



UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO

FACULTAD DE INGENIERIA

FALLA DE ORIGEN

SISTEMA DE VISUALIZACION Y ANALISIS DE
SIMULACIONES DE DINAMICA MOLECULAR

FALLA DE ORIGEN

T E S I S

QUE PARA OBTENER EL TITULO DE:

INGENIERO EN COMPUTACION

PRESENTA:

RODRIGO CORREA GARCIA

DIRECTOR DE TESIS:

DR. LUIS JAVIER ALVAREZ NOGUERA



MEXICO, D. F.

1995



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

DEDICATORIAS

Este trabajo está dedicado a mis padres Leonor García y Esteban Correa, a mis hermanos Germán, Marcela y Erika, a mis amigos de la preparatoria y de la facultad, y a mis familiares. A todos ellos como muestra de cariño por todo lo que me han dado y por lo que he aprendido de ellos.

AGRADECIMIENTOS

Agradezco sinceramente al Dr. Luis Javier Álvarez por todo el apoyo que me brindó para la realización de este trabajo, a Eduardo León por su valiosa ayuda en aspectos fisicoquímicos, a Carmen Valencia por la asesoría en diversos programas y paquetes de aplicación, y al equipo del laboratorio de visualización de la DGSCA por las facilidades que me brindaron para la utilización del equipo.

Durante la realización del trabajo que aquí se expone el autor contó con una beca del proyecto DGAPASC-000192 financiado por Cray Research, Inc. en el marco del convenio de apoyo a la investigación Cray-UNAM.

ÍNDICE

| | |
|--|-----------|
| 1. INTRODUCCIÓN | 1 |
| 2. PLANEACIÓN | 3 |
| 2.1 Definición del problema | 3 |
| 2.2 Objetivo | 3 |
| 2.3 Recursos..... | 3 |
| 2.3.1 Recursos Humanos | 3 |
| 2.3.2 Recursos de Programas..... | 3 |
| 2.3.3 Recursos de Equipo..... | 4 |
| 2.4 Justificación..... | 4 |
| 3. ANÁLISIS DEL SISTEMA..... | 6 |
| 3.1 Descripción del sistema | 6 |
| 3.2 Diagrama de flujo de datos | 7 |
| 3.3 Diccionario de datos..... | 8 |
| 3.4 Requerimientos del sistema..... | 9 |
| 3.5 Criterios de selección de requerimientos | 9 |
| 4. DISEÑO Y DESARROLLO DEL SISTEMA..... | 12 |
| 4.1 Herramientas y conceptos..... | 12 |
| 4.1.1 Bibliotecas gráficas "GL" de Silicon Graphics | 12 |
| 4.1.2 Bibliotecas <i>Forms</i> | 13 |
| 4.1.3 Funcionamiento y manejo del color | 14 |
| 4.2 Sistema de Visualización y Análisis de Simulaciones de Dinámica Molecular | 15 |
| 4.3 Módulos | 16 |
| 4.3.1 Dibujar átomos | 16 |
| 4.3.2 Buscar vecinos..... | 20 |
| 4.3.3 Dibujar conexiones | 22 |
| 4.3.4 Buscar Hoyos | 22 |
| 4.3.5 Dibujar hoyos | 29 |

| | |
|--|-----------|
| 4.3.6 Calcular volumen y densidad..... | 29 |
| 4.3.7 Calcular potencial electrostático..... | 31 |
| 4.3.8 Dibujar curvas de potencial..... | 35 |
| 5. FUNCIONAMIENTO Y RESULTADOS..... | 38 |
| 5.1 Procesos subsidiarios..... | 38 |
| 5.1.1 Buscar hoyos..... | 38 |
| 5.1.2 Calcular potencial electrostático..... | 39 |
| 5.1.3 Calcular volumen y densidad..... | 40 |
| 5.2 Opciones del sistema..... | 41 |
| 5.2.1 Abrir..... | 44 |
| 5.2.2 Ejes..... | 44 |
| 5.2.3 Caja..... | 44 |
| 5.2.4 Rotaciones..... | 45 |
| 5.2.5 Traslaciones..... | 49 |
| 5.2.6 Partículas..... | 50 |
| 5.2.7 Conexiones..... | 53 |
| 5.2.8 Hoyos..... | 55 |
| 5.2.9 Potenciales..... | 57 |
| 5.2.10 Calidad de esferas..... | 61 |
| 5.2.11 Grabar..... | 62 |
| 5.2.12 Salir..... | 63 |
| 6. APLICACIONES..... | 64 |
| 7. CONCLUSIONES..... | 67 |
| 8. BIBLIOGRAFÍA..... | 69 |

1. INTRODUCCIÓN

Con el advenimiento de las grandes computadoras digitales, en una gran cantidad de ramas de la ciencia, y en especial de las ciencias exactas (física, química, matemáticas, geofísica, ingeniería, etc.), se ha introducido una tercera manera de hacer investigación, complementaria a la teoría y a la experimentación. Ésta se conoce como simulación numérica.

