyecciones experimentales son asignadas de acuerdo al modelo 3D *0thmodel*, usando el *proceso de refinamiento de la orientación.* Este proceso conforma la segunda parte del *método de refinamiento por comparación directa de las proyecciones.*

Se realiza una reconstrucción tridimensional ya con nuestras proyecciones experimentales con sus parámetros translacionales y rotacionales conocidos. La reconstrucción tridimensional será nuestro *1stmodel.*

A partir de aquí se aplica el proceso iterativo del refinamiento de la geometría por el *método de refinamiento por comparación directa de las proyecciones.* Las proyecciones experimentales se centran ahora translacionalmente usando el *proceso de refinamiento de los parámetros de translación* y su orientación angular es también asignada por el *proceso de refinamiento de la orientación.* Ambos procesos se realizan usando el *1stmodel* como prototipo.

Se realiza una nueva reconstrucción tridimensional con los parámetros refinados y se itera hasta la convergencia.

7.3.3 Resultados

El alineamiento translacional inicial, por el *método de la estimación del centroide mediante segmentación de la masa*, nos resultó en 157 proyecciones experimentales a las que se pudo encontrar el centroide. La clasificación de las máscaras (ver Sec. 6.1), en la Fig. 7.9, nos indica que todas las máscaras son útiles, no se desechó ninguna, por lo que iniciamos con 157 proyecciones experimentales. Puesto que estamos trabajando con datos simulados, conocemos perfectamente la localización del centroide de las proyecciones, de forma que podemos calcular el error inicial entre los desplazamientos encontrados por nuestro precedimiento y los teóricos. De

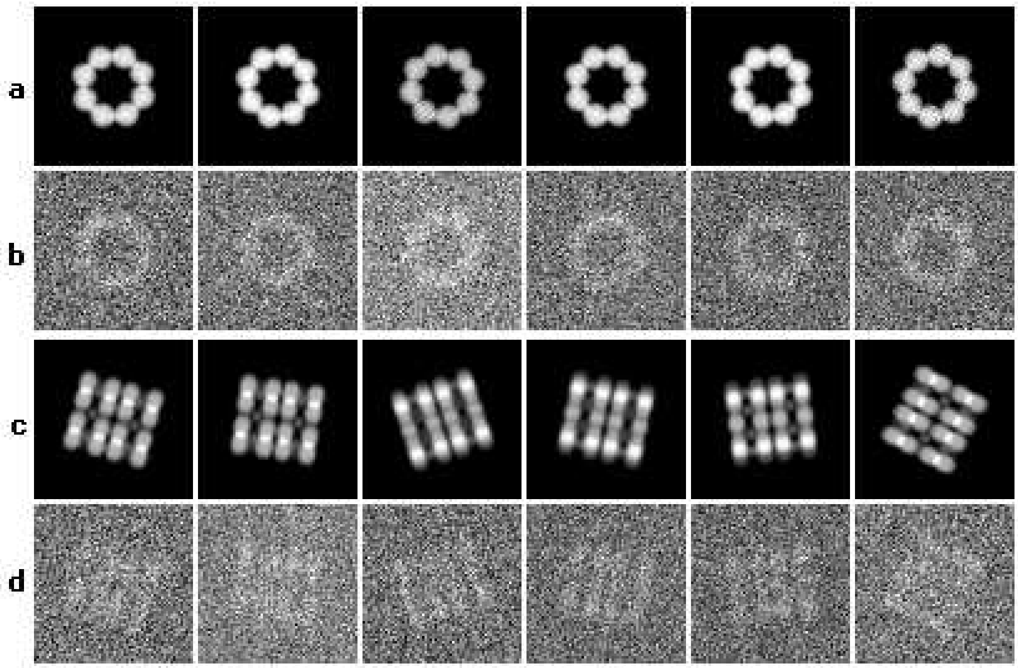
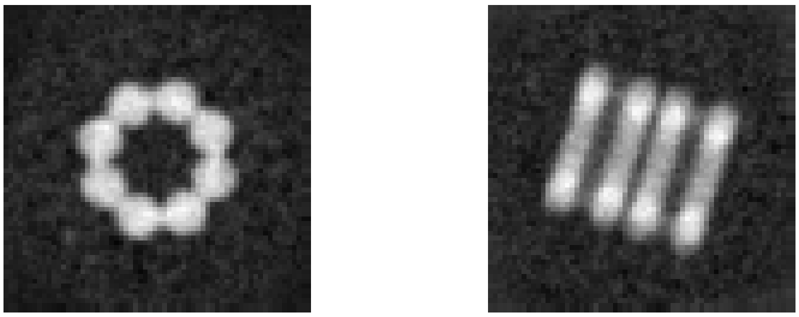
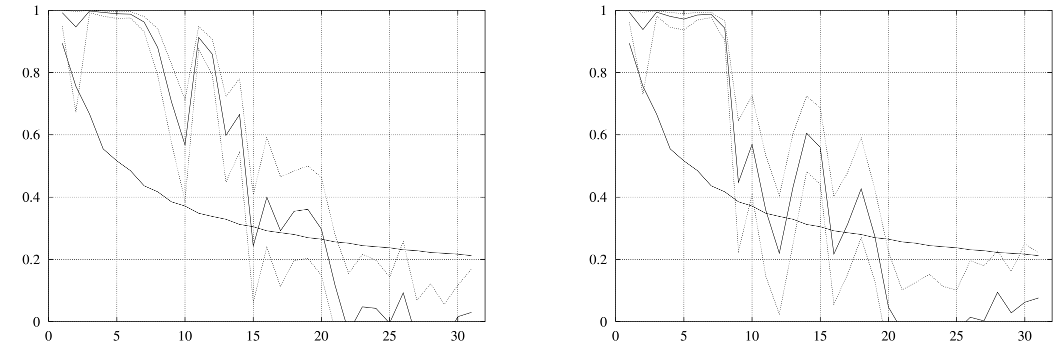


Figura 7.5: Galería de imágenes representativas. (a) 6 proyecciones del fantasma NCCT a θ aleatorio y $\varphi = 0°$. (b) Las mismas imágenes de (a) más un ruido a un nfr$= 0.80$. (c) 6 proyecciones a θ y ψ aleatorios y $\varphi = 90°$. (d) Las misma imágenes en (c) más un ruido Gausiano acorde para tener un nfr$= 0.80$.



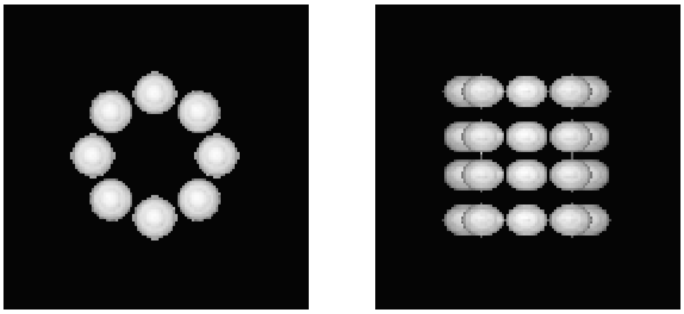
(a) (b)

Figura 7.6: Imágenes medias en (a) de la vista frontal y (b) de la vista lateral.



(a) (b)

Figura 7.7: Distribuciones de la Correlación en Anillos de Fourier (FRC) para las medias de la Fig. 7.6. En (a) correspondiente a la vista frontal (Fig. 7.6(a)), la medida de resolución es de 14 pixels de Fourier. En (b) el FRC correspondiente a la vista lateral, la medida de resolución es de 11 pixels de Fourier. Si suponemos que nuestras proyecciones fueron muestreadas a un tamaño de 4.7 Å/pixel, las resoluciones serían de $1/21$ Å^{-1} y $1/27$ Å^{-1}, para las medias de la vista frontal y lateral, respectivamente.



(a) (b)

Figura 7.8: , Visualización de superficie del prototipo generado, llamado "0thmodel", a partir de las medias de la Fig. 7.6. En (a) la vista superior y en (b) la vista lateral.

esta forma obtuvimos que el error cuadrático medio (ver Ec. 6.7) inicial en los desplazamientos encontrados fue de: $e_x^0 = 0.689047$ pixels^2 y $e_y^0 = 0.420786$ pixels^2.

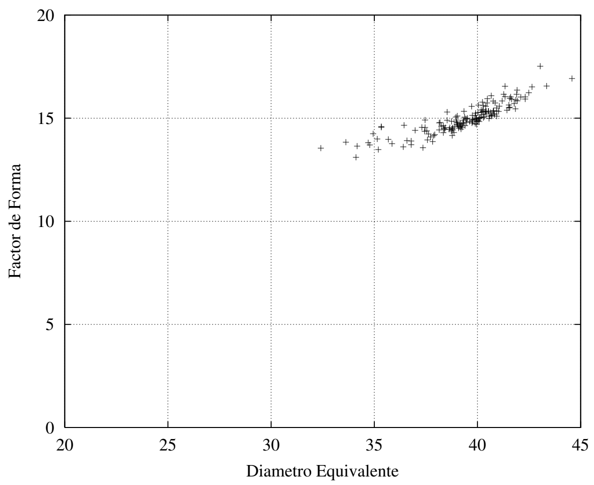


Figura 7.9: Resultados de la clasificación para las 157 máscaras encontradas de las 173 proyecciones experimentales. Claramente se ve un sólo grupo, por lo que todas las 157 máscaras están bien.

Las 157 proyecciones experimentales fueron filtradas con un filtro pasa-bajas a los radios de Fourier 8-12, con un perfil coseno alzado. La frecuencia de corte está en el pixel 10 de Fourier. Si suponemos que cada pixel es de tamaño 4.7 Å, la frecuencia de corte estaría en 1/30 Å^{-1}. Una muestra de las imágenes sin filtrar y filtradas se puede ver en la Fig. 7.10

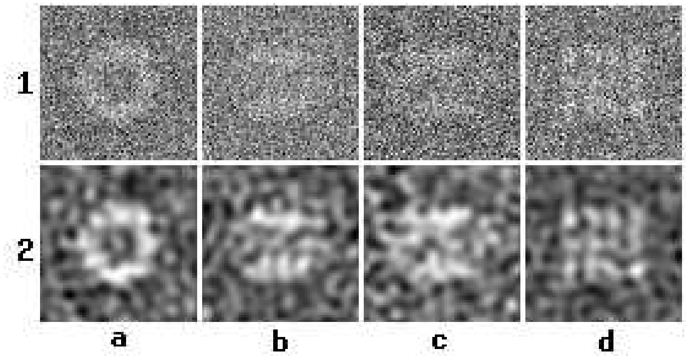


Figura 7.10: Ejemplos de las proyecciones experimentales. En la fila 1 cuatro proyecciones a un fnl=0.80, en la fila 2 las proyecciones filtradas con un filtro paso-bajas entre los radios 8-12 de Fourier. Las proyecciones en la columna "a" fueron realizadas a $(0°, 0°, 0°)$ $[(\theta, \phi, \psi)]$, las de la columna "b" a $(0.0°, 32.0°, 0.0°)$; en la columna "c" a $(0.0°, 48.0°, 0.0°)$; y "d" a $(0.0°, 88.0°, 0.0°)$.

Las orientaciones de las 157 proyecciones experimentales filtradas fueron asignadas a partir del modelo *0thmodel*, en la Fig. 7.8, usando el programa **smaster3d**.

Los datos del histograma del cambio que presenta el valor del ángulo de inclinación, con respecto a las orientaciones correctas, puede verse en la Tabla 7.3. La primera línea del histograma de la Tabla 7.3 nos dice que existen 136 proyecciones experimentales las cuales tienen un error menor a 5° en el valor asignado del ángulo de inclinación.

Error (en grados)	Número de Proyecciones	Porcentaje del Total	Porcentaje Acumulativo
5	136	86.62%	86.62%
10	17	10.83%	97.45%
15	3	1.91%	99.36%
20	1	0.64%	100.00%

Tabla 7.3: Datos del histograma del cambio en el ángulo de inclinación asignado a las proyecciones usando el *0thmodel* como prototipo con respecto a las orientaciones correctas

Con las 157 proyecciones experimentales, con el centro de masa y las orientaciones iniciales conocidas, se realizó una reconstrucción tridimensional con el programa SPIDER [51] usando el algoritmo de retroproyección pesada general (para una geometría arbitraria) [93]. El volumen reconstruido se filtró con un filtro pasa bajas a los radios de Fourier 8-12. A la reconstrucción tridimensional filtrada le llamamos *1stmodel*. La reconstrucción 3D se realizó con las proyecciones experimentales *sin* filtrar.

Ahora se aplica una iteración completa del *método por comparación directa de las proyecciones*:

1. Las proyecciones experimentales filtradas fueron centradas translacionalmente usando el *1stmodel* como prototipo. Se utilizó el programa **alignxy** (ver sec. 4.3). El error cuadrático medio (ver Ec. 6.7) en los desplazamientos encontrados fue de: $e_x^1 = 0.273988$ pixels^2 y $e_y^1 = 0.165970$ pixels^2, que son menores comparándolos con nuestros errores cuadráticos medios iniciales de $e_x^0 = 0.689047$ pixels^2 y $e_y^0 = 0.420786$ pixels^2

2. Usando también el *1stmodel* como prototipo se encuentra la orientación de las proyecciones experimentales filtradas, usando el programa **smaster3d**. El cambio en la asignación del valor en el ángulo de inclinación, con respecto a su valor correcto, puede verse en la Tabla 7.4

En esta etapa del proceso ya tenemos una iteración del refinamiento. Con los desplazamientos y las orientaciones refinadas, usando el *1stmodel*, de las proyecciones experimentales, se realiza de nuevo una reconstrucción 3D, llamada ahora *2ndmodel*. Varias vistas del volumen 2ndmodel pueden verse en la Fig. 7.11(e-h).

La prueba de convergencia puede realizarse ahora, ya que contamos con dos orientaciones asignadas: (1) las orientaciones iniciales, asignadas usando el *0thmodel* como prototipo, y (2) las primeras orientaciones asignadas usando el *1stmodel* como

Error (en grados)	Número de Proyecciones	Porcentaje del Total	Porcentaje Acumulativo
5	146	92.99%	92.99%
10	11	7.01%	100.00%

Tabla 7.4: Datos del histograma del cambio en el ángulo de inclinación asignado a las proyecciones usando el *1stmodel* como prototipo con respecto a las orientaciones correctas

prototipo. Los datos del histograma del cambio en el valor del ángulo de inclinación entre las asignaciones (1) y (2) puede verse en la Tabla 7.5. Como se observa en la misma Tabla 7.5, 139 proyecciones, el 88.54% del total, tienen un cambio menor a 5°, por lo que se cumple la condición, bajo el criterio de J. Frank [50], del umbral que indica la estabilización de la solución en el proceso iterativo del refinamiento angular: el 80% de las imágenes se mueven menos de 5° en el ángulo de inclinación.

Error (en grados)	Número de Proyecciones	Porcentaje del Total	Porcentaje Acumulativo
5	139	88.54%	88.54%
10	14	8.92%	97.45%
15	2	1.27%	98.73%
20	2	1.27%	100.00%

Tabla 7.5: Datos del histograma del cambio en el ángulo de inclinación asignado a las proyecciones en la primera iteración del refinamiento. Como claramente se ve, se ha estabilizado el proceso: el 88.64% de las imágenes tiene un cambio menor de cinco grados en el ángulo de inclinación.

El volumen final, nuestra reconstrucción tridimensional obtenida por el proceso de refinamiento, se observa Fig. 7.11(e-h), como una visualización de superficie. Esta reconstrucción corresponde al ”volumen experimentalºbtenido a partir de los proyecciones del fantasma NCCT y puede considerarse el NCCT al que se quería llegar en todo el proceso. Como comparación se presenta en Fig. 7.11(a-h) varias vistas de superficie del fantasma NCCT filtrado con un filtro pasa-bajas a los radios 8-12 de Fourier. Visualmente, el NCCT y nuestra reconstrucción, son equivalentes.