La simulación está basada en el hecho de que una gran cantidad de fenómenos físicos y químicos se pueden "modelar". A partir de una teoría se construye un modelo matemático que describe las consecuencias de la teoría, es decir, predice y explica los fenómenos en los cuales la teoría en cuestión está basada.

El modelo normalmente está constituido por un conjunto de ecuaciones, por ejemplo, un modelo del sistema planetario de la tierra lo constituyen las ecuaciones clásicas del movimiento de los planetas, lunas y asteroides, junto con la ley de la gravitación universal. Resolviendo adecuadamente el conjunto de ecuaciones que constituyen el modelo, se pueden adquirir nuevos conocimientos acerca de los fenómenos asociados a él. Por ejemplo, resolviendo las ecuaciones clásicas del movimiento de los planetas se pueden predecir eclipses, conjunciones planetarias, colisiones de meteoritos con planetas, etc.

La solución de las ecuaciones del modelo, no siempre es posible obtenerla analíticamente, es por esta razón que se recurre a la solución numérica haciendo uso de computadoras con grandes recursos de almacenamiento y procesamiento de datos.

Una consecuencia necesaria del uso de grandes computadoras es la producción de grandes cantidades de información, sobre todo cuando se trata de resolver sistemas cuyas propiedades físicas se obtienen como resultado del cálculo estadístico. Para poder analizar e interpretar la información que se obtiene de la simulación de sistemas de este tipo, se recurre entonces a la visualización en estaciones de trabajo especializadas para este efecto.

Gracias a los avances tecnológicos de las estaciones de trabajo y a su gran capacidad gráfica, es posible convertir grandes cantidades de datos en imágenes. Las imágenes obtenidas visualizando resultados de una simulación numérica, en las actuales estaciones de trabajo de alto poder gráfico, permiten presentar los resultados no como una serie de cifras sino de forma gráfica adecuada para su análisis e interpretación.

Los métodos de la física estadística en conjunción con las grandes capacidades de cómputo que ofrece una supercomputadora, han permitido desarrollar métodos de simulación de materiales a nivel microscópico. Uno de estos métodos es el de dinámica molecular (DM), en el que se resuelven simultáneamente las ecuaciones clásicas del movimiento de N partículas que representan átomos de diferentes especies dentro de un volumen V contenido en una caja imaginaria llamada "caja computacional". Éstos interactúan según alguna ley de interacción dada por un potencial. En el método de dinámica molecular las propiedades termodinámicas y estructurales del sistema bajo estudio se calculan a partir de promedios estadísticos en el tiempo. Algunas cantidades obtenidas se pueden representar en forma de gráficas x vs y , sin embargo, otras propiedades, sobre todo estructurales y sus relaciones con otras cantidades físicas, es necesario representarlas directamente, como se verían con un microscopio imaginario. En esta tesis se presenta un sistema especializado de visualización y análisis de simulaciones de dinámica molecular, desarrollado en una estación de trabajo gráfica en lenguaje "C" que utiliza las bibliotecas gráficas de Silicon Graphics. El sistema de simulación de dinámica molecular fue desarrollado previamente en el Laboratorio de Simulación de Materiales de la Dirección General de Servicios de Cómputo Académico de la UNAM.

2. PLANEACIÓN

2.1 Definición del problema

Debido a la producción masiva de datos en investigaciones numéricas, se convierte en necesidad la utilización de herramientas que simplifiquen su interpretación. Ya que los fenómenos que se estudian con el método de dinámica molecular están enfocados a materiales a nivel microscópico, la herramienta que mejor resuelve el problema de interpretación es la visualización. En consecuencia, una parte muy importante que debe integrarse al sistema de simulación es el de visualización.

2.2 Objetivo

El objetivo de este trabajo es el de desarrollar un sistema que facilite el análisis de resultados de simulaciones de dinámica molecular, haciendo de la visualización una herramienta muy poderosa para el estudio de materiales a nivel microscópico.

2.3 Recursos

2.3.1 Recursos Humanos

Para la elaboración de este sistema, por ser un trabajo de tesis de licenciatura, todas las etapas de desarrollo del mismo están a cargo de una sola persona.

Así mismo, es indispensable el apoyo de investigadores en ramas de la física y la química puesto que son los interesados en estudiar y observar procesos fisicoquímicos y estructurales en los materiales de interés. Estos procesos pueden involucrar fenómenos como cambios de fase, cambio en el radio y carga de los átomos, formación de cavidades en las estructuras, fenómenos de superficie, entre otros. La determinación y observación de todas estas características de los materiales está siempre supervisado por los investigadores.

2.3.2 Recursos de Programas

La DGSCA ha adquirido programas comerciales de visualización de propósito general para estaciones de trabajo como MPGS (del inglés *Multi Purpose Graphic System*), especializados en cálculos cuánticos como Unichem y bibliotecas de graficación como *DORÉ* de TITAN, en donde

se comenzó a desarrollar el sistema que se describe en esta tesis. Estas aplicaciones han servido para que el desarrollo del que trata esta tesis aproveche ideas que se han utilizado con éxito en los paquetes y programas mencionados.

Las estaciones de trabajo en las cuales se desarrolló este sistema cuentan con un sistema operativo UNIX, con el compilador estándar de "C", el compilador de FORTRAN y una gran cantidad de subrutinas de bibliotecas gráficas. También se cuenta con un poderoso conjunto de herramientas de archivo y acabado de imágenes tanto para impresión como para video.

2.3.3 Recursos de Equipo

En la DGSCA se cuenta con el equipo necesario para el desarrollo de esta tesis. Posee la supercomputadora, en donde además de realizarse las simulaciones de la DM, se tiene la opción de ejecutar módulos subsidiarios del sistema de visualización.

Como el sistema necesita manejar grandes cantidades de datos y desplegar gráficas con alta definición, es necesario el uso de una estación de trabajo cuya capacidad de memoria tolere un gran volumen de información y al mismo tiempo que tenga alta capacidad gráfica. El Laboratorio de Simulación de Materiales de la dependencia cuenta con dos estaciones de trabajo *Indigo 2* de *Silicon Graphics* que tienen un excelente manejo de imágenes por medio de un conjunto de circuitos especializados, además de tres pequeñas estaciones de trabajo *Indy*. Por supuesto, también se cuenta con la red de comunicación necesaria para poder transferir datos de la supercomputadora a las estaciones de trabajo y viceversa.

En cuanto a equipo de impresión se refiere, la DGSCA cuenta con impresoras a color, de inyección de tinta, láser y de proceso fotográfico de muy alta calidad, además de un sofisticado equipo de edición profesional de video.

2.4 Justificación

La UNAM realiza una gran cantidad de investigaciones de todo tipo, que constituyen aproximadamente el 80 % de las investigaciones de todo el país. De este porcentaje, las investigaciones por medio de algoritmos que simulan el comportamiento de la naturaleza, han cobrado gran importancia con la adquisición de la supercomputadora CRAY-YMP por parte de la UNAM.

Los resultados de las simulaciones numéricas, en el caso que nos ocupa, con el método de dinámica molecular, suelen ser archivos de datos gigantescos, que para ser analizados tendrían que pasar por una serie de programas o filtros que generan archivos más pequeños, lo que trae como consecuencia un uso excesivo del espacio en disco, pérdida de tiempo y de información.

Por otro lado, ciertos fenómenos que ocurren en una estructura molecular, no se pueden observar en un laboratorio experimental. Sin embargo, con una simulación numérica y un sistema de visualización es posible "ver" todas las etapas de un experimento numérico. Debido a esto es necesario tener una herramienta que permita analizar de una manera más fácil y rápida los resultados obtenidos en las experimentaciones numéricas. Para poder analizar los resultados de simulaciones de fenómenos complejos en la supercomputadora, es necesario contar con una herramienta especializada de visualización, sin ella, este análisis sería tardado y engorroso.

Por otra parte, el desarrollo de una herramienta propia representa un ahorro al no comprar sistemas existentes en el mercado cuyos precios son muy elevados, están en otro idioma o mal traducidos, y sus aplicaciones no son tan flexibles. Además constituye una oportunidad para formar recursos humanos capaces de desarrollar sistemas especializados.

3. ANÁLISIS DEL SISTEMA

3.1 Descripción del sistema

Consiste en la transformación de los datos, producto de una experimentación numérica de dinámica molecular en la supercomputadora, en imágenes que permitan observar fenómenos microscópicos de la estructura de los materiales. Esta transformación puede ser directamente de la supercomputadora a la pantalla de la estación de trabajo. En muchos casos se requiere reprocesar el resultado de la simulación para obtener nuevas características que de igual forma se representan gráficamente.