Con estos resultados se ha demostrado la eficacia del refinamiento de la geometría y la posibilidad de usar como prototipo un volumen sintético, bajo alguna consideración *a priori*, que en este caso fue el orden 8 en la simetría cilíndrica.

Un resultado, aparte, sobre la aseveración de Roseman *et al* [96], y de J. Frank [50] (p. 243), que señalan sobre la convergencia del proceso iterativo del *método de refinamiento por comparación directas de las proyecciones* en *unos pocos pasos*. Es nuestro experimento bastó un sólo paso del proceso para la convergencia del mismo, con lo que se puede inducir que con datos experimentales el número de iteraciones sea de 3 a 5.

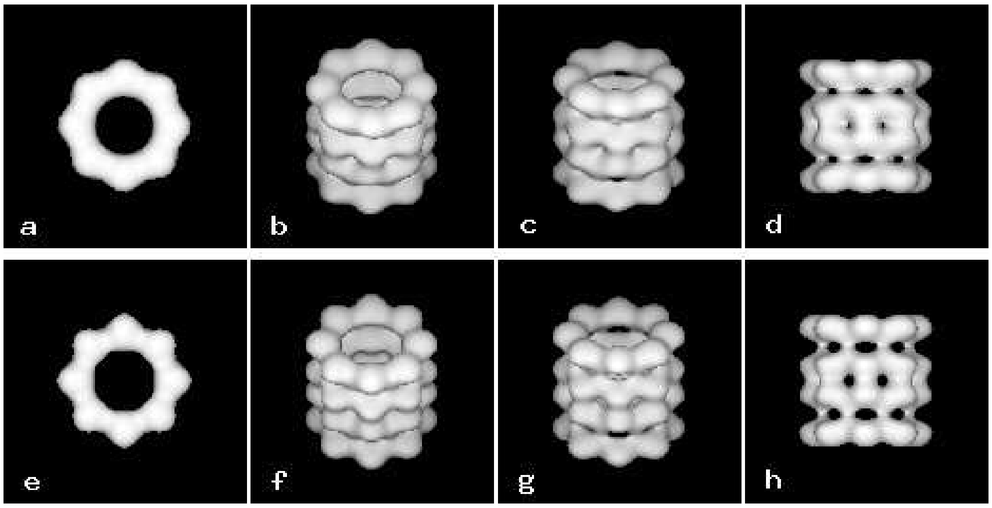


Figura 7.11: Varias vistas de superficie de la reconstrucción tridimensional final en a-h. Las vistas a-d corresponden al fantasma NCCT filtrado con un filtro pasa baja a los radios 8-12 voxels de Fourier, con un perfil coseno alzado. La frecuencia de corte es a 10 voxels de Fourier que correspondería, si suponemos que cada voxel es de 4.7 Åpor lado, a la resolución de 1/30 $Å^{-1}$. La reconstrucción final también fue filtrada de la misma manera.

Capítulo 8

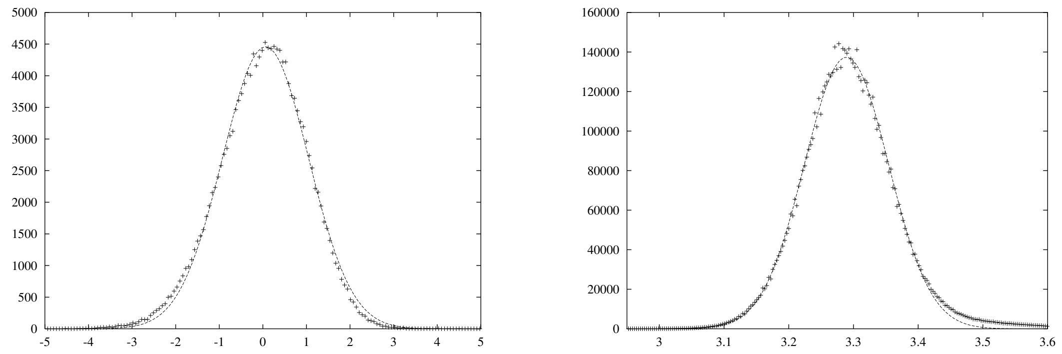
Discusión

8.1 Sobre el Modelo de Ruido

Se ha usado, en todas las simulaciones de esta memoria, un modelo simple de ruido: aditivo, independiente de la señal, con una distribución Gausiana, de media cero y desviación estándar σ. Este es el modelo usado por Farrow y Ottensmeyer [41, 42] en sus trabajos para simular imágenes de proyección tomadas en el modo campo obscuro del Microscopio Electrónico de Transmisión (MET). Estos autores reconocen que el modelo simple de ruido Gausiano introducido en las imágenes de proyección es sólo una aproximación de todos los procesos que destruyen la señal en las micrografías, y que no considera variaciones más complicadas en la estructura de interés que pueden ocurrir como resultado del daño del espécimen debido al haz de electrones del MET.

Penczek *et al* [83, 84] y Radermacher [91] también usan el modelo simple de ruido aditivo Gausiano en sus simulaciones. Penzeck *et al* mencionan que sus simulaciones no reflejan enteramente las características de los datos de microscopía electrónica, ya que en realidad existe un fuerte componente de ruido que puede clasificarse como "sal y pimienta", esto es, imperfecciones de varias clases, motas de polvo, residuos de proteína, etc. Aún más, en el caso de la reconstrucción tridimensional de partículas aisladas, por el esquema cónico aleatorio o por reconstitución angular, el conjunto de proyecciones no corresponden realmente al mismo objeto: las proyecciones provienen de diferentes partículas, mientras que solamente de supone que estas partículas son idénticas.

De hecho, el modelo de ruido usado en esta memoria —el modelo simple Gausiano— simula únicamente el componente de ruido debido al efecto de irradiar con un número finito de electrones la muestra y el ser recogidos éstos en una película fotográfica [101]. Esta es la fuente de ruido más importante bajo condiciones de alta resolución: especímenes finos (con un espesor menor de 100 nm) y dosis mínimas de electrones ($< 10\ e^{-}/\text{Å}^{2}$), principalmente. Bajo estas condiciones, y siguiendo a los autores citados en el párrafo anterior, el componente del ruido puede ser considerado como aditivo, independiente de la señal y con una distribución Gausiana. Así, el modelo básico de formación de imagen usado en este estudio fue $g = f + n$, donde g es la



(a) (b)

Figura 8.1: Histogramas de imágenes digitalizadas. En (a) el histograma de una micrografía digitalizada a tamaño 3000×2000 pixels, procedente de una muestra de tinción negativa. En (b) el histograma de 34 imágenes de tamaño 64×64 pixels extraídas del "fondo" de una micrografía tomada de una muestra preparada por criomicroscopía. En (a) y (b) se ha sobrepuesto una curva Gausiana cuyos parámetros fueron obtenidos por un ajuste de mínimos cuadrados.

imagen grabada, f el espécimen "perfecto", y n el ruido.

Para apoyar experimentalmente nuestra asunción del modelo del ruido se presenta en la Fig. 8.1(a) el histograma de una micrografía digitalizada, a un tamaño de 3000×2000 pixels, a partir de una preparación de tinción negativa. En la Fig. 8.1(b) se presenta el histograma acumulativo de 34 imágenes de tamaño 64×64 pixels del "fondo" de una micrografía digitalizada de una preparación criocongelada. En ambas figuras de ha sobrepuesto el modelo de una Gausiana, obtenida por un ajuste por mínimos cuadrados al histograma respectivo. Como puede observase en las Fig. 8.1, nuestra asunción del modelo se ajusta muy bien a nuestros datos experimentales.

8.2 El Escalamiento en la Densidad

Una consecuencia de nuestras simulaciones es que se tiene las proyecciones calculadas directamente a partir de un modelo tridimensional de densidad uno, y nuestras proyecciones presentan, por lo tanto, valores mayores o iguales a cero en los valores de sus pixels. A estas proyecciones se les suma un ruido Gausiano de media cero, acorde a nuestro modelo de ruido ya explicado arriba en la sección anterior.

Sin embargo, en una aplicación real no podemos asumir que, de entrada, el valor esperado del ruido que contienen sea cero. De hecho, es de vital importancia introducir una etapa de corrección del escalamiento en las densidades de las proyecciones que prevenga analizarlas con pesos indebidos en cada proyección, lo que causaría una representación distorsionada de la estructura bajo estudio [50, p. 216].

Boisset *et al* [5] realizan un procedimiento de escalamiento que hace uso explícito de la distribución estadística del ruido de fondo. Este procedimiento es el que sugerimos se debe realizar en el caso de tener un espécimen que muestre varias vistas preferentes (en el caso de tener una sola vista preferente basta con normalizar todas las imágenes a una misma media y desviación estándar dada). El procedimiento que realizan Boisset *et al* es el siguiente: se asume que el ruido de fondo que rodea a cada partícula tiene la misma distribución estadística. (1) Se usa una porción grande de una de las micrografías inclinadas para generar un histograma de referencia. (2) Se realiza otro histograma de las densidades del área alrededor de la partícula (de manera semejante al área que representa el parámetro l en la Fig. 6.1). (3) El histograma generado en (2) es comparado con el de referencia en (1), y los parámetros a y b de una transformación lineal de densidades

$$D'_i = aD_i + b \tag{8.1}$$

son estimados. Y, (4) esta transformación de la Ec. 8.1 es aplicada a la todos los pixels de la imagen de la partícula, con lo que resulta que, al final de aplicar este procedimiento, todas las imágenes con las que se realizará la reconstrucción tridimensional tendrán idéntica estadística del ruido. De hecho, esta normalización resulta en imágenes cuyas estadísticas encajan más allá del primer y segundo momento (su media y desviación estándar).

8.3 Sobre la Digitalización y el Teorema del Muestreo

Acorde al teorema de Whittaker-Shannon, una función continua limitada en banda puede ser representada por una serie de medidas discretas (o "muestras") tomadas a intervalos regulares. Puede demostrarse que la función original puede ser enteramente reconstruida a partir de sus muestras [80]. Una imagen, representada como una función bidimensional, requiere muestrearse con un paso de tamaño $1/(2B)$, siendo B el límite en banda en el espacio de Fourier. En la práctica, la resolución que se puede obtener a partir de los especímenes biológicos aislados raramente excede los $1/20$ Å^{-1}, de manera que un paso en el muestreo de 10 Å en la escala del objeto podría satisfacer la estipulación del teorema de muestreo. Sin embargo, tal "muestreo crítico" no toma en cuenta etapas subsecuentes que deterioran la resolución. Deterioro que es inevitable debido a que la imagen digitalizada tiene que ser rotada y desplazada, y cada una de estas transformaciones geométricas involucra alguna interpolación y remuestreo.

La interpolación es equivalente a un promedio ponderado sobre pixels vecinos en la imagen digitalizada. Por ejemplo, la interpolación bilinear, usada para rotar y desplazar la imagen digitalizada, involucra cuatro pixels y se calcula como:

$$X = a * (1 - x) * (1 - y) + b * x * (1 - y) + c * (1 - x) * y + d * x * y \tag{8.2}$$

donde X es el valor interpolado, a, b, c y d son los valores de los pixels que colindan en el nuevo punto, y (x, y) son los valores fraccionales de sus coordenadas, a $0 \leq x \leq 1$ y $0 \leq y \leq 1$.

En cada uno de los métodos de refinamiento de la geometría tratados en esta memoria, se utilizan interpolaciones y remuestreo. En el método de refinamiento por comparación directa de las proyecciones (Cap. 4) cada imagen se remuestrea en coordenadas polares usando una interpolación bicuadrática, que involucra a 5 pixels vecinos. Para el método de refinamiento por transformadas de Radon (Cap. 5), la misma transformada bidimensional de Radon es un remuestreo de la imagen que involucra rotaciones y sumas en cada línea de la imagen (en el eje x). Los métodos no basados en prototipo (Cap. 6, secciones 6.2 y 6.3) utilizan una interpolación implícita en la contribución de cada pixel en el cálculo de cada integral de línea sobre la imagen (sin rotarla).

De esta forma, y en orden para mantener en mínimo las pérdidas subsecuentes de resolución debidas a las interpolaciones y remuestreos, es costumbre usar un tamaño de paso cercano a 5 Å, que resulta en un sobremuestreo por un factor de 2 relativo a la resolución de $1/20$ Å^{-1}, que como ya se mencionó arriba, es el límite de resolución normal para especímenes biológicos aislados.

La pérdida de resolución es aún mayor cuando se realiza la reconstrucción tridimensional, debido a la imprecisión en la alineación de las distintas vistas, así como por el hecho de que no contar normalmente con imágenes que representen un muestreo uniforme de todas las direcciones del espacio tridimensional.

En nuestro experimento del refinamiento y reconstrucción tridimensional del Cap. 7, nuestra resolución alcanzada es de 10 pixels de Fourier (correspondiente a $1/30$ Å^{-1} si suponemos un tamaño de 4.7 Å en el intervalo de muestreo). Aquí el límite de resolución nos lo dio el modelo inicial de partida: las vistas frontal y lateral, con las que se construyó el modelo sintético, presentaron una resolución de 14 y 11 pixels de Fourier, respectivamente. Las proyecciones fueron filtradas a 10 pixels de Fourier para alinearlas con este modelo inicial, ya que abajo de la frecuencia correspondiente a 10 pixels de Fourier tanto como las proyecciones como el modelo 3D comparten la misma información. Las reconstrucciones tridimensionales realizadas subsecuentemente se realizaron con las proyecciones sin filtrar y fueron las reconstrucciónes las que se filtraron a 10 pixels de Fourier.

8.4 Sobre el Modelo de Programación Paralela

Antes de discutir sobre la utilización del modelo de programación paralela utilizado introduciremos algunos conceptos, para situarnos en un mismo marco de referencia y posteriormente apoyarnos en ellos.

8.4.1 Modelos de arquitectura

Diferentes arquitecturas paralelas pueden requerir diferentes algoritmos para resolver el mismo problema eficientemente. ¿Es posible evitar el diseño de un nuevo algoritmo para cada arquitectura? Esto podría ser posible si hubiera un modelo abstracto universal para las arquitectura paralelas. Los programas paralelos diseñados para este modelo universal podrían ser portados a ordenadores específicos.

El modelo de von Neumann es un modelo computacional universal para ordenadores secuenciales, que posibilita el dineño de programas secuenciales portables. Aunque difieran la arquitectura del procesador y la organización de la memoria entre los ordenadores secuenciales, todos ellos conforman el mismo modelo abstracto. Consecuentemente, un programa diseñado para el modelo von Neumann puede compilarse y ejecutarse con relativa eficiencia sobre cualquier ordenador secuencial. Mas aún, la existencia de un modelo universal ha jugado un papel en el crecimiento impresionante de las aplicaciones elaboradas para máquinas secuenciales. Si los algoritmos tuviesen que ser rediseñados para cada nueva arquitectura secuencial, el costo del desarrollo del software sería mucho más alto, limitando grandemente el coste efectivo de las aplicaciones para estos ordenadores.

Un modelo universal para los ordenadores paralelos debe exhibir dos características. Primero, debe ser lo suficientemente general como para englobar todas las principales propiedades de una gran clase de máquinas paralelas. Segundo, los programas diseñados para este modelo abstracto deben ejecutarse eficientemente sobre las actuales máquinas paralelas. Estás características representan metas conflictivas. Por una parte, un modelo universal de ordenadores paralelos no debería hacer la suposición sobre el grado de conectividad entre los procesadores que la conforman. Por otra parte, es un hecho que muchas estrategias de diseño de algoritmos paralelos sí que hacen ciertas suposiciones sobre este grado de conectividad. Redes tales como los arreglos lineales o las mallas tienen un grado relativamente bajo de conectividad, los algoritmos diseñados para estas redes serán óptimos para máquinas con estas características. No obstante, los mismo algoritmos pueden ser subóptimos para arquitecturas con un alto grado de conectividad, debido a que los recursos de comunicaciones serán subutilizados. Por el otro lado, si el modelo asume un grado alto de conectividad entre los procesadores, entonces los algoritmos diseñados para el modelo pueden ser subóptimos, por ejemplo, sobre una arquitectura como las mallas, y óptimo, por ejemplo, sobre un hipercubo. Hasta ahora la concepción de un modelo universal de ordenador paralelo, junto con las herramientas necesarias para proyectar ese modelo en cualquier tipo de ordenador paralelo concreto de una forma eficaz, es aún un tema abierto de investigación.

8.4.2 Paradigmas de programación paralela

Diferentes lenguajes de programación paralela imponen diferentes paradigmas de programación. Las variaciones entre paradigmas están motivados por factores diversos. Primero, hay una diferencia en la cantidad de esfuerzo invertido en la escritura de programas paralelos. Algunos lenguajes requieren más trabajo del programador, mientras otros que requieren menos trabajo resultan en un código menos eficiente. Segundo, un paradigma de programación puede ser más eficiente que otros para programar sobre ciertas arquitecturas de ordenadores. Tercero, varias aplicaciones tienen diferentes tipos de paralelismo, así, diferentes lenguajes de programación han sido desarrollados para explotarlos. Aquí se discutirá en gran detalle estos factores. La mayor parte de esta sección ha sido extraída de la ideas de Kumar *et al* [67].

Programación paralela explícita versus implícita

Un camino para desarrollar un programa paralelo es introducir explícitamente el paralelismo. Esta aproximación, llamada *programación paralela explícita*, requiere un algoritmo paralelo que especifique explícitamente que procesadores cooperarán y cómo en la resolución de un problema específico. La tarea del compilador es directa. Este genera código para las instrucciones especificadas por el programador. La tarea del programador, no obstante, puede ser difícil.

Otro camino para desarrollar un programa paralelo es usar un lenguaje de programación secuencial y tener un compilador que inserte los constructores necesarios para ejecutarlo sobre el ordenador paralelo. Esta aproximación, llamada *programación paralela implícita*, es relativamente fácil de usar por parte del programador porque delega la mayoría de la esfuerzo de la paralelización al compilador.

Desafortunadamente, la conversión automática de un programa secuencial a un programa paralelo eficiente es muy difícil porque el compilador debe analizar y entender las dependencias entre diferentes partes del código secuencial para asegurar un mapeo eficiente sobre el ordenador paralelo. El compilador debe particionar el programa secuencial en bloques y analizar las dependencias entre bloques. Los bloques entonces son convertidos a tareas independientes que son ejecutadas sobre distintos procesadores. El análisis de las dependencias es complicado por estructuras de control tales como ciclos, bifurcaciones, y llamadas a procedimientos. Más aún, frecuentemente hay varios caminos para escribir un programa secuencial para una aplicación dada. Algunos programas secuenciales son más fáciles que otros para que el compilador genere un código paralelo eficiente. Por lo tanto, el éxito de la paralelización automática también depende de la estructura del código secuencial.

Espacio de direcciones compartido versus paso de mensajes

En el paradigma de programación del espacio de direcciones compartido (shared-address-space), el programador ve su programa como una colección de procesos accesando un grupo central de variables compartidas. El estilo de programación del espacio de direcciones compartido es conveniente, naturalmente, para ordenadores SASA (del inglés Shared Address Space Architecture). Un programa paralelo sobre un ordenador SASA comparte los datos almacenándolos en la memoria globalmente accesible. Cada procesador accesa los datos compartidos por medio de escrituras y lecturas a las variables compartidas. No obstante, más de un procesador podría accesar la misma variable compartida al mismo tiempo, originando resultados impredecibles e inestables. Un lenguaje de programación que soporte este paradigma por el espacio de direcciones compartido debe proveer las primitivas necesarias para resolver los problemas de exclusión mutua.

En el paradigma de programación *por paso de mensajes*, el programador ve sus programas como una colección de procesos con variables locales privadas y con la habilidad de enviar y recibir datos entre procesos mediante paso de mensajes. En este paradigma, no hay variables compartidas entre los procesos. Cada proceso tiene sus variables locales, y ocasionalmente envía o recibe datos desde otros procesos. El estilo de programación por paso de mensajes es adecuado, naturalmente, para ordenadores de paso de mensajes.

Los ordenadores con el espacio de direcciones compartido pueden programarse también usando el paradigma por paso de mensajes. Ya que la mayoría de los ordenadores con el espacio de direcciones compartido tienen arquitecturas no uniformes de acceso a memoria, la emulación de paso de mensajes explota mejor los datos localmente y resulta en una mejora del rendimiento para muchas aplicaciones.

Actualmente también existe el paradigma de *memoria distribuida compartida* [87] que intenta sacar partido de los esquemas de espacio de direcciones compartido y por paso de mensajes.

Paralelismo en datos versus paralelismo en control

En algunos problemas, muchos elementos de datos están sujetos a idéntico procesamiento. Tales problemas pueden paralelizarse asignando elementos de datos a varios procesadores, cada uno de los cuales realiza cálculos idénticos sobre sus propios datos. Este tipo de paralelismo es llamado *paralelismo en datos*. Un ejemplo que exhibe paralelismo en datos es la multiplicación de matrices. Cuando se multiplican dos matrices A y B, de tamaño $n \times n$, para obtener la matriz $C = c_{i,j}$, cada elemento $c_{i,j}$ es calculado realizando el producto escalar de la i-ésima fila de A con la j-ésima columna de B. Por lo tanto, cada elemento $c_{i,j}$ es calculado realizando idénticas operaciones sobre diferentes datos, lo cual representa paralelismo en datos.

Varios lenguajes de programación han sido desarrollados para hacer fácil la explotación del paralelismo en datos. Tales lenguajes son llamados *lenguajes de programación de datos-paralelos* y los programas escritos en tales lenguajes son llamados *programas de datos-paralelos*. Un programa de datos-paralelos contiene una secuencia simple de instrucciones, cada una de los cuales es aplicada a los elementos de datos en pasos por bloques. Los programas de datos-paralelos son apropiados para ordenadores SIMD (del inglés Single Instruction stream, Multiple Data stream) tal como una MasPar MP-1 o MasPar MP-2 (en estás máquinas una sola unidad de control despacha las instrucciones a cada unidad de procesamiento). Una unidad de control central irradia las instrucciones a los procesadores, los cuales contienen los datos. Los procesadores ejecutan el flujo de instrucciones sincrónicamente. Los programas de datos-paralelos pueden también ejecutarse en ordenadores MIMD (del inglés Multiple Instruction stream, Multiple Data stream, estas máquinas son capaces de ejecutar un programa diferente e independiente en cada procesador). Sin embargo, la ejecución síncrona estricta de un programa de datos-paralelos sobre un ordenador MIMD resulta en un código ineficiente ya que este requiere una sincronización global después de cada bloque de instrucciones. Una solución a este problema es relajar la ejecución síncrona de instrucciones. En este modelo de programación, llamado de *programa único y múltiples datos* o SPMD (por sus siglas en inglés), cada procesador ejecuta el mismo programa de manera asíncrona. La sincronización toma lugar cuando los procesadores necesitan intercambiar datos. Por lo tanto, el paralelismo en datos puede ser explotado sobre un ordenador MIMD aún sin un lenguaje de programación explícito para datos-paralelos.

El *paralelismo en control* se refiere a la ejecución simultánea de diferentes flujos de instrucciones, aunque típicamente se aplica a diferentes flujos de datos. Un ejemplo de paralelismo en control es la segmentación (*pipelining*, en inglés). Mediante la segmentación, el cálculo es paralelizado por medio de la ejecución de un programa diferente en cada procesador y enviando resultados intermedios al siguiente procesador. El resultado es un flujo de datos entre los procesadores. Los algoritmos para problemas que requieren paralelismo en control usualmente se mapean bien sobre ordenadores paralelos MIMD porque el control del paralelismo requiere múltiples flujos de instrucciones. En contraste, los ordenadores SIMD soportan solamente un flujo de instrucciones y no son capaces de explotar el paralelismo en control eficientemente.

Muchos problemas exhiben una cierta cantidad de ambos esquemas, de paralelismo en datos y paralelismo en control. La cantidad de paralelismo en control disponible en un problema es usualmente independiente del tamaño del problema y es, de esta manera, limitado. En contraste, la cantidad de paralelismo en datos, en un problema dado, se incrementa con el tamaño del problema. Por lo tanto, acorde al uso eficiente de un número grande de procesadores, es necesario explotar el paralelismo en datos inherente a una aplicación.

8.4.3 Nuestro paradigma: Paso de mensajes

En la programación de los métodos de refinamiento de la geometría tridimensional usando un prototipo, de los capítulos 3-5 de esta memoria, se utilizó el paradigma de paso de mensajes.

Nuestra plataforma de desarrollo fue una máquina Power Challenge serie L, de Silicon Graphics, con 8 procesadores R10000. Es una máquina multiprocesadora, esto es, una máquina SASA. Entonces, tenemos una plataforma de memoria compartida donde lo más obvio sería utilizar el paradigma de espacio de direcciones compartido. Sin embargo, una máquina SASA como nuestra Power Challenge puede programarse también usando el paradigma por paso de mensajes [87, 67]. Ya se ha probado la viabilidad de esta aproximación. En concreto, Shekhar *et al* 1996 [107], construyeron un Sistema Geográfico de Información (GIS, en sus siglas en inglés) de alto rendimiento, paralelizándolo usando el paradigma de paso de mensajes sobre una máquina SASA, una Challenge también de SGI. Los autores elaboraron su propia librería para el paso de mensajes aprovechando operaciones SASA como read, write, lock y unlock para emularlos.

SGI provee con nuestra Power Challenge las librerías de PVM/MPI [111, 18, 47, 67, 32] que son librerías de uso público para la creación de programas paralelos usando el paradigma de paso de mensajes. La versión de PVM que se usó en nuestra aplicación es la elaborada por SGI por lo que hace uso de las características de memoria compartida para sacar el máximo beneficio de la arquitectura de la máquina Power Challenge mediante paso de mensajes.

8.4.4 Nuestra aplicación con el paradigma de memoria compartida

¿Es la mejor aproximación nuestra realización con el paradigma de paso de mensajes? ¿O es —simplemente— "aceptable"? De nuestros resultados con nuestros algoritmos de refinamiento de la geometría lo confirmamos parcialmente. Y para verificarlo, repetimos el desarrollo con una programación implícita bajo el paradigma de memoria compartida. Así, con el apoyo del Departamento de Arquitectura de Computadores de la Universidad de Málaga, se modificó el programa secuencial que realiza el refinamiento de la geometría por comparación directa de las proyecciones con directivas que explotan el paradigma del espacio de direcciones compartido. El nuevo programa fue realizado en su máquina Origin 2000 de SGI, una máquina SASA.

SGI tiene los precompiladores Power C, PCA (Power C Analizer), y Power FORTRAN, PFA (Power FORTRAN Acelerator) para la realización de programas bajo el paradigma del espacio de direcciones compartido. Estos precompiladores insertan en el código fuente secuencial directivas especiales, generando un nuevo código fuente modificado, que habilita la ejecución eficiente del programa bajo las máquinas multiprocesadoras de SGI.

El nuevo código fuente producido por el PCA o PFA es entendido por el compilador normal de C o FORTRAN. Estos últimos producen el programa ejecutable que puede ejecutarse concurrentemente.

En el paradigma del espacio de direcciones compartido de SGI se usa el modelo de *hebras*. Los procesos que participan en la ejecución paralela se disponen en una organización maestro/hijo (traducción del inglés *master/slave*). El proceso original es el maestro. Este crea cero o más hijos que le asistan. Cuando es encontrado un ciclo que se ha paralelizado, el maestro manda una petición de ayuda a los hijos. Cuando el ciclo es completado, los hijos esperan al maestro y entonces el maestro reanuda su ejecución normal. El proceso maestro y cada uno de los procesos hijos son llamados una *hebra de ejecución* o simplemente una *hebra*. Por defecto, el número de hebras es igual al número de procesadores sobre una máquina en particular [108, Cap. 7].

El método de refinamiento de la geometría basado en la comparación directa de las proyecciones fue programado en el lenguaje de programación C. El resultado de

Proc. Hijos	Tiempo (min,seg)	Aceleración
8	2'51"	6.1
4	5'03"	3.4
2	8'44"	2.0
1	17'22"	1.0

Tabla 8.1: Tiempos de ejecución y la aceleración, según el número de procesos hijos del programa paralelo en PVM, para el método por comparación directa de las proyecciones. Los tiempos fueron medidos en la máquina SASA Origin 2000 de SGI.

aplicar el precompilador PCA, en la máquina Origin 2000, sobre el código fuente original, fue muy malo. El precompilador automático concentraba la paralelización en lazos muy internos, con lo cual la mayor parte del tiempo ejecuta una sola hebra y solo en pequeños (ridículos) trozos se ejecutaban todas las hebras. Además, el coste de lanzar las hebras y sincronizarlas en su finalización, más las esperas activas por conseguir la CPU (parece que cuenta el tiempo en cola de procesos preparados en competencia con otros procesos) elevaba el tiempo de ejecución de los 15 min, del programa secuencial original, a más de una hora. Resumiendo, el precompilador PCA no encontraba nada fácilmente paralelizable a pesar de la regularidad del código. Esto se debe a que, cuando aparece una función cualquiera en el código, PCA o equivalentes asumen que hay dependencias de datos y no paralelizan.

Un análisis del código indicaba que era muy paralelo y regular (el tiempo de ejecución de realizar una proyección o correlacionar un par de proyecciones es siempre el mismo) y que se esperaban resultados muy similares a los obtenidos con la realización por paso de mensajes usando PVM, esto apoyándose en la portabilidad y la disponibilidad de PVM/MPI en máquinas SASA.

El método de refinamiento de la geometría basado en la comparación directa de las proyecciones posee dos tareas muy costosas y claramente paralelizables: (1) el cálculo de todas las reproyecciones de un volumen, (2) el cálculo de la correlación cruzada de cada imagen experimental con todas las reproyecciones eligiendo la más similar. Los tiempos de ejecución de la versión paralela en PVM, en la máquina SASA Origin 2000, para alinear 10 imágenes experimentales con respecto a 5266 reproyecciones, pueden verse en la Tabla 8.1. La línea de comandos fue:

```
smaster3d 8 0 0 0.0 0.0 2.0 volumen.img prj.sel 2 30 1
```

La aceleración en la Tabla 8.1 no es la ideal puesto que las medidas has sido tomadas en un entorno real con la máquina muy cargada, sin posibilidad de medir tiempos reales de CPU consumidos por la aplicación. Sin embargo, las medidas anteriores también se realizaron con el alineamiento de solo una imagen, obteniéndose aceleraciones perfectas. De todo esto lo que se puede deducir es que en condiciones favorables de disponibilidad de máquina, el resultado seria muy próximo al ideal.