3.2 Diagrama de flujo de datos

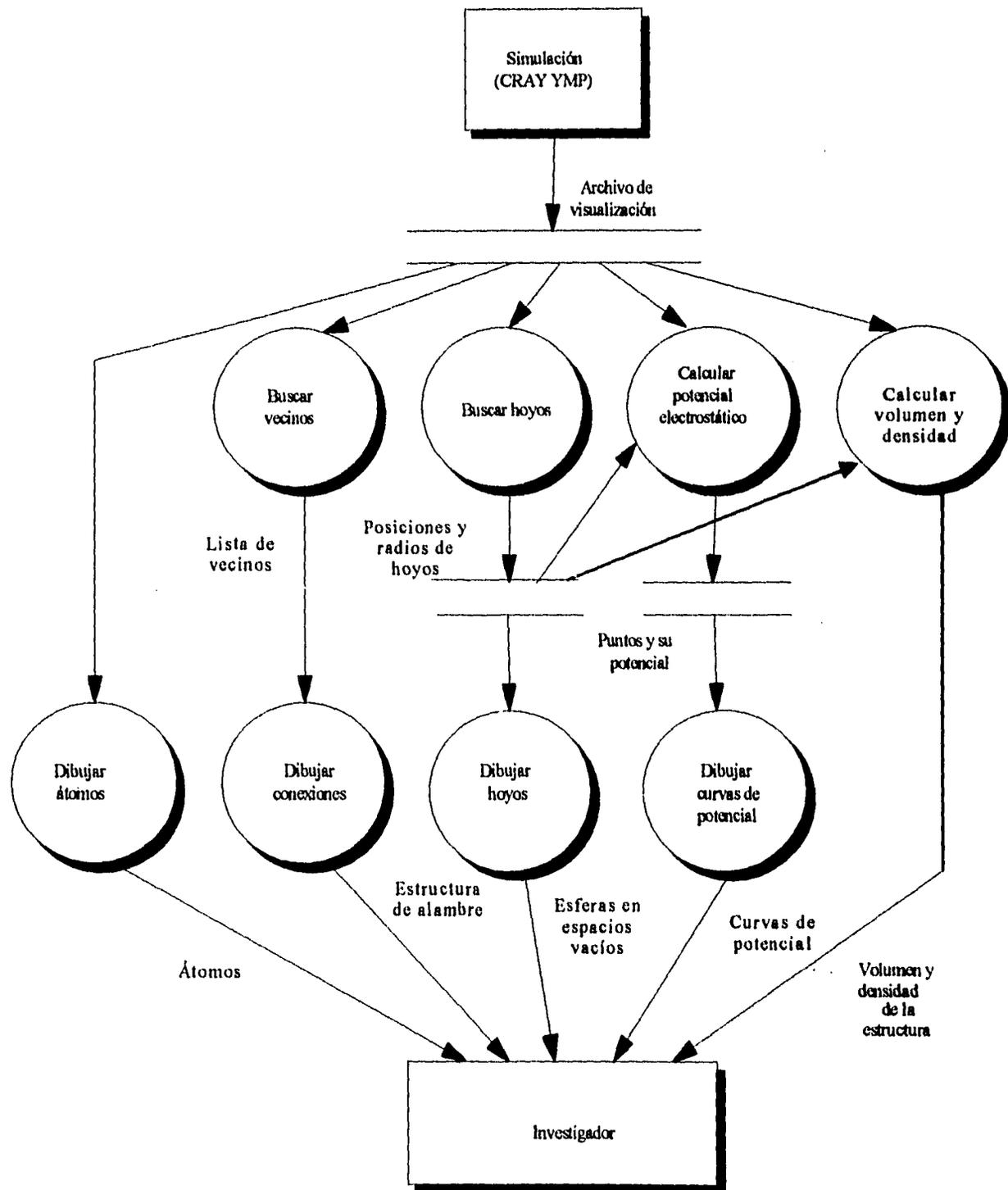


Figura 3.1 Diagrama de flujo de datos del sistema.

3.3 Diccionario de datos

- Nombre Archivo de visualización.
Dónde se usa . Dibujar átomos
 Buscar vecinos
 Buscar hoyos
 Calcular potencial electrostático
 Calcular volumen y densidad.
Derivado de ... Simulación numérica en la supercomputadora.
Propósito Proporcionar la información necesaria para obtener las características gráficas de las partículas.
 Proporcionar información para calcular distancias y vecindades.
 Dar información acerca del espacio vacío de la estructura.
 Brindar información para el cálculo del potencial electrostático.
 Dar información acerca de la densidad de la estructura.
- Nombre Lista de vecinos.
Dónde se usa . Dibujar conexiones.
Derivado de ... Buscar vecinos.
Propósito Indicar, para cada partícula, el conjunto de partículas que se encuentran dentro de una esfera de radio dado.
- Nombre Posiciones y radios de hoyos.
Dónde se usa . Dibujar hoyos
 Calcular potencial electrostático.
Derivado de ... Buscar hoyos
Propósito Proporcionar la ubicación y tamaño de los hoyos para ser dibujados.
 Transferir información sobre el espacio donde se pretende calcular el potencial electrostático.
 Indicar el espacio libre de la estructura.
- Nombre Puntos y su potencial.
Dónde se usa . Dibujar curvas de potencial.
Derivado de ... Calcular potencial electrostático.
Propósito Proporcionar el valor del potencial electrostático en los puntos de la estructura.
- Nombre Volumen y densidad
Derivado de ... Calcular volumen.
Propósito Mostrar tanto el volumen como la densidad de la estructura.
- Nombre Representación de los átomos.
Derivado de ... Dibujar átomos.
Propósito Visualizar partículas.
- Nombre Estructura de alambre.