La versión del programa del método de comparación directa de las proyecciones en memoria compartida se paralelizó usando las directivas de SGI del modelo de hebras. La paralelización se realizó "a mano", dados los pésimos resultados ob-

Proc. Hijos	Tiempo (min,seg)	Aceleración
8	0’48”	7.79
4	1’37”	3.86
2	3’10”	1.97
1	6’14”	1.00

Tabla 8.2: Tiempos de ejecución y la aceleración, según el número de hebras, para el método por comparación directa de las proyecciones. Los tiempo fueron medidos en la máquina SASA Origin 2000 de SGI.

tenidos con el PCA. De esta forma se identificaron los dos bucles que realizan el mayor consumo de tiempo de calculo: 1) el bucle que realiza las reproyecciones y 2) el bucle que posteriormente compara cada imagen experimental con todas las reproyecciones. La paralelización ha sido directa. El código secuencial se tuvo que transformar para poder usar la directiva *pfor* que ejecuta un bucle en paralelo, también se cambió una estructura de datos que almacenaba los ángulos a los que se realizan las reproyecciones en una matriz para poder iterar sobre ella con un simple bucle *for*. Las complicaciones venían de la necesidad de estudiar en detalle el uso de las variables en el programa, puesto que en la versión de memoria compartida hay que elegir cuidadosamente que variables son compartidas y cuales serán locales a cada hebra de ejecución. En el bucle que realiza el alineamiento y posteriormente encuentra el mejor, se ha paralelizado tanto los alineamientos como el cálculo del máximo (mediante una reducción, usando secciones criticas) Tras esto los tiempos que se han medido son comparables a la versión en PVM (6.7 —sobre 8— frente al 6.1 de PVM) para un nivel de carga similar (muy cargado) al que se registraba cuando se tomarón los tiempos de la versión en PVM. De nuevo, los tiempos no son ideales (son peores que los que corresponderían a un uso exclusivo de la máquina).

Durante un corto espacio de tiempo en el que la máquina estaba poco cargada se han tomado tiempos para la versión en memoria compartida obteniéndose los resultados de la Tabla 8.2. De estos resultados tenemos que suponer que, con la misma disponibilidad de recursos en la maquina, la versión en PVM subiría su rendimiento hasta alcanzar a la versión en memoria compartida. Creemos que para un problema tan regular los resultados deberían ser muy igualados. La versión de PVM que se esta usando es la realizada por SGI (tanto en la Power Challenge como en la Origin), por lo que hace uso de las características de memoria compartida para sacar el máximo beneficio de la arquitectura de las máquinas al usar el paso de mensajes. Con el uso de las directivas especificas de memoria compartida conseguimos mayor control con menos esfuerzo, es fácil de paralelizar añadiendo poco código y, por ejemplo, la planificación del trabajo para mejorar el balanceo (si fuese necesario) se puede realizar automáticamente con sólo cambiar un parámetro en las directivas.

8.5 Sobre el Tiempo de Ejecución de los Programas de Refinamiento de la Geometría Basados en Prototipo

Vamos a realizar aquí un análisis de el por qué el tiempo de ejecución del método de refinamiento basado en transformadas de Radon sobrepasa en gran medida el tiempo de ejecución del método de refinamiento basado en la comparación directa de las proyecciones. El análisis se basará en el número de Transformadas Discretas de Fourier (TDF) que se utiliza en la comparación de dos imágenes para hallar el tercer ángulo de Euler, esto es, la rotación "en el plano" relativa entre ellas.

El método basado en la comparación directa de las proyecciones, en su segunda parte de la búsqueda de la orientación, utiliza una FFT inversa potencia de 2 [1] para el cálculo del coeficiente de correlación cruzada entre las dos proyecciones: las dos proyecciones son remuestreadas a coordenadas polares, a un número de muestras potencia de 2 por anillo; se calcula la TDF de los anillos para cada imagen y se multiplican los coeficientes correspondientes, entonces se realiza la FFT inversa. La descripción precisa del desarrollo matemático puede verse en la referencia [83]. El ángulo de rotación entre las imágenes viene expresado por la posición del máximo en la correspondiente función de correlación cruzada radial (ver la sec. 4.2).

El método basado en las transformadas de Radon esta formulado en la correlación cruzada de las transformadas bidimensionales de Radon [2] de las dos proyecciones. Sean $\hat{f}(p,\epsilon)$ la transformada 2D de Radon de la proyección de referencia f y $\hat{g}(p,\epsilon)$ la transformada 2D de Radon de la proyección experimental g con su ángulo α (la rotación en el plano) desconocida. De forma para hallar el ángulo α, la proyección f se rota a incrementos en el ángulo α y se correlaciona con la proyección experimental. El coeficiente de correlación cruzada más grande nos da el correspondiente valor de α. Usando transformadas de Radon es fácil el cálculo de la rotación de la imagen f, explícitamente no es necesario realizar la rotación de f y calcular nuevamente su TR, la mismas propiedades de la TR nos dan la TR de la imagen rotada. El cálculo de $\hat{f}(p,\epsilon+\alpha)$, correspondiente a la TR de la proyección rotada por un ángulo α, se obtiene por permutaciones e invirtiendo las líneas de la TR original $\hat{f}(p,\epsilon)$. Si $l(\epsilon)$ es una línea en la TR, entonces $l(\epsilon+\pi) = \mathrm{mirror}(l(\epsilon))$.

El cálculo de la correlación cruzada entre las TR's, C_{α_i}, se realiza de la siguiente manera. Se calcula la correlación cruzada normalizada entre cada par de líneas de

[1] Un algoritmo que realiza la Transformada Discreta de Fourier sobre una secuencia de tamaño N, donde N es el producto de dos o más enteros, se conoce como Transformada Rápida de Fourier (FFT, por sus siglas en inglés) [80]

[2] Aquí, estrictamente hablando, debería referirse a la Transformada Discreta de Radon (TDR). La transformada de Radon esta definida en el continuo, sin embargo aquí se usa como sinónimo de la TDR, por simplificación.

las TR's, $c_{\alpha_i}(p', \epsilon)$

$$c_{\alpha_i}(p', \epsilon) = \frac{1}{\sigma_f \sigma_g} \int \hat{g}(p, \epsilon) \hat{f}(p + p', \epsilon + \alpha_i) \, \mathrm{d}p \tag{8.3}$$

donde σ_f y σ_g son la desviación estándar de $\hat{f}$ y $\hat{g}$, respectivamente. Y se promedian todas las c_{α_i} en $p' = 0$

$$C_{\alpha_i} = \frac{1}{N} \sum_{l=1}^{N} c_{\alpha_i}(0, \epsilon) \tag{8.4}$$

Para el cálculo de la Ec. 8.3 primeramente se realiza la TDF de las TR's. Seguidamente se multiplican las respectivas líneas de las TDF de las RT's y se calculan las TDF's inversas (una TDF inversa por cada línea). Las propiedades de la TR se mantienen para la TDF de una TR. La $\hat{F}(p, \epsilon + \alpha)$, para un nuevo cálculo de C_{α_i} a otro valor de α_i distinto, se obtiene por permutaciones e inversiones de la $\hat{F}(p, \epsilon)$ original.

Por lo tanto, el número de FFT's inversas necesarias para calcular la correlación cruzada entre la TDF de dos TR's, es igual a N líneas de las RT (Ec. 8.4) y que es igual a la amplitud de ϵ con los que se calcularon las TR's. El número de FFT's inversas para hallar el ángulo de rotación α es igual a N multiplicado por el número de valores distintos α_i en los que se realiza la búsqueda (teniendo ya calculadas la TDF de las TR's).

Resumiendo, el número de FFT's inversas necesarias para hallar el ángulo de rotación entre dos proyecciones es:

1. Para *método por comparación directa de las proyecciones*

   1 (una), para $0° \leq \alpha < 360°$.

   ignorando las FFT's necesarias para calcular la TDF de los anillos de las imágenes remuestreadas en coordenadas polares.

2. Para el *método por transformadas de Radon*

   $$N * \left(\frac{\alpha_{\text{final}} - \alpha_{\text{inicial}}}{\Delta\alpha} + 1 \right) \tag{8.5}$$

   donde

   $$N = \frac{\epsilon_{\text{final}} - \epsilon_{\text{inicial}}}{\Delta\epsilon} + 1 \tag{8.6}$$

   ignorando las FFT's necesarias para obtener las TDF de las TR de ambas proyecciones.

FFT potencia 2		FFT multiradix		
Tamaño	Tiempo	Tamaño	Tiempo	Relación de tiempos
64	11.56	46	35.28	1/3.0
128	24.00	92	56.88	1/2.4
256	50.27	138	108.60	1/2.2

Tabla 8.3: Tiempo de CPU, en microsegundos, para realizar una FFT potencia de 2 y multiradix para tres distintos tamaños de secuencia. Los tiempos se midieron sobre una máquina Power Challenge de SGI

Ahora, el cálculo de la TDF en el segundo método se realiza por el algoritmo multiradix por descomposición en números primos de Singleton [109], donde el tamaño de las secuencias no tiene que ser necesariamente un número potencia de 2 (aunque hay números "prohibidos" donde el algoritmo no funciona). Una medición de los tiempos de CPU requeridos para realizar la FFT potencia de 2 y multiradix, para tres diferentes tamaños de secuencia, se presentan en la Tabla 8.3.

Observando los datos de la Tabla 8.3 se observa que la FFT multiradix tarda 3 veces más tiempo que una FFT potencia de 2, para un tamaño de secuencia de 46 y 64 muestras, respectivamente. Con los datos de la realización de los experimentos del capítulo 7, N, en la Ec. 8.6, es igual a 90 (la TR se realiza entre $0°$ y $178°$. con un paso angular de $2°$). El número total de FFT's inversas, para el método de refinamiento por transformadas de Radon, es igual a $90 * [(44° - 0°)/2° + 1]$ (la búsqueda en α se realiza en la amplitud $0° \leq \alpha \leq 44°$, a un paso de $\Delta\alpha = 2°$) igual a 2070 FFT's inversas. Y como una FFT multiradix tarda 3 veces más tiempo que una FFT potencia de 2, la cual utiliza el método de refinamiento por comparación directa de las proyecciones, concluimos que el método de refinamiento por transformadas de Radon tarda $1980 * 3 = 6210$ veces más, considerando solamente, en este análisis, el cálculo de las transformadas de Fourier inversas.

8.6 De la Aplicación de los Métodos Basados en Prototipo

Para realizar una reconstrucción tridimensional de una macromolécula biológica a partir de imágenes de partículas individuales a una resolución mayor a 30 Å es necesario realizar la corrección de la Función de Transferencia de Contraste (CTF, por sus siglas en inglés) del microscopio electrónico de transmisión (ver sec. 2.5). Los trabajos de Zhu *et al* [130] y Stark *et al* [110] apuntan hacia la alta resolución ($<$ 10 Å) a partir de imágenes de partículas aisladas. Zhu *et al* realizan la reconstrucción del ribosoma 70S de *E. coli* a 25 Å con la corrección de la CTF a partir de pares de micrografías del espécimen tomadas en el campo obscuro y brillante a diferentes subenfoques. Stark *et al* han realizado la reconstrucción de dos estados del mismo ribosoma 70S de *E. coli* a 20 Å, estos autores realizan la corrección de la CTF en la reconstrucción tridimensional final ya que todas las micrografías fueron tomadas al mismo subenfoque. En todas estas reconstrucciones se han utilizado un refinamiento de la geometría basándose en un prototipo. Los prototipos utilizados han sido reconstrucciones previas a baja resolución, Zhu *et al* [130] han utilizado un prototipo obtenido con la aplicación del esquema cónico aleatorio y Stark *et al* han usado un prototipo construido aplicando la reconstitución angular.

Con criomicroscopía más la corrección de la CTF del microscopio electrónico, y aplicando alguna de los métodos de refinamiento de la geometría usando un prototipo, se podría alcanzar la resolución de 10 Å, o mayor, para las reconstrucciones tridimensional a partir de partículas aisladas. Todo apunta a que solo existe una limitación en el número de imágenes que son necesarias para alcanzar dicha resolución de 10 Å. Para el caso de virus icosaédricos, que presentan una gran simetría, Conway *et al* [21] han alcanzado 9 Å a partir de 600 imágenes de la cápsida del virus de la hepatitis B; y Böttcher *et al* [7] han alcanzado 7.4 Å de resolución a partir de 6 400 imágenes de la cápsida del mismo virus de la hepatitis B.

Para aumentar la resolución, por encima de los 10 Å, será necesario corregir un efecto más: el efecto de la diferencia en el aumento entre distintas micrografías. Este efecto, despreciable a resoluciones por debajo de los 10 Å aparece por la toma de varias micrografías, y es debido a que las condiciones del microscopio electrónico son difíciles de mantener entre experimento y experimento. Todos los microscopios (incluyendo el microscopio óptico) deben calibrarse para poder determinar de forma precisa el aumento total, ya que un número de variables pueden causar una variación del aumento de hasta un 20% a 30% sobre un periodo corto de tiempo [8, p. 147]. Aún en los modernos microscopios electrónicos, que tienen una pantalla digital que despliega el aumento aproximado total cuando las lentes son cargadas, sólo garantizan ser exactos en un $+/-$ 5% a 10% de los valores desplegados [8]. Esta corrección en el escalamiento (aparte de la orientación y origen, además de la corrección de la CTF) de la imagen fue realizada por Böttcher *et al* [7], para alcanzar la resolución de 7.4 Å para la reconstrucción tridimensional, por criomicroscopía, de la cápsida icosaédrica del virus de la hepatitis B. Trus *et al* [113] también aplicaron la corrección en el escalamiento para la reconstrucción de la cápsida del papilomavirus bovino, obteniendo una resolución de 9 Å.

El tener una mayor resolución de las reconstrucciones tridimensionales nos permitiría conocer a fondo la estructura de la macromolécula biológica bajo estudio: a 30 Å se conoce su forma global; la cadenas de aminoácidos comienzan a distinguirse a resoluciones de 3-4 Å y, finalmente, a 1-2 Å se pueden localizar los átomos individuales.

La eficacia en la aplicación de cualquiera de los dos métodos de refinamiento de la geometría basados en un prototipo tratados en esta memoria, 1) el método por comparación directa de las proyecciones y 2) el método basado en la transformada de Radon, por nuestra prueba realizada en la sección 7.2, resultan ambos exactos, bajo la prueba de convergencia de J. Frank [50]: el 80%, o más, de las imágenes deben moverse menos de 5° en el ángulo de inclinación. Bajo el punto de vista del tiempo de ejecución, el método 1) lleva clara ventaja, el método 2) tarda 460 veces más que el método 1). Considerando la ejecución de los programas paralelos con 5 procesos hijos, el método 1) tarda 75 seg (ver Fig. 4.2 en la pág. 47) para realizar 5266 reproyecciones y un segundo en comparar estas 5266 reproyecciones con una imagen experimental (el espacio de búsqueda aquí es el espacio tridimensional completo); el método 2) tarda 207 seg para comparar una imagen experimental con 2644 reproyecciones (o lo que es equivalante a un cuarto del espacio de búsqueda tridimensional), pero aqui las reproyecciones son extraidas de la transformada de Radon tridimensional del prototipo, además a la imagen experimental se tiene que realizar previamente su transformada de Radon y la transformada discreta de Fourier de esta transformada de Radon. Bajo el punto de vista del espacio en disco necesario, el método 2) consume más: es necesario realizar, y guardarlas en disco, las transformadas de Radon tanto de todas las proyecciones experimentales como del prototipo y también las transformadas discretas de Fourier de todas las transformadas de Radon. El consumo de espacio en disco en bytes del método 1), referido como D_{M1}, es

$$D_{M1} = 4MN^2 \tag{8.7}$$

para M imágenes experimentales de tamaño $N \times N$ y 4 bytes por cada elemento en coma flotante. El valor dado por la Ec. (8.7) representa el espacio necesario para realizar el filtraje, que puede no ser necesario, para todas las imágenes experimentales. El consumo de espacio en disco en bytes para el método 2), D_{M21}, es

$$D_{M21} = 8M \left(\frac{\epsilon_{\text{final}} - \epsilon_{\text{inicial}}}{\Delta\epsilon} + 1 \right) (S + 1) \tag{8.8}$$

donde $\epsilon_{\text{inicial}}$ y ϵ_{final} son los límites angulares a donde se realiza la transformada de Radon tomando S muestras para cada línea de la transformada. La Ec. (8.8) representa el espacio en disco necesario para almacenar las transformadas de Radon y las transformadas de Fourier de las transformadas de Radon de M imágenes experimentales y considerando 4 bytes para cada elemento en coma flotante. Además hay que añadir el espacio necesario para almacenar la transformada tridimensional de Radon y su transformada de Fourier, que es igual a:

$$D_{M22} = 8 \left(\frac{\theta_{\text{final}} - \theta_{\text{inicial}}}{\Delta\theta} + 1 \right) \left(\frac{\varphi_{\text{final}} - \psi_{\text{inicial}}}{\Delta\varphi} + 1 \right) (S + 1) \tag{8.9}$$

donde S se hace igual a la misma S de la Ec. (8.8): el número de muestras en cada línea de la transformada de Radon. $\Delta\theta$ se hace igual a $\Delta\varphi$ para la Ec. (8.9).