Derivado de ... Dibujar conexiones.
Propósito Visualizar, por medio de conectores, la coordinación de las partículas.

Nombre Representación de esferas en el espacio vacío.
Derivado de ... Dibujar hoyos.
Propósito Visualizar sistemas de poros.

Nombre Curvas de potencial.
Derivado de ... Dibujar curvas de potencial.
Propósito Visualizar zonas de interacción electrostática.

3.4 Requerimientos del sistema

- Estación de trabajo *Silicon Graphics*
- Monitor de alta resolución de 20" con resolución de 1280 * 1024 pixeles
- Ratón
- Memoria principal de 32 Mbytes
- Sistema operativo IRIX 4.5.0 (versión de UNIX) o alguna versión más reciente
- Espacio en disco duro superior a 400 Mbytes
- Caja de perillas de control (opcional)
- Compilador de "C"
- Bibliotecas gráficas FORMS

3.5 Criterios de selección de requerimientos

Como se mencionó anteriormente, debido al volumen de información que se maneja en la experimentación numérica y a la necesidad de observar los resultados con alta definición, se optó por desarrollar el sistema en una estación de trabajo *Silicon Graphics*, ya que son las máquinas con mejor manejo de gráficos en el mercado y porque el Laboratorio de Simulación de Materiales de la DGSCA cuenta con un número considerable de estas estaciones de trabajo. De hecho, toda la visualización que se realiza en la dependencia, se lleva a cabo con este tipo de máquinas.

En virtud de que el tipo de estación de trabajo seleccionada cuenta con periféricos especializados para la manipulación de imágenes, se utiliza la caja de perillas que es un dispositivo de entrada que consta de un conjunto de potenciómetros que reportan un valor, entre un intervalo

predeterminado, a partir de su posición inicial. La caja tiene ocho perillas numeradas de la siguiente forma:

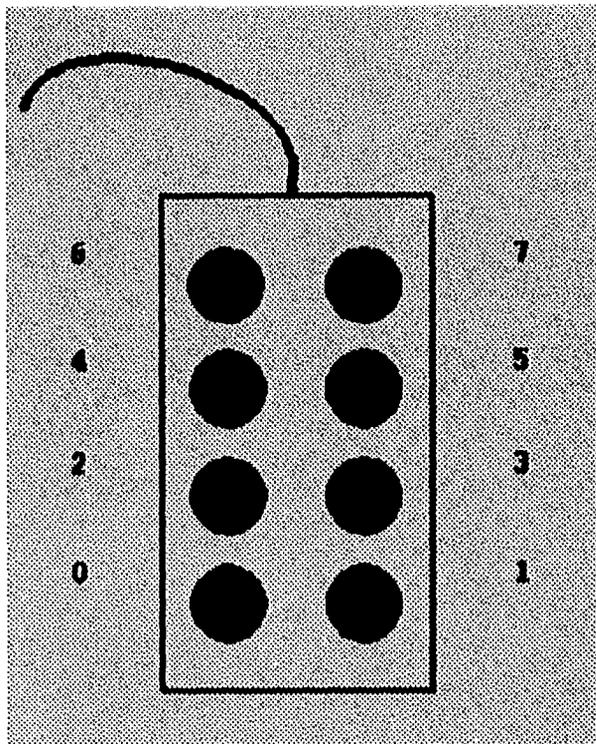


Figura 3.2 Caja de perillas.

Con este dispositivo se puede manipular la estructura que se visualiza, asignándole una tarea específica a cada perilla, con lo cual se obtiene un control que no depende ni del teclado, ni del ratón.

La elección del lenguaje "C" para el desarrollo de este sistema, obedece a la facilidad que brinda para el manejo de la memoria dinámica, pudiendo manejar estructuras de datos sofisticadas, así como el rico conjunto de operadores para la manipulación de los datos. Un factor decisivo en la inclinación hacia "C", fue que ya se contaba con el compilador y pensar en otro lenguaje implicaba una adquisición innecesaria. Este compilador contiene un conjunto de bibliotecas gráficas muy poderoso que se incluyen en los programas de forma sencilla.