El espacio total en disco para el método 2), indicado como D_{M2}, es entonces igual a la suma de las ecuaciones (8.8) y (8.9):

$$D_{M2} = D_{M21} + D_{M22} \tag{8.10}$$

Si suponemos que el producto de los factores $[(\epsilon_{\text{final}} - \epsilon_{\text{inicial}})/Delta\epsilon + 1] * (S + 1)$ de la Ec. (8.8) es igual a N^2 y haciendo la misma suposición con el producto de los factores angulares y el muestreo de la Ec. (8.9) igual a N^3, la relación entre el espacio en disco del método 2) al método 1) es:

$$\frac{D_{M2}}{D_{M1}} = 2 + \frac{2N}{M} \tag{8.11}$$

el segundo término del lado derecho de la Ec. (8.11) es despreciable si $M \gg N$, condición que es fácil de tener en la práctica dado que el número de imágenes experimentales, M, es de cientos comparado con el tamaño de las mismas imágenes, $N \times N$. Así, concluyendo, el método 2) —basado en la transformada de Radon— consume como mínimo el doble de espacio en disco que el método 1) —basado en la comparación directa de las proyecciones—.

El consumo de memoria del método por comparación directa de las proyecciones es mucho mayor que el del método basado en la transformada de Radon, ya que las reproyecciones del prototipo se calculan en tiempo de ejecución y no se almacenan en disco duro. El espacio aproximado necesario para las reproyecciones, del método por comparación directa de las proyecciones, puede verse en la Tabla 4.1. En memoria, el método basado en la transformada de Radon consume la necesaria para almacenar la transformada de Fourier de la transformada de Radon del prototipo y una sola imagen.

8.7 De la Aplicación de los Métodos no Basados en Prototipo

Los métodos de refinamiento no basados en un prototipo:

1. Refinamiento de los ángulos de inclinación y de rotación (en la sec 6.2).

2. La estimación del ángulo de inclinación para partículas con simetría cilíndrica (en la sec. 6.3)

De los resultados de ambos métodos, nos indican que el nivel fraccional de ruido máximo en que nuestro proceso es operable es de 0.20. El nivel fraccional de ruido de 0.20 corresponde, según Farrow y Ottensmeyer [41] a las imágenes digitalizadas de micrografías tomadas en *campo obscuro* [60, 8] de macromoléculas individuales sin teñir. Las micrografías tomadas con el modo campo obscuro del Microscopio Electrónico de Transmisión (MET) presentan un mejor contraste, y menor ruido, que las tomadas en el modo *campo brillante* del MET, sin embargo el modo campo obscuro produce imágenes a menor resolución [101].

También ambos métodos desarrollados podrían aplicarse a imágenes procedentes del Microscopio de Transmisión con Barrido (STEM) [24] o a imágenes procedentes del Microscopio Electrónico con Filtración de Energía [85]. Ambos microscopios nos dan imágenes con una razón señal-a-ruido más alta que las imágenes obtenidas del MET en modo normal (campo brillante).

Otra aplicación posible de los métodos de refinamiento no basados en un prototipo es aplicarlos a clases, más que a imágenes aisladas. Una clase representa el promediado de un número de imágenes aisladas, y el promediado mejora sustancialmente la razón señal-a-ruido. Esta aproximación a sido recientemente usada por Beniac *et al* 1997 [3], para reconstruir la proteina *mielina* a 28 Å de resolución. Ellos obtienen 100 clases de imágenes promediadas a partir de 3117 imágenes iniciales de tinción negativa. Para alinear las imágenes originales usaron técnicas de alineamiento multi-referencia [120] y para la separación de las clases usaron el análisis de correspondencia y la clasificación jerarquica ascendente [116]; todas estas técnicas son de aplicación común para realizar la reconstitución angular (ver sec. 2.8.1) [104, 34].

Una anotación más, los métodos de refinamiento no basados en prototipo presentados en esta memoria no se paralelizaron debido a que el tiempo de ejecución y su consumo de recursos es bastante bajo (unos pocos magabytes de memoria). El tiempo de ejecución fue del orden de unos pocos minutos en una estación de trabajo SPARCstation 10, de SUN.

Capítulo 9

Conclusión

En el trabajo de esta memoria de tesis, concluyendo, se ha desarrollado lo siguiente:

1. Análisis y realización de los programas secuenciales para efectuar el refinamiento de la estructura tridimensional de macromoléculas biológicas. Los métodos considerados son los dos existentes actualmente:

   (a) El método basado en la comparación directa de las proyecciones, debido a Penczek *et al* [84].

   (b) El modelo basado en las transformadas de Radon, debido a Radermacher [91].

   Los programas actualmente están disponibles en XMIPP [73], el software de uso público desarrollado por nuestro grupo de trabajo, para el procesamiento de datos de microscopiía electrónica.

2. Se han paralelizado los programas del punto anterior bajo el paradigma de pasos de mensajes utilizando las librerias PVM [111, 67, 47]. De esta forma la aplicación paralela es fácilmente transportable a un amplio conjunto de máquinas paralelas.

3. Se ha desarrollado un nuevo procedimiento, el *método de la estimación del centroide de las proyecciones mediante segmentación de la masa* [26], para mejorar la alineación translacional de imágenes de partículas individuales. Este procedimiento también nos permite centrar primeramente las imágenes de proyección para agregarlas a una reconstrucción tridimensional en baja resolución previamente obtenida, de manera que se mejora la resolución de la reconstruccón tridimensional final.

4. Se ha desarrollado un nuevo método para la estimación de los dos ángulos de Euler involucrados en una reconstrucción en base al esquema cónico aleatorio. El método es aplicable a imágenes con un nivel fraccional de ruido hasta de 0.20, que lo hace interesante de cara a futuras aplicaciones en STEM y microscopía con filtración de energía.

5. Se ha desarrollado un nuevo método para la estimación del ángulo de inclinación en el caso de tener partículas con simetría de plegamiento cilíndrica. El nivel fraccional de ruido hasta el cual el método funciona correctamente es en el entorno de 0.20, aunque cuanto mayor sea la simetría más ruidosas pueden ser las imágenes de entrada.

6. Se ha desarrollado una nueva metodología que nos permitiría obtener una reconstruccion tridimensional sin pasar por el método de colección de datos de la geometría cónica aleatoria o la reconstitución angular:

   Se tiene un grupo de imágenes de proyección. El origen de coordenadas inicial de cada proyección se calcula aplicando el *método de la estimación del centroide mediante la segmentación de la masa.* La orientación inicial, los valores angulares, de cada proyección se calculan mediante el *método de refinamiento de la orientación* usando como prototipo un modelo construido a partir de la información de dos, o más, vistas preferentes del espécimen y considerando alguna información *a priori* como puede ser la simetría. Con los parámetros de translación y orientación iniciales conocidos para todas las proyecciones, se realiza una primera reconstrucción tridimensional del espécimen bajo estudio. A partir de aquí se puede refinar la geometría del espécimen reconstruido tomando ésta primera reconstrucción como prototipo, y usando cualquiera de los dos métodos de refinamiento de la geometría basados en in prototipo del punto 1.

Apéndice A

Optimización del Algoritmo para el Trazo de Círculos

Supongamos que tenemos una imagen de tamaño $N \times N$. Se quiere superponer sobre la imagen una máscara anular entre los radios r_1 y r_2, todos los pixels dentro del la máscara se pondrán a 1, los pixels fuera de ella se pondrán a 0. El algoritmo usado tradicionalmente [1] para aplicar alguna transformación de este tipo sobre círculos en los pixels de la imagen cuadrada es:

Función **mask** (img[N][N], r_1, r_2)

$c = (N-1)/2$ /** El centro de la imagen **/
for $r_1 \leq r \leq r_2$ **do**
    $R_1 = r - 0.5$
    $R_2 = r + 0.5$
    for $0 \leq i < N$ **do**
        for $0 \leq j < N$ **do**
            $x = c - j$
            $y = c - i$
            $d = x^2 + y^2$
            if $d \geq R_1^2$ AND $d < R_2^2$ **then**
                /** El punto i, j está en el anillo **/
                img[i][j] = 1
            endif
        endfor
    endfor
endfor

Directamente puede verse el inconveniente de esta aproximación: Para cada radio se recorren los índices i, j sobre toda la imagen y las operaciones se realizan en coma

[1] Este algoritmo se encuentra en los archivos fuente de SPIDER [51] y en los archivos fuente antiguos de XMIPP [73].

flotante. Una forma novedosa de mejorar esta función, que describimos por primera vez en ésta memoria de tesis, es cambiarla por otra incremental sobre la "cuadricula" de la imagen (cada paso sólo puede ser en un pixel) [10]. Además, se puede sacar provecho de la simetría del círculo [46].

Escribiendo la ecuación del circulo como una función de error: $F(x,y) = x^2 + y^2 - r^2$; $F(x,y)$ valdrá cero cuando los valores x, y estén sobre el circulo de radio r; $F(x,y)$ será positiva si los puntos x, y están fuera del círculo, y negativa si están dentro de él. Reescriendo la función de error como:

$$F(x,y) = e = x^2 + y^2 - (R+a)^2 \tag{A.1}$$

Resolviendo la ecuación (A.1) para el primer cuadrante, tenemos tres casos: 1) cuando $x \to x+1$, 2) cuando $y \to y-1$, y 3) cuando $x \to x+1,\ y \to y-1$. Para el primer caso tenemos:

$$\begin{aligned} F(x+1,y) = e_1 &= (x+1)^2 + y^2 - (R+a)^2 \\ &= x^2 + y^2 - (R+a)^2 + 2x + 1 \\ &= e + 2x + 1 \end{aligned} \tag{A.2}$$

Sustrayendo la Ec. (A.2) de (A.1) tenemos:

$$\Delta x = 1 + 2x \tag{A.3}$$

Resolviendo la Ec. (A.1) para los otros dos casos, llegamos a:

$$\Delta y = 1 - 2y \tag{A.4}$$

$$\Delta xy = \Delta x + \Delta y \tag{A.5}$$

El punto inicial de la Ec. (A.1) es $F(0,r)$, pero para una imagen de tamaño par no existen los ejes y el punto inicial corresponde a $F(1,r)$. Resolviendo (A.1) para este punto llegamos a

$$F(1,r) = 1 - 2Ra - a^2 \tag{A.6}$$

Pero $a = \pm 1/2$, como puede verse en las lineas 3-4 de la función original, por lo que $F(1,r) = 1 - R - 1/4$, para $a = +1/2$; y $F(1,r) = 1 + R - 1/4$ para $a = -1/2$. Se puede prescindir del término $-1/4$ perdiendo precisión para los círculos de radio 1 y 2, pero ganando velocidad al realizar todas las operaciones con números enteros, por lo que los puntos iniciales quedan: $F(1,r)_{+1/2} = 1 - R$ y $F(1,r)_{-1/2} = 1 + R$

Entonces, es necesario calcular los puntos iniciales $F(1,r)_{+1/2}$ y $F(1,r)_{-1/2}$, y tenemos tres opciones para escoger el siguiente pixels en $(x+1,y)$, $(x,y-1)$ ó $(x+1,y-1)$. Para la evaluación de estas tres opciones sólo tenemos que sumar las constantes Δx, Δy y Δxy a los puntos iniciales. Se escoge la opción que satisfaga $e_{-1/2} \geq 0$ AND $e_{+1/2} < 0$. Así el algoritmo optimizado queda:

Función optimizada **mask** (img[N][N], r_1, r_2)

$c = N/2$ /** El centro de la imagen **/
for $r_1 \leq r \leq r_2$ **do**
if $r == 1$ **then** /** caso especial **/
$(c-1, c-1), (c-1, c), (c, c-1), (c, c)$
else if $r == 2$ **then** /** caso especial **/
$(c-2, c-1), (c-2, c), (c-1, c-2), (c-1, c+1)$
$(c, c-2), (c, c+1), (c+1, c-1), (c+1, c)$
else
$x = 1$
$y = r$
$e_{-1/2} = 1 + r$
$e_{+1/2} = 1 - r$
while $x < y$ **do**
x, y son parte del círculo:
$(c-y, c-x), (c-x, c-y), (c-y, c+x-1), (c-x, c+y-1)$
$(c+y-1, c+x-1), (c+x-1, c+y-1),$
$(c+y-1, c-x), (c+x-1, c-y)$
$e^{new}_{-1/2} = e_{-1/2} + 2x + 1$
$e^{new}_{+1/2} = e_{+1/2} + 2x + 1$
if $e^{new}_{-1/2} \geq 0$ AND $e^{new}_{+1/2} < 0$ **then**
$e_{-1/2} \leftarrow e^{new}_{-1/2}$
$e_{+1/2} \leftarrow e^{new}_{+1/2}$
$x \leftarrow x + 1$
else
$e^{new}_{-1/2} = e_{-1/2} - 2y + 1$
$e^{new}_{+1/2} = e_{+1/2} - 2y + 1$
if $e^{new}_{-1/2} \geq 0$ AND $e^{new}_{+1/2} < 0$ **then**
$e_{-1/2} \leftarrow e^{new}_{-1/2}$
$e_{+1/2} \leftarrow e^{new}_{+1/2}$
$y \leftarrow y - 1$
else
$e_{-1/2} \leftarrow e^{new}_{-1/2} + 2x + 1$
$e_{+1/2} \leftarrow e^{new}_{+1/2} + 2x + 1$
$x \leftarrow x + 1$
$y \leftarrow y - 1$
endif
endif
endwhile
if $x == y$ **then**
$(c-y, c-x), (c-y, c+x-1), (c+y-1, c+x-1), (c+y+1, c-x)$
endif
endif

```
endfor
```

Apéndice B

Cálculo de la Resolución por Submuestreo Aleatorio

Todos los desarrollos planteados en esta tesis están motivados por la necesidad de aumentar la resolución espacial que se obtiene a partir del análisis de imágenes de macromoléculas biológicas aisladas obtenidas en un Microscopio Electrónico de Transmisión. Como consecuencia, y ya en etapas muy tempranas de este trabajo, nos planteamos realizar un estudio en profundidad sobre los métodos comúnmente empleados en el cálculo de la resolución de una reconstrucción tridimensional. Este estudio es el que se presenta en este apartado.

De forma más concreta, se presenta una aproximación empírica a la tarea de estimar el límite de resolución reproducible que puede alcanzarse cuando se promedia un grupo de imágenes. La solución propuesta esta basada en un método de submuestreo aleatorio. Se derivan así límites de confianza para dos de las medidas de reproductibilidad más ampliamente usadas, el Residuo de Fase Diferencial (DPR, por sus siglas en inglés), y la Correlación en Anillos de Fourier (FRC, por sus siglas en ingleś). Se presentan resultados para especímenes teñidos por una sal pesada (tinción negativa) y especímenes empotrados en hielo. Estos resultados proponen un marco común para observaciones previas, además de sugerencias con respecto a estas medidas a ser más analizadas. La comunicación original se este estudio puede verse en De la Fraga, *et al*, (1995) [27].

B.1 Introducción

Es una tarea común en muchas aplicaciones el hecho de incrementar la razón señal-a-ruido de un grupo de dado de imágenes promediandolas. Esta aproximación se apoya en la suposición de que todas las imágenes son diferentes realizaciones de una "señal" común que está contaminado con ruido aleatorio. Si el término del ruido es aditivo y de media cero, será cancelado después de sumar un número infinito de imágenes. Sin embargo, es difícil en un caso real obtener una medición significativa de la información reproducible alcanzada por promediar sólo un número limitado de imágenes similares.

Métodos intensivos en cómputo basados en remuestreo [38] han ganado importancia en muchos campos biológicos debido a que requieren muy poco en el sentido del modelado, suposiciones, o análisis, y pueden aplicarse automáticamente a la estimación de muchos parámetros, sin importar cómo de complicadas sean sus distribuciones [37]. Aquí se usarán estos métodos para calcular los límites de confianza para dos de las mediciones de información reproducible comúnmente practicadas para analizar imágenes promediadas, el Residuo de Fase Diferencial (DPR) [52] y la Correlación en Anillos de Fourier (FRC) [121, 102]. La metodología de análisis utilizada se aplicará al estudio de dos tipos de preparaciones, el primero obtenido de una muestra en tinción negativa de la chaperonina citoplasmática TCP-1 [74], y el segundo para una preparación crio-congelada de la chaperonina GroEL [75]. Los límites de confianza pueden obtenerse por un método simple de percentiles de la distribución estadística de los parámetros, la cual puede ser aproximada por submuestreo aleatorio [37] de la muestra observada.

B.2 Descripción de la Aproximación

La aproximación experimental está basada en la idea de estimar la distribución estadística de las medidas DPR y FRC para un grupo de datos típico.

Supongamos que tenemos un grupo de N imágenes. Las operaciones para realizar ambas medidas bajo estudio se describen en el pseudocódigo de la Fig. B.1.

Como puede verse en la línea 1(b) de la Fig. B.1, el DPR y el FRC se calculan en el espacio de Fourier, el cual es dividido en anillos concéntricos.

La aplicación del procedimiento de submuestreo aleatorio [36] es como sigue: Dado el grupo de N muestras, se consideran muchas de las maneras de dividir estos datos en dos partes. Cada dos particiones independientes generadas aleatoriamente (líneas 3(a) y 3(c) en la Fig. B.1) es llamada una pseudoréplica. Entonces, se generan muchas pseudoréplicas y se aplican a ellas la medida bajo estudio (línea 3(e) de la Fig. B.1). Estos valores generados nos permiten computar la distribución estadística de la medida dada para el caso actual bajo estudio.

El número de pseudoréplicas necesarias para alcanzar una precisión dada ha sido estudiado para varios métodos de remuestreo [36], pero no para el caso aquí contemplado. Por lo tanto se estudió el comportamiento del método del submuestreo aleatorio propuesto (Fig. B.1) para un amplio espectro de pseudoréplicas, desde N hasta $100N$. Los resultados permanecen estables a partir de N, mostrándose que pueden obtenerse resultados precisos sin introducir un número grande de pseudoréplicas. Los tiempos típicos de ejecución fueron del orden de unos pocos minutos en una estación de trabajo SUN Sparc 10 (con 300 imágenes, para nuestro caso).

A los resultados (línea 5 de la Fig B.1) se les aplica el criterio de que su valor debería estar por encima del nivel de ruido con un 99% como límite de confianza. Nótese que esta razón es usada normalmente en estimadores basados en el FRC [121], pero un criterio más estricto es común en los del DPR [52]. La utilización en

1. **for** $0 \leq i < N$ **do**
   (a) img = read_image(name$_i$);
   (b) IMG[i] = do_fft(img);
2. **endfor**
3. **for** $0 \leq i < N$ **do**
   (a) select_random(N/2 images);
   (b) IMG$_1$ = average_selected_images;

   (c) select_random(N/2 images);
   (d) IMG$_2$ = average_selected_images;

   (e) operation(IMG$_1$, IMG$_2$); /** DPR o FRC **/
4. **endfor**
5. Apply_threshold_to_results;

Figura B.1: Esquema en pseudocódigo de la aproximación usada

este trabajo del mismo criterio para las dos medidas tiene como objeto únicamente el situarlas en un mismo marco de referencia.

B.3 Microscopía Electrónica y Procesado de Imagen

Alícuotas de la chaperonina TCP-1 fueron aplicadas a rejillas cubiertas de colodión y carbón y teñidas con 1% de acetil uranilo. Para la preparación de los especímenes crio-congelados de la chaperonina GroEL se siguieron los métodos generales [35], con la preparación de la película soporte somo se describe en [115]. Las imágenes fueron grabadas en una película Kodak SO-163 usando un microscopio electrónico JEOL 1200EX-II operado a 100 kV con un sistema adicional para realizar el proceso conocido como “dosis mínima”. La Criomicroscopía fue realizada usando un criobrazo GATAN modelo 626. Las imágenes fueron grabadas a 0° con una magnificación de $50,000\times$ usando un desenfoque de 5000-$15,000$ Å. Las micrografías fueron reveladas usando el revelador D19 (de Kodak) durante 12 min y digitalizadas usando una cámara Eikonix 1412 con una ventana de muestreo de tamaño 4.7 Å pixel^{-1}.

Fueron usadas seis micrografías de la chaperonina TCP-1 y cinco de la GroEL. De cada grupo de micrografías se seleccionaron dos grupos de datos, uno conteniendo 300 imágenes del ruido de fondo, y otro conteniendo 300 marcos típicos del espécimen. El procesado de las imágenes individuales, todas ellas de dimensiones 64×64 pixels, fue llevado a cabo como en [15]. Las imágenes de las chaperoninas fueron alineadas usando un algoritmo de referencia dinámico [83]. En la Fig. B.4 se presenta una selección representativa de las imágenes. Las Figs. B.4(a) y B.4(b) corresponden al fondo y al espécimen para la preparación de TCP-1, mientras que las Figs. B.4(c) y B.4(d) corresponden a la preparación de GroEL. A los datos de les aplicó el procedimiento de la Fig. B.1 para obtener las distribuciones estadísticas del DPR y el FRC.

El modelo básico de formación de imagen usado fue $g = f + n$, con g siendo la imagen grabada, f la imagen del espécimen "perfecto", y n el ruido. Calculando la distribución estadística del DPR y FRC aplicadas a imágenes que no contienen espécimen biológico, esto es, $f = 0$, se estudió el término del ruido. Por otro lado, el estudio de las medidas sobre las imágenes de los especímenes nos provee la distribución éstas cuando se aplican a la imagen g. Bajo estas consideraciones, el punto de intersección de las distribuciones nos indicaría la resolución más allá de la cual la variabilidad observada podría ser debida totalmente al ruido.

En lo que sigue, estas distribuciones serán caracterizadas por sus medias y sus límites al 99% de confianza. La meta fue obtener el valor de corte más allá del cual no existe información. De esta manera se intenta minimizar la probabilidad de aceptar como señal una muestra de ruido o de rechazar una muestra de señal considerándola como ruido. (errores tipo I y II) [122].

B.4 El Residuo de Fase Diferencial

Esta medida de la información reproducible fue introducida por Frank y colaboradores [52] y está definida como

$$\Delta\theta = \left[\frac{\sum(|F_1| + |F_2|)(\delta\theta)^2}{\sum(|F_1| + |F_2|)}\right]^{1/2}$$

donde F_1 es un coeficiente de Fourier sobre la imagen 1, y F_2 es el coeficiente correspondiente sobre la imagen2. $\delta\theta$ es la diferencia de fase entre F_1 y F_2. El residuo de fase es calculado a lo largo de círculos concéntricos en la transformada de Fourier. El algoritmo de Bresenham, [45] cap. 11, fue usado para obtener los círculos, dando un incremento casi lineal en el número de coeficientes con respecto al radio.

El límite teórico del DPR, bajo la suposición de tener datos sólo con ruido blanco, es de 103.92° [119], y es costumbre, aunque de alguna manera es arbitrario, tomar la medición valor de 45° [52]. No han sido reportados estudios sobre los límites de confianza.

La distribución empírica para el espécimen teñido negativamente este presentada en la Fig. B.3(a). Las líneas marcadas con N corresponden a la distribución para el

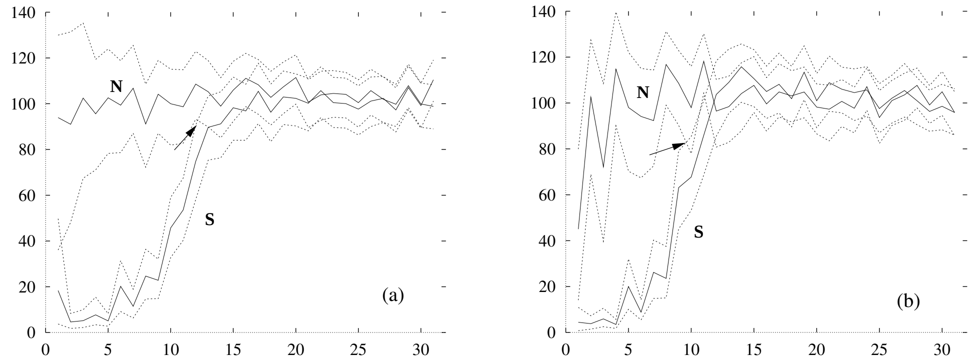


Figura B.2: Distribución del Residuo de Fase Diferencial (DPR) para el ruido (N) y las imágenes conteniendo espécimen (S): (a) corresponde al caso de la chaperorina TCP-1 (teñida negativamente) y (b) para la proteína GroEL (criocongelada). La línea continua representa el valor medio, mientras que las líneas punteadas representan los límites de confianza al menos y más 99%. El punto de corte entre las dos curvas son el 99% de nivel de confianza se encuentra a 12 pixels de Fourier (marcado con una flecha) en (a) y a 9 pixels de Fourier en (b).

caso del ruido. La distribución de las imágenes conteniendo espécimen biológico es mostrada en la misma figura, etiquetado con S. Inmediatamente puede notarse que el limite teórico del ruido alrededor de 104° es efecto se mantiene, y que el nivel de corte en el que más allá de él no existe señal, en el promediado y con un 99% de confianza, es alrededor de 12 pixels de Fourier. El valor del DPR en el punto de corte esta alrededor de 80°. La distribución estadística para el espécimen criocongelado se muestra en la Fig. B.3(b). El nivel de corte en el que más allá no hay señal presente, en el promediado y con un 99% de confianza, es de alrededor de 9 pixels de Fourier. Como era de esperarse [114], el valor de corte de 45° que es usualmente asociado con esta medida es más que conservativo, y a ese umbral hay aún información válida que está presente en el promediado.

B.5 La Correlación en Anillos de Fourier

Esta medida fue introducida por van Heel y colaboradores [121], y por Saxton y Baumeister [102]. Está definida como

$$\mathrm{FRC} = \frac{\sum F_1 F_2^*}{\left(\sum |F_1|^2 \cdot \sum |F_2|^2\right)^{1/2}}$$

De nuevo, F_1, F_2 son los dos coeficientes de Fourier de las imágenes 1 y 2, y como antes, el FRC es calculado a lo largo de anillos concéntricos.

La aproximación teórica aproximada de esta medida para el caso de una señal que solo contenga ruido blando, resulta en una curva definiendo un umbral que está dado por la expresión $2/\sqrt{N}$, siendo N en número de puntos dentro de cada círculo donde se aplica la medida.

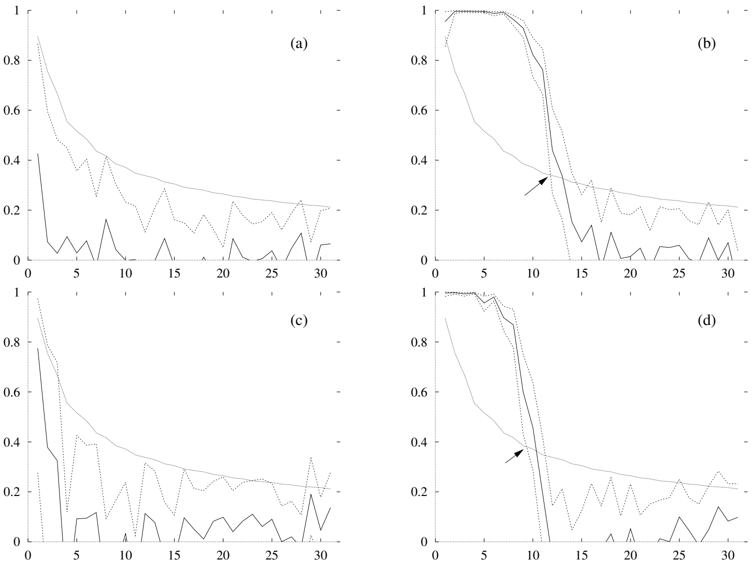


Figura B.3: Distribución de la Correlación en Anillos de Fourier (FRC): (a) y (b) corresponden al caso de imágenes conteniendo únicamente ruido y al caso de imágenes que si contienen un espécimen, respectivamente, resultado del estudio de la chaperonina TCP-1; mientras que (c) y (d) corresponden al caso del ruido y del espécimen para la proteína GroEL. La línea gruesa representa el valor medio, mientras que la línea punteada representa los límites de confianza al menos y más 99%. El punto de corte entre las dos curvas con el 99% de confianza esta a 12 pixels de Fourier (marcado con una flecha) para la chaperonina TCP-1, y a 9 pixels de Fourier para la GroEL.

La distribución experimental obtenida en este trabajo para el caso del ruido correspondiente a las muestras teñidas negativamente de TCP-1 se presenta en la Fig. B.2(a), mostrando que el límite teórico aproximado es de hecho similar al límite de confianza del 99% percentil calculado empíricamente por submuestreo aleatorio. En la Fig. B.2(b) se presentan los resultados obtenidos para el subgrupo de imágenes conteniendo espécimen. El punto de corte entre las curvas del estimador del FRC con un 99% de límite de confianza para el espécimen y el limite teórico aproximado para el ruido es marcado con una flecha, y está alrededor de 12 pixels de Fourier.

Para la GroEL criocongelada, la distribución empírica para el caso del ruido es presentado en la Fig. B.2(c). En la B.2(d) se presentan los resultados obtenidos para el subgrupo conteniendo espécimen. El punto de corte obtenido como de describió en el párrafo anterior se encuentra alrededor de 9 pixels de Fourier.

B.6 Discusión

La distribución estadística de dos medidas importantes de la información reproducible, el Residuo de Fase Diferencial (DPR) y la Correlación en Anillos de Fourier (FRC), han sido obtenidas para el caso de datos reales, primero para un grupo conteniendo parte de una micrografía real sin espécimen en ellas, que se ha considerado como una realización del ruido; y segundo para imágenes conteniendo especímenes biológicos. Las distribuciones obtenidos nos proveen límites precisos sobre la variabilidad de estas mediciones cuando diferentes muestras de subpoblaciones son tomadas en cuenta.

Un punto importante a ser comentado aquí es que ninguna de las dos mediciones, ni el DPR ni el FRC, están directamente conectadas con cualquier forma de “razón señal-a-ruido”, y, por lo tanto, los resultados de este trabajo no proveen información sobre el tema. El otro método usado en este campo que intenta medir la razón señal-a-ruido del promediado de un grupo de imágenes bajo la suposición de una serie de hipótesis estadísticas es el llamado SSNR (La Razón Espectral Señal-a-Ruido, por sus siglas en inglés) [114]. Esta medida no ha sido considerada en este trabajo debido a que su cálculo es diferente que el de las dos medidas aquí tratadas, y por lo tanto diverge de la situación que se ha completado en estas pruebas.