Las interfaces con el usuario como los menús con los que se pueden efectuar elecciones, cajas de diálogos que le permiten al usuario introducir datos de entrada, botones que el usuario puede oprimir con el ratón, ventanas de información, etc., permiten un manejo más sencillo del sistema. Las bibliotecas FORMS fueron diseñadas para que la construcción de estas interfaces

fuera más sencilla y no consumiera tanto tiempo en el desarrollo del sistema. Estas rutinas de dominio público, fueron desarrolladas para estaciones *Silicon Graphics* en el lenguaje "C", por lo que pueden ser utilizadas por programas en "C" y "C++" directamente.

4. DISEÑO Y DESARROLLO DEL SISTEMA

Antes de pasar a la descripción de los procesos del sistema es necesario, en aras de la claridad, describir las características esenciales de los sistemas que se visualizan. Éstos están constituidos por átomos y sus conexiones. Los átomos se representan gráficamente con esferas cuyos radios y colores se determinan de acuerdo con sus características químicas como peso atómico, radio iónico y carga efectiva. Las conexiones entre los átomos se representan gráficamente por prismas hexagonales, cuya longitud se especifica de acuerdo con las propiedades químicas del sistema simulado. Los espacios vacíos entre partículas se representan con esferas blancas que pueden hacerse transparentes y que en el contexto de este trabajo se llaman "hoyos". Tanto las partículas y conexiones, como los hoyos están, como se mencionó anteriormente, contenidos en una caja, que puede ser cúbica o romboédrica, a la que en el programa se le llama "caja". Todos estos elementos de los sistemas simulados se representan y manejan como objetos, ya que las bibliotecas gráficas utilizadas están orientadas a objeto. Una información importante para el análisis estructural y de reactividad química en los sistemas simulados, es la conformación espacial del potencial eléctrico que se establece en la estructura y especialmente dentro de las cavidades representadas por los hoyos. En adelante se hará referencia a cada una de estos objetos y características.

4.1 Herramientas y conceptos

Con el objeto de explicar el diseño y desarrollo de cada uno de los procesos que conforman el sistema, es necesario hacer una breve descripción de las herramientas que llevan a cabo algunos procesos, y de los conceptos en los cuales se basa el manejo de ciertas características gráficas.

4.1.1 Bibliotecas gráficas "GL" de Silicon Graphics

Las bibliotecas gráficas *GL* son un conjunto de rutinas que proveen soporte, en alto y bajo niveles, para el manejo de gráficos en estaciones de trabajo *Silicon Graphics*. Una de las grandes ventajas del uso de estas rutinas es que al abrir la ventana o área de dibujo, se definen de manera

sencilla las transformaciones de proyección que mapean el sistema de 3 dimensiones definido en coordenadas cartesianas de punto flotante, o en otras palabras tripletas (x,y,z) , a coordenadas de pantalla. Esto significa que tanto el punto de vista del observador, como la perspectiva de la imagen, se definen por el programador en función de ángulos y distancias en el sistema cartesiano, entendible y fácil de manejar, e internamente se construye una matriz de transformación que opera sobre las coordenadas cartesianas para obtener los correspondientes valores en pixeles. Esta matriz y las operaciones que realiza, a menos que el programador desee obtener su valor y trabajar a un nivel más bajo, son como una caja negra que simplifica el manejo de los objetos que se despliegan en pantalla. Por ejemplo, una operación de rotación de 90° alrededor del eje x sobre los objetos en pantalla, se reduce a la instrucción **rot (90, "x")**.

Entre las rutinas gráficas que se utilizan en el sistema, se encuentran las que permiten el manejo de unidades externas, como el ratón y las perillas. Éstas permiten una comunicación fluida con las unidades de entrada y a la vez se tiene la facilidad de delimitar los valores que puede tomar una unidad. Por ejemplo, en el caso de las perillas de control que pueden ser utilizadas para indicar diferentes niveles de transparencia de los hoyos, se inicializa la perilla con un valor de cero que significa que la esfera que representa a un hoyo no tenga transparencia, y una vuelta completa de la perilla corresponde al valor máximo equivalente a mayor transparencia.

Otro conjunto de rutinas gráficas importantes, son aquellas con las que se pueden asignar ciertas características a un elemento gráfico, con lo cual se crean los llamados "materiales", es decir, que a un elemento gráfico, como por ejemplo una esfera, se le pueden atribuir ciertas características de brillo, reflexión, transparencia, centro de luminosidad, etc.