La dispersión de los valores de las medidas de DPR y del FRC, cuando son aplicadas al caso de tener sólo espécimen, indican, dentro de un 99% como límite de confianza alrededor de sus medias, que sus valores cambian hasta un 30%. De hecho, una dispersión de 30° se aprecia en las Figs. B.3(a) y B.3(b) para el DPR, mientras hay una dispersión de alrededor de 0.3i es evidente en las Figs. B.2(b) y B.2(d) para el FRC.

La observación de Radermacher [89] de que el DPR y el FRC proveen resultados similares cuando el umbral de significancia del DPR es tomado alrededor de 85° coincide con los resultados de este trabajo, ya que ese umbral representa en las situaciones prácticas el limite más bajo al 99% más allá del cual el ruido es el término dominante. Esto es, ambos métodos nos dan la misma respuesta cuando es propuesta la misma pregunta. Es importante resaltar que la metodología propuesta en este trabajo provee los límites de confianza apropiados para un grupo de datos concreto a ser analizado, sin requerir *a priori* cualquier umbral fijo.

Un último punto a mencionar es que la aproximación propuesta aquí es independiente de hipótesis sobre distribuciones estadísticas concretas, que es un punto de diferencia muy grande con respecto a cualquier otro procedimiento y que, además, puede ser extendido o otras medidas de información reproducible. El tiempo de computo necesario para todos los cálculos es solamente del orden de unos pocos minutos en una estación de trabajo estándar.

B.7 Conclusión

En conclusión, los resultados aquí presentados indican que es necesario un cálculo explícito de la variabilidad de las mediciones que comúnmente son usadas para estimar la información reproducible de una imagen promediada. La indicación del valor de una simple medición sin considerar el actual ruido experimental o la variabilidad de las imágenes parece no ser un método adecuado.

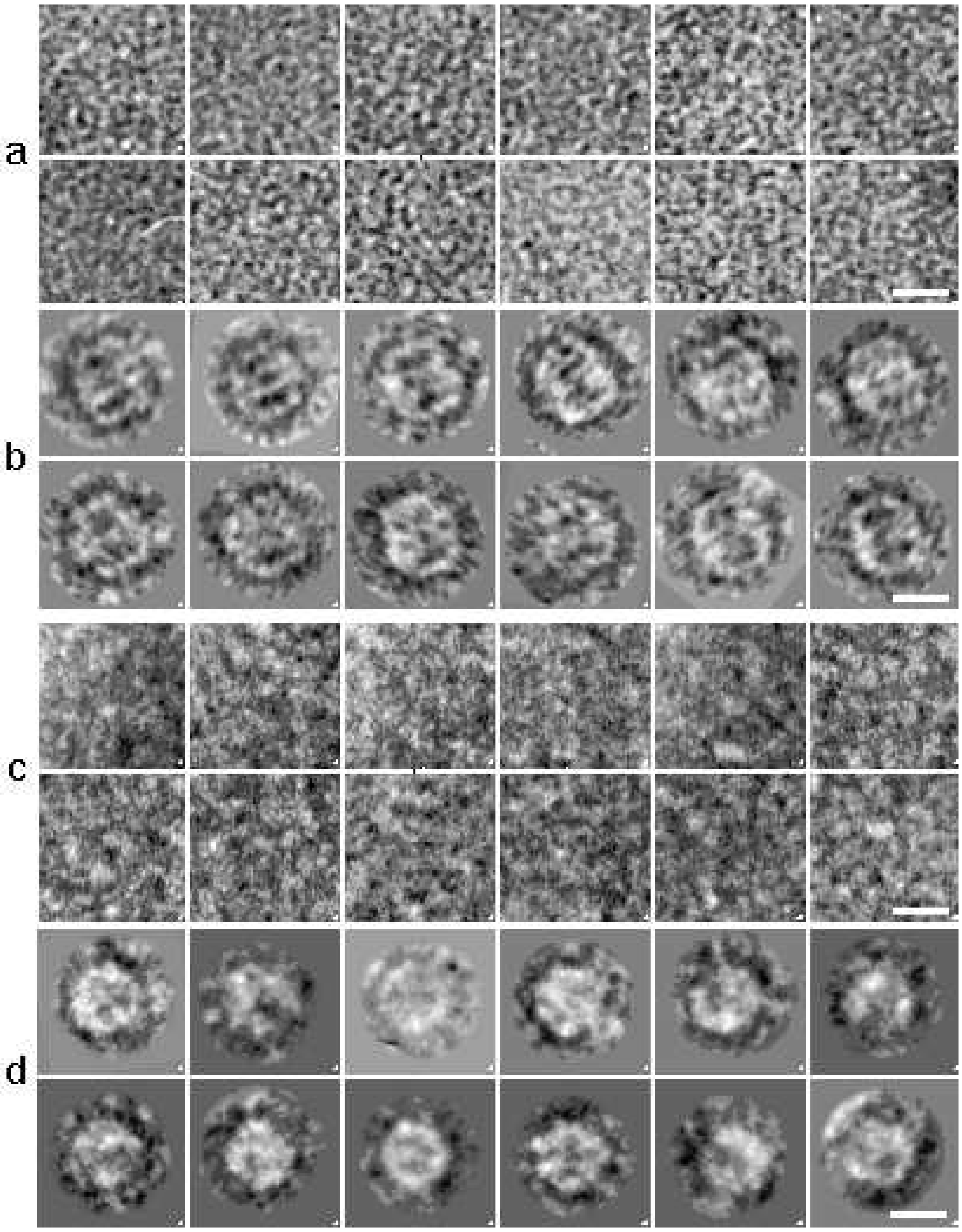


Figura B.4: Una galería representativa de imágenes conteniendo 12 muestras de cada grupo. (a) Fondo obtenido de una preparación en tinción negativa de la chaperonina TCP-1, (b) La chaperonina TCP-1 después del alineamiento, (c) fondo de una preparación criocongelada, (d) La chaperonina GroEL después del alineamiento. La barra de escala representa 10 nm.

Bibliografía

[1] B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts, and J.D. Watson. *Molecular Biology of the Cell.* Garland Publishing, 1994.

[2] T.S. Baker and J.E. Johnson. Low resolution meets high: towards a resolution continuum from cells to atoms. *Current Opin. Struct. Biol.*, 6:585–594, 1996.

[3] D.R. Beniac, M.D. Luckevich, G.J. Czarnota, T.A. Tompkins, R.A. Ridsdale, F.P. Ottensmeyer, M.A. Moscarello, and G. Harauz. Three-dimensional structure of myelin basic protein. I. reconstruction via angular reconstitution of randomly oriented single particles. *J. Biol. Chem.*, 272(7):4261–4268, 1997.

[4] E.J. Boekema, J.A. Berden, and M. van Heel. Structure of mitochondrial f_1-ATPase studied by electron microscopy and image processing. *Biochim. Biophys. Acta*, 851:353–360.

[5] N. Boisset, P. Penczeck, P. Franois, J. Frank, and J.N. Lamy. Three-dimensional architecture of human $\alpha 2$–macroglobulina transformed with methylamine. *J. Mol. Biol.*, 232:522–529, 1993.

[6] N. Boisset, P. Penczek, J.-C. Taveau, J. Lamy, J. Frank, and J.Ñ. Lamy. Three-dimensional recontruction of *androctonus australis* hemocyanin labeled with a monocronal fab fragment. *J. Struct. Biol.*, 115:16–29, 1995.

[7] B. Böttcher, S.A. Wynne, and R.A. Crowther. Determination of the fold of the core protein of hepatitis b virus by electron microscopy. *Nature*, 386:88–91, 1997.

[8] J.J. Bozzola and L.D. Russell. *Electron Microscopy. Principles and Techniques for Biologists.* Jones and Bartlett, 1992. Boston, USA.

[9] R.N. Bracewell. *The Fourier Transform and its Applications.* McGraw-Hill, 1986.

[10] J. Bresenham. Pixel-processing fundamentals. *IEEE CG&A*, 16:74–82, Jan 1996.

[11] J. A. Browne, G. T. Herman, and D. Odhner. SNARK93: A programming system for image reconstruction from projections. Technical Report MIPG198, Medical Image Processing Group, University of Pennsylvania, Philadelphia, P.A. U.S.A., aug 1993.

[12] J. M. Carazo. *Electron Tomograpy*, chapter The Accuracy of 3D Reconstructions from Incomplete Information and the Use of Restoration Methods. Joachim Frank. Plenium Press, 1992.

[13] J. M. Carazo and J. Frank. Three-dimensional matching of macromolecular structures obtained from electron microscopy. *Ultramicroscopy*, 25:13–22, 1988.

[14] J.M. Carazo and J.L. Carrascosa. Information recovery in missing angular data cases: An approach by the convex projections method in three dimensions. *J. Microsc.*, 145:23–43, 1987.

[15] J.M. Carazo, S. Marco, G. Abella, J.L Carrascosa, J.P. Secilla, and M. Muyal. *J. Struct. Biol*, 106:211–215, 1991.

[16] J.M. Carazo, T. Wagenknecht, and J. Frank. Variations of the three-dimensional structure of *escherichia coli* ribosome in the range of overlap views. *Biophys. J.*, 55:465–477, 1989.

[17] T.L. Casavant, P. Tvrdík, and F. Plásil. *Parallel Computers, Theory and Practice*. IEEE Comp. Soc. Press, 1996.

[18] S-Li Chang, D. H. Du, J. Hsieh, R.P.Tsang, and M. Lin. Enhanced pvm communications over a high-speed lan. *IEEE Parallel & Distributed Tech.*, 3:20–32, 1995.

[19] R.H. Cheng, V.S. Reddy, N.H. Olson, A.J. Fisher, T.S. Backer, and J.E. Johnson. Funtional implications of quiasi-equivalence in a t=3 icosohedral animal virus established by cryo-electron microscopy and X-ray crystallography. *Structure*, 2:271–282, 1994.

[20] W. Chiu. What does electron cryomicroscopy provide that X-ray crystallography and NMR spectroscopy cannot. *Annu. Rev. Biophys. Biomol. Struct.*, 22:233–255, 1993.

[21] J.F. Conway, N. Cheng, A. Zlotnick, P.T. Wingfield, S.J. Stahl, and A.C. Steven. Visualization of a 4-helix bundle in the hepatitis b virus capsid by cryo-electron microscopy. *Nature*, 386:91–94, 1997.

[22] L.A. Crowl. How to measure, present, and compare parallel performance. *IEEE Parallel & Distributed Tech.*, 2:9–25, 1994.

[23] R.A. Crowther, L.A. Amos, J.T. Finch, D.J. De Rosier, and A. Klug. Three dimensional reconstruction of spherical viruses by fourier synthesis from electron micrographs. *Nature*, 226:421–425, 1970.

[24] G.J. Czarnota, D.W. Andrews, N.A. Farrow, and F.P. Ottensmeyer. A structure for the signal sequence binding protein SRP54: 3D reconstruction from STEM images of single molecules. *J. Struct. Biol.*, 113:35–46, 1994.

[25] Felix de Hass, J.-C. Taveau, N. Boisset, O. Lambert, S.N. Vinogradov, and J. Lamy. Three-dimensional recontruction of the chlorocruorin of the polychaete annelid *eudistylia vancouverii*. *J. Mol. Biol.*, 225:140–153, 1996.

[26] L.G. de la Fraga, J.M. Carazo, and G.T. Herman. A method for obtaining a low resolution 3d reconstruction from very noisy projections. In *Proc. VII National Symposium on Pattern Recognition and Image Analysis*, volume 1, pages 347–352, 1997.

[27] L.G. de la Fraga, J. Dopazo, and J. M. Carazo. Confidence limits for resolution estimation in image averaging by random subsamplig. *Ultramicroscopy*, 60:385–391, 1995.

[28] D.J. De Rosier and A. Klug. Reconstruction of three dimensional structures from electron micrographs. *Nature*, 217:130–134, 1968.

[29] D.J. De Rosier and P.B. Moore. Reconstruction of three dimensional structures from electron micrographs of structures with helical symmetry. *J. Mol. Biol.*, 52:335–369, 1970.

[30] S.R. Deans. *The Radon transform and some of its applications.* Wiley, 1983. New York.

[31] J. Dengler. A multiresolution approach to the 3d reconstruction from an electron microscope tilt series solving the alignment problem without gold particles. *Ultramicroscopy*, 30:337–348, 1989.

[32] J.J. Dongarra, D.A. Otto, M. Snir, and D. Walker. A message passing standard for mpp and workstations. *Communications of the ACM*, 30:337–348, 1989.

[33] K.H. Downing and D.A. Grano. Analysis of photographic emulsions for electron microscopy of two-dimensional crystalline specimens. *Ultramicroscopy*, 7:381–404, 1982.

[34] P.D. Dube, E.V. Orlova, F. Zemlin, M. van Heel, J.R. Harris, and J. Markl. Three-dimensional structure of keyhole limpet hemocyanin by cryoelectron microscopy and angular reconstitution. *J. Struc. Biol*, 115:226–232, 1995.

[35] J. Dubochet, M. Adrian, J.J. Chang, J.C. Homo, J. Lepault, A.W. McDowall, and P. Schultz. Cryo-electron microscopy of vitrified specimens. *Rev. Biophys.*, 21:129–228, 1988.

[36] B. Efron. *Am. Statist.*, 82:36–40, 1987.

[37] B. Efron and G. Gong. *Am. Statist.*, 37:36–39, 1983.

[38] B. Efron and R. Tibsirani. *Science*, 253:390–391, 1991.

[39] A. Engel. Biological applications of scanning probe microscopes. *Annual Rev. Biophys. Chem*, 20:79–108, 1991.

[40] H.P. Erickson and A. Klug. Measurement and compensation of defocusing and aberrations by fourier processing of electron micrographs. *Phil. Trans. Roy. Soc. Lond.*, B 261:105–118, 1971.

[41] N. A. Farrow and F. P. Ottensmeyer. A posteriori determination of relative projection directions of arbitrarily oriented macromolecules. *J. Opt. Soc. Am. A*, 9(10):1749–1760, 1992.

[42] N. A. Farrow and F. P. Ottensmeyer. Automatic 3D alignment of projection images of randomly oriented macromolecules. *Ultramicroscopy*, 52:141–156, 1993.

[43] J.J. Fernández, J.M. Valpuesta, J.R. Sanjurjo, J.L. Carrascosa, and J.M. Carazo. Image processing of two-dimensional biological crystals in transmission electron microscopy. In *Proc. VII National Symposium on Pattern Recognition and Image Analysis*, volume 1, pages 221–226, 1997.

[44] J.T. Finch and A.J. Klug. *J. Mol. Biol.*, 13:1–12, 1965.

[45] J. D. Foley and A. van Dam. *Fundamentals of Interactive Computer Graphics.* Addison-Wesley, 1984.

[46] J. D. Foley, A. van Dam, S. K. Feiner, and J.F. Hughes. *Computer Graphics: Principles and Practice*, chapter 5. Addison-Wesley, second edition, 1990.

[47] I. T. Foster. *Designing and building parallel programs: concepts and tools for parallel software engineering.* Addison-Wesley, 1994.

[48] N.R. Francis, G.E. Sosinsky, D. Thomas, and D.J. De Rosier. Isolation, characterization and structure of bacterial flagellar motors containing the switch complex. *J. Mol. Biol.*, 235:1261–1270, 1994.

[49] J. Frank. The envelope of electron microscopic transfer functions for partially coherent illumination. *Optik*, 38:519–536, 1973.

[50] J. Frank. *Three-Dimensional Electron Microscopy of Macromolecular Assemblies.* Academic Press, 1996. San Diego, CA.

[51] J. Frank, M. Radermacher, P. Penczek, J. Zhu, Y. Li, M. Ladjadj, and A. Leith. SPIDER and WEB: Processing and visualization of images in 3D electron microscopy and related fields. *J. Struct. Biol.*, 116:190–199, 1996.

[52] J. Frank, A. Verschoor, and M. Boublik. Computer averaging of electron micrographs of 40s ribosomal subunits. *Science*, 214:1353–1355, 1981.

[53] A.B. Goncharov and M.S. Gelfand. Determination of mutual orientation of identical particles from their projections by the moments method. *Ultramicroscopy*, 25:317–328, 1988.

[54] G. Harauz and F. P. Ottensmeyer. Direct three-dimensional reconstruction for macromolecular complexes from electro micrographs. *Ultramicroscopy*, 12:309–320, 1984.

[55] G. Harauz and F. P. Ottensmeyer. Nucleosome reconstruction via phosphorus mapping. *Science*, 226:936–940, 1984.

[56] H. Harauz. Representations of rotations ny unit quaternions. *Ultramicroscopy*, 33:209–213, 1990.

[57] P. W. Hawkes, editor. *Computer Processing of Electron Microscope Images.* Springer-Verlag, 1980.

[58] P.W. Hawkes. The electron microscope as a structure projector. In J. Frank, editor, *Electron Tomography*, pages 17–38. Plenum, 1992.

[59] P.W. Hawkes and E. Kasper. *Principles of Electron Optics: Wave Optics*, volume 3. Academic Press, 1996.

[60] P.W. Hawkes and U. Valdrè, editors. *Biophysical Electron Microscopy.* Academic Press, 1990.

[61] R. Henderson, J.M. Baldwin, T.A. Ceska, F. Zemlin, E. Beckmann, and K. H. Downing. Model for the structure of bacteriorhodopsin based on high-resolution electron cryo-microscopy. *Journal of Molecular Biology*, 213:899–929, 1990.

[62] R. Henderson and P.N. Unwin. Three-dimensional model of purple membrane obtained by electron microscopy. *Nature*, 257:28–32, 1975.

[63] G. T. Herman. *Image Reconstruction from Projections: The Fundamentals of Computerized Tomography.* Academic Press, 1980.

[64] T.W. Jeng, R.A. Crowther, G. Stubbs, and W. Chiu. Visualization of α-helices in TMV by cryo-electron microscopy. *J. Mol. Biol.*, 205:251–257, 1989.

[65] J.C. Kendrew. Myoglobin and the structure of proteins. *Science*, 139:1259–1266, 1963.

[66] W. Kühlbrandt, D.N. Wang, and Y. Fujiyoshi. Atomic model of plant light-harvesting complex by electron crystallography. *Nature*, 367:614–621, 1994.

[67] V. Kumar, A. Grama, S. Gupta, and G. Karypis. *Introduction to Parallel Computers Computing: Design and Analysis of Algorithms.* The Benjamin/Cumming Publising Company, Inc., 1994.

[68] O. Lambert, N. Boisset, J.-C. Taveau, G. Préaux, and J.Ñ. Lamy. Three-dimensional recontruction of *sepia officinalis* hemocyanin from frozen-hidrated specimens. *Arch. Bioch. and Bioph.*, 316(2):950–959, 1995.

[69] O. Lambert, N. Boisset, J.-C. Taveau, G. Préaux, and J.Ñ. Lamy. Three-dimensional recontruction of the α_D and β_C-hemocyanins of *helix pomatia* from frozed-hidrated specimens. *J. Mol. Biol.*, 248:431–448, 1995.

[70] J.P. Langmore and M.F. Smith. Quantitative energy-filtered electron microscopy of biological molecules in ice. *Ultramicroscopy*, 46:349–373, 1992.

[71] F.A. Lenz. Transfer of image information in the electron microscope. In U. Valdrè, editor, *Electron Microscopy in Material Sciences*, pages 540–569. Academic Press, 1971.

[72] R. Marabini, G.T. Herman, and J.M. Carazo. 3D reconstruction in electron microscopy using ART with smooth spherically symmetric volume elements (blobs). *Biophys. J.*, 1997. In Press.

[73] R Marabini, I. M. Masegosa, M. C. San Martin, J.J. Fernández, S. Marco, L. G. de la Fraga, C. Vaquerizo, and J. M. Carazo. Xmipp: An image processing package for electron microscopy. *J. Struct. Biol.*, 116:237–240, 1996.

[74] S. Marco, J. L. Carrascosa, and J. M. Valpuesta. Reversible interaction of β-actin along the channel of the TCP-1 cytoplasmic chaperonin. *Biophysical Journal*, 67:364–368, 1994.

[75] S. Marco, J.M. Valpuesta, G. Rivas, G. Andres, C. San Martín, and J. L. Carrascosa. A structural model for the GroEL chaperonin. *FEMS Microbiology Lett.*, 106:301–308, 1993.

[76] N.E. McKenzie, P.R. Gooley, and L.A. Hardway. Solving solution structures of physiologically relevant proteins by nmr spectroscopy. *Annual Rev. Physiol.*, 54:749–773, 1992.

[77] G.A. Meek. *Practical Electron Microscopy for Biologist.* John Wiley and Sons, 1982.

[78] R.M. Mersereau and A.V. Oppenheim. Digital reconstruction of multidimensional signals from their projections. *Proc. IEEE*, 62:1319–1338, 1974.

[79] D.G. Morgan, C. Owen, L.A. Melanson, and D.J. De Rosier. *J. Mol. Biol.*, 249:88–110, 1995.

[80] A.V. Oppenheim and R. W. Schafer. *Digital Signal Processing*. Prentice-Hall, 1975.

[81] A. Papoulis. *Probability, Random Variables and Stochastic Process*, chapter 5. McGraw-Hill, third edition, 1991.

[82] T. Pavlidis. *Algorithms for Graphics and Image Processing*. Computer Science Press, 1982.

[83] P. Penczek, M. Radermacher, and J. Frank. Three-dimensional reconstruction of single particles embeded in ice. *Ultramicroscopy*, 40:33–53, 1992.

[84] P. A. Penczek, R. A. Grassuni, and J. Frank. The ribosome at improved resolution: new techniques for for merging and orientation refinement in 3D cryo-electron microscopy of biological particles. *Ultramicroscopy*, 53:251–270, 1994.

[85] P.A. Penczek, J. Zhu, and J. Frank. A common-lines based method for determining orientations for $n > 3$ particle projections simultaneously. *Ultramicroscopy*, 63:205–218, 1996.

[86] M.F. Peruts. X-ray analysis of hemoglobin. *Science*, 140:863–869, 1963.

[87] J. Protrć, M. Tomasević, and V. Milunitović. Distributed shared memory: Concepts and systems. *IEEE Parallel & Distributed Tech.*, 4:63–79, 1996.

[88] S.W. Provencher and R.W. Vogel. Three-dimensional reconstruction from electron micrographs of disordered specimens. i. method. *Ultramicroscopy*, 25:209–222, 1988.

[89] M. Radermacher. The three-dimensional reconstruction of single particles from random and non-random tilt series. *J. Electron Microsc. Tech.*, 9:359–394, 1988.

[90] M. Radermacher. Weighted back-projection methods. In J. Frank, editor, *Electron Tomography*, pages 91–115. Plenum, 1992.

[91] M. Radermacher. Three-dimensional reconstruction from random projections: Orientational alignment via Radon transforms. *Ultramicroscopy*, 53:121–136, 1994.

[92] M. Radermacher, J. Frank, and C.A. Mannella. Correlation averaging: Lattice separation and resolution assessment. In *Proceedings of the 44th Annual Meeting EMSA*, pages 140–143, 1986.

[93] M. Radermacher, T. Wagenknecht, V. Verschoor, and J. Frank. Three-dimensional reconstruction from single-exposure, random conical tilt series applied to the 50S ribosomal subunit of *escherichia coli*. *J. Microscopy*, 146:113–136, 1987.

[94] J. Radon. Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten. berichte über die verhandlungen der königlich sächsischen gesellschaft der wissenschaften zu leipzig. *Math. Phys. Klasse*, 69:262–277, 1917.

[95] A. M. Roseman, S. Chen, H. White, K. Braig, and H. R. Saibil. The chaperonin ATPase cycle: Mechanism of allosteric switching and movements of substrate-binding domains in GroEL. *Cell*, 87:241–251, 1996.

[96] A.M. Roseman, S. Chen, H. Write ans K. Braig, and H.R. Saibil. The chaperonin ATPase cycle: mechanism of allosteric switching and movements of substrate-binding domains in GroEL. *Cell*, 87:241–251, 1996.

[97] J. C. Russ. *The Image Processing Handbook. 2nd. ed.* CRC Press, 1994.

[98] M. Salomone. Naturaleza 1, ordenadores 0. Periódico: EL PAIS, 4 de junio 1997. Sec. SOCIEDAD. p. 32.

[99] D.B. Salzman. A method of general moments for orienting 2d projections of unknown 3d objects. *Comput. Vis. Graphics and Image Proc*, 50:129–156, 1990.

[100] Carmen San Martin Pastrana. *Procesamiento de Imágenes de Microscopía Electrónica Aplicado al Estudio de la Estructura de Dos ADN-Helicasas Hexaméricas Representativas.* PhD thesis, Universidad de Santiago de Compostela, Junio 1997.

[101] W. O. Saxton. *Computer Techniques for Image Processing in Electron Microscopy.* Academic Press, 1978.

[102] W.O. Saxton and W. Baumeister. The correlation averaging of a regularly arranged bacterial cell envelope protein. *J. Microsc.*, 127:127–138, 1982.

[103] R.R. Schröder, E. Hofmann, and J.F. Menetret. Zero-loss energy filtering as improved imaging mode in cryo-electron microscopy of frozen-hydrated specimens. *J. Struct. Biol.*, 105:28–34, 1990.

[104] I.I. Serycheva, E.V. Orlova, W. Chiu, M.B. Sherman, S. Hamilton, and M. van Heel. Electron cryomicroscopy and angular reconstitution used to visualize the skeletal muscle calcium release channel. *Struc. Biol*, 2:18–24, 1995.

[105] M.I. Sezan. An overview of convex proejctions theory and its application to image recovery problems. *Ultramicroscopy*, 40:55–67, 1992.

[106] M.I. Sezan and H. Stark. Image restoration by the method of convex projections. ii. applications and numerical results. *IEEE Trans. Med. Imaging*, 1:95–101, 1982.

[107] S. Shekhar, S. Ravada, V. Kumar, D. Chubb, and G. Turner. Parallelizing a gis over a shared addres space architecture. *Computer*, 29(12), 1996.

[108] Silicon Graphics Inc. *MIPSpro Fortran 77 Programmer's Guide*.

[109] R.C. Singleton. An algorithm for computing the mixed radix fast fourier transform. *IEEE Trans. Audio Electroacust.*, AU-17:93–103, 1969. Source code in module FFT from package GO, in NETLIB: Send mail to netlib@ornl.gov, with "send index".

[110] H. Stark, E.V. Orlova, J. Rinke-Appel, N. Jünke, F. Mueller, M. Rodnina, W. Wintermeyer, R. Brimacombe, and M. van Heel. Arrangement of tRNAs in pre- and post-translocational ribosomes revealed by cryomicroscopy. *Cell*, 88:19–28, 1997.

[111] V.S. Sunderam. Network-based concurrent computing on the pvm sytem. *Concurrency: Practice and Experience*, 2(4):293–311, 1992.

[112] F. Thon. Phase contrast electron microscopy. In U. Valdrè, editor, *Electron Microscopy in Material Sciences*, pages 572–625. Academic Press, 1971.

[113] B.L. Trus, R.B.S. Roden, H.L. Greenstone, M. Vrhel, J.T. Schiller, and F.P. Booy. Novel structural features of bovine papillomavirus capsid revealed a three-dimensional reconstruction to 9 å resolution. *Nature Struc. Biol.*, 4(5):413–420, 1997.

[114] M. Unser, B. Trus, and A. Steven. Parallelizing a gis over a shared addres space architecture. *Ultramicroscopy*, 23:39–12, 1987.

[115] N. Unwin, C. Toyoshima, and E. Kubalek. *J. Cell Biol*, 107:1123–1130, 1988.

[116] van Heel. Clasification of very large electron microscopical image data sets. *Optick*, 82:114–126, 1989.

[117] M. van Heel. In A. Csanády, P. Röhlich, and D. Szabó, editors, *Proc. 8th Eur. Congr. on Electron Microscopy, Budapest*, volume 2, pages 1347–1348, 1984.

[118] M. van Heel. Angular reconstitution: a posteriori assignment of projection direction for 3D reconstruction. *Ultramicroscopy*, 21:111–124, 1987.

[119] M. van Heel. Similarity measures between images. *Ultramicroscopy*, 21:95–99, 1987.

[120] M. van Heel and J. Frank. Use of multivariate statistics in analysing the images of biological macromolecules. *Ultramicroscopy*, 6:187–194, 1981.

[121] M. van Heel, W. Keegstra, W. Schutter, and E.F.J. van Bruggen. *Life Chemistry Reportsm, Suppl. 1, The Structure and Function of Invertebrate Respiratory proteins*, chapter Arthropod hemocyanin structures studied by image analysis, pages 69–73. Harwood Academic, UK., 1982.

[122] S. B. Vanderman. *Statistics for Engineering Problem Solving.* IEEE Press, New York, 1994.

[123] R.W. Vogel and S.W. Provencher. Three-dimensional reconstruction from electron micrographs of disordered specimens. ii. implementation and results. *Ultramicroscopy*, 25:223–240, 1988.

[124] R.H. Wade. A brief look at imaging and contrast transfer. *Ultramicroscopy*, 46:145–156, 1992.

[125] R.H. Wade and J. Frank. Electron microscope transfer functions for partially coherent axial illumination and chromatic defocus spread. *Optik*, 49:81–92, 1977.

[126] T.Y. Young and K-S. Fu, editors. *Handbook of Pattern Recognition and Image Processing.* Academic Press, 1986.

[127] Z.H. Zhou and W. Chiu. Prospects for using an IVEM with a FEG for imaging macromolecules towards atomic resolution. *Ultramicroscopy*, 49:407–416, 1993.

[128] J. Zhu. *Methods to achieve high resolution in three-dimensional cryo-electron microscopy of macromolecules and application to the* 70S *Escherichia Coli ribosome*. PhD thesis, School of Public Health, Dept. Biomedical sciences. State university of New York, 1996.

[129] J. Zhu and J. Frank. Accurate retrieval of transfer function from defocus series. In *Proc. 13th International Congress on Electron Microscopy*, volume 1, pages 465–466, 1994.

[130] J. Zhu, P.A. Penczek, R. Schröder, and J. Frank. Three-dimensional reconstruction with contrast transfer function correction from energy-filtered cryoelectron micrographs: Procedure and application to the 70S escherichia coli ribosome. *J. Struct. Biol.*, 118:197–219, 1997.

— No: mi tesis es que la vida que hay sobre la Tierra puede cuidar de si misma. Para la mentalidad de un ser humano, cien años es mucho tiempo. Cien años atrás no teníamos automóviles ni aviones ni ordenadores ni vacunas ... Era un mundo del todo diferente. Pero, para la Tierra, cien años no es *nada*. Este planeta vive y respira en una escala mucho más vasta. No nos podemos imaginar sus lentos y poderosos ritmos, y carecemos de humildad para intentarlo. Hemos sido residentes de este planeta el tiempo de un abrir y cerrar de ojos. Si mañana desaparecemos, la Tierra no nos echará en falta.

Ian Malcolm
Séptima Iteración. Destruyendo el Mundo.
Jurassic Park
Michael Crichton