4.1.2 Bibliotecas *Forms*

Las interfaces de usuario son muy importantes para facilitar la comunicación con el sistema. Generalmente, las aplicaciones que permiten construir las interfaces gráficas de usuario tienen un costo elevado y consumen muchos recursos. Por esta razón se eligió el conjunto de rutinas de dominio público llamadas *Forms*. Éstas son de uso sencillo y con una excelente presentación gráfica, con lo cual se facilita la comprensión del sistema y sus procesos.

Una forma de las rutinas *Forms* es una ventana donde se colocan diferentes objetos. Existen muchos tipos de objetos, como por ejemplo, botones que el usuario puede oprimir con el

ratón, barras con las cuales se selecciona un valor determinado, campos en los cuales el usuario puede escribir textos de entrada, menús en donde se realizan selecciones, mensajes de ayuda, etc. Cada vez que el usuario cambia el estado del objeto en particular en una de las formas desplegadas, el sistema es notificado para que se ejecute la acción correspondiente.

4.1.3 Funcionamiento y manejo del color

Un monitor convencional tiene tres cañones de color que barren 60 veces por segundo el área de la pantalla. Durante este barrido, cada cañón apunta directamente a cada uno de los pixeles en un tiempo muy corto. Los cañones disparan electrones que chocan en la pantalla provocando que ésta brille.

Cada pixel en la pantalla está compuesto por tres diferentes tipos de fósforo que se encienden en rojo, verde o azul. Un cañón activa sólo los fósforos rojos, otro los verdes, y el tercero los azules. Como cada cañón recorre los pixeles, el número de electrones disparados (la intensidad) es modificado pixel por pixel. En otras palabras, considerando sólo el cañón de color rojo, si los electrones no han sido disparados al pixel, sus fósforos no brillan, y se ve negro. Si el cañón dispara con su mayor intensidad, el fósforo se enciende en rojo brillante. Las intensidades intermedias provocan que el color varíe entre negro y rojo brillante. Lo mismo sucede para los otros cañones, excepto que los colores varían de negro a verde brillante, o de negro a azul brillante. El color que el ojo percibe es la combinación de los tres colores. Las diferentes intensidades de disparo generan una amplia variedad de colores.

Cada cañón puede ser colocado en 256 niveles diferentes de intensidad, desde el completamente apagado (0) hasta el completamente encendido (255). Las intensidades de los cañones rojo, verde y azul determinan el color del pixel. Esto es expresado como una terna *RGB* (del inglés *Red, Green, Blue*), es decir, tres números entre 0 y 255 indicando las intensidades de cada color.

Así por ejemplo, el negro se representa por (0,0,0), el rojo brillante por (255,0,0), y así (255,255,0) = amarillo (fig. 4.1), (0,255,255) = turquesa, (255,0,255) = morado rojizo, (255,255,255) = blanco, etc. Los diferentes tonos de gris se obtienen colocando las tres componentes con el mismo valor. Como cada cañón tiene 256 valores posibles, existen $256*256*256 = 16'777,216$ colores diferentes.

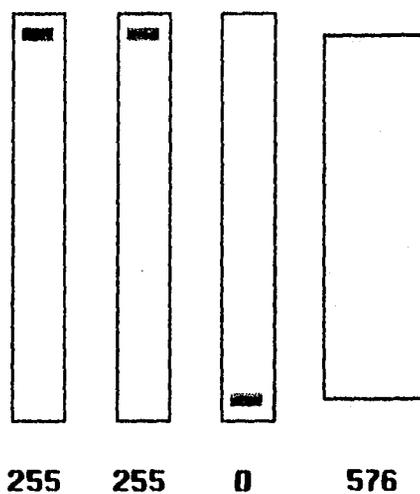


Figura 4.1 Formación del color.

4.2 Sistema de Visualización y Análisis de Simulaciones de Dinámica Molecular

El proceso de visualización y análisis de una simulación comienza cuando el cálculo numérico en la supercomputadora termina, obteniéndose un conjunto de archivos de datos con información relevante del sistema simulado.

Una simulación de DM que se ejecuta en la supercomputadora de la UNAM se tarda de una a 20 horas de tiempo de procesador dependiendo de las características de la misma. El volumen de información que arroja una corrida de DM puede llegar a varios cientos de Mbytes. De los aproximadamente 20 archivos generados por la simulación, el de interés para el sistema de visualización contiene la información de la posición de cada átomo y su carga. Este archivo se transfiere de la supercomputadora a la estación de trabajo donde es utilizado en varios procesos para obtener nueva información y ser representada gráficamente.

El sistema tiene un diseño modular, lo cual permite escoger los procesos que se quieren realizar y en un futuro incrementar las funciones del sistema agregando nuevos módulos, ya que el compilador permite ligar nuevos archivos objeto sin necesidad de recompilar todos los módulos ya existentes.

Los módulos que realizan las funciones de búsqueda de hoyos, cálculo de potenciales y cálculo de volumen y densidad, son módulos intermedios, es decir, que generan nueva

información. Esto es de gran ventaja, ya que permite la transportabilidad de dichos procesos a la supercomputadora, aprovechando las herramientas de optimización en los procesos vectoriales con la consecuente reducción del tiempo de generación de los archivos y restando sólo la tarea de transmitirlos a la estación de trabajo para su visualización. Sin embargo, estos módulos sólo se aplican a estructuras estáticas, es decir, que el archivo de visualización representa el estado de la estructura únicamente en un tiempo dado o bien una estructura promedio.

La tarea de analizar la información se convierte en una combinación de los módulos que permiten visualizar una estructura. A continuación se describe cada uno de estos módulos.

4.3 Módulos

4.3.1 Dibujar átomos

El objetivo de este módulo es transformar el archivo de posiciones de los átomos en un elemento gráfico que nos permita verlos. Es por esto que se hace uso de esferas como elemento gráfico para representar a los átomos en la pantalla.

El primer proceso que se realiza en este módulo es el de lectura del "archivo de visualización" arrojado por el programa de la dinámica molecular. Éste es un archivo en código ASCII lo cual permite hacer la transferencia vía la red sin problemas. Otra ventaja de que el archivo de visualización sea ASCII es que se puede manipular fácilmente y seleccionar sólo lo que se quiere visualizar, por ejemplo, se pueden eliminar partículas del archivo o incluir otras para referencia o futuros cálculos.

En el archivo de visualización, cuyo formato se puede consultar en el capítulo 5, se incluyen los valores de las siguientes variables:

TITULO, cadena de caracteres identificadora de la estructura que se visualiza. Ésta es necesaria ya que se manejan muchos archivos de visualización y el nombre del archivo resulta insuficiente para poder saber las condiciones bajo las cuales se obtuvo el resultado.

Las variables reales **boxx**, **boxy** y **boxz**, definen el tamaño de la caja computacional y son utilizadas para inicializar ciertas variables que permiten ver a una distancia razonable toda la estructura, es decir, que con estas variables se define la matriz de transformación que va a mapear las coordenadas cartesianas a coordenadas de pantalla.

gama indica el ángulo de inclinación que tiene las paredes laterales de la caja con respecto al eje x y se utiliza para definir las características de las cajas romboédricas.

Con la DM se obtienen las posiciones de los átomos en un instante de tiempo T . Pero si se toma un reporte de dichas posiciones en varios instantes consecutivos de tiempo y se despliegan en pantalla, entonces se puede observar la evolución temporal de las partículas durante el tiempo simulado, es decir que se verá una animación del comportamiento de la estructura. Un número entero, NC , indica cuántas muestras fueron tomadas en el tiempo para generar el archivo, es decir, indica el número de cuadros de la "filmación" del proceso.

El número de átomos en cada cuadro no es constante, ya que probablemente el archivo de visualización contiene sólo una parte de una gran estructura. Esto sucede cuando dentro de la DM se centra una esfera de un determinado radio de corte, lo que implica que al estar en movimiento los átomos, entran y salen constantemente de la esfera, con lo cual varía el número de partículas por cuadro. Por esto es necesario indicar al inicio de cada cuadro el número de átomos, NA , que contiene.

La etiqueta, e , es un entero identificador único de cada partícula del archivo de visualización. Esto se hace con el fin de distinguir a los átomos de la misma especie y así poder seguir el comportamiento de un átomo en particular.

x, y, z son las coordenadas en el espacio de los átomos dentro de la caja computacional.

La carga, q , de las partículas se representa mediante el color de las esferas. Se utiliza un valor para hacer una correspondencia entre ésta y una gama de colores predefinida. Las unidades de la carga no son importantes ya que el color se obtiene mediante una interpolación entre los valores extremos de las cargas para cada átomo.

El tipo, t , es un entero que indica el número atómico de la partícula.

Las unidades de longitud de la caja así como las de las posiciones de los átomos deben ser angstroms (\AA). Las coordenadas de las partículas tienen como referencia el centro de la caja computacional, es decir, que las coordenadas $(0,0,0)$ equivalen al centro de la caja.

Los archivos que no están en este formato no son válidos para el sistema, produciendo un aviso de error de formato para el usuario.

```

    "In", 0.94, 100, -100, 3, 0, 180, 0, 100, 180,
    "Sn", 0.83, 100, -100, 3, 255, 0, 120, 35, 0,
    "Sb", 0.94, 100, -100, 1, 150, 100, 200, 80, 20,
    "Te", 0.66, 100, -100, 2, 50, 0, 250, 0, 250,
    "I ", 0.58, 100, -100, 3, 255, 0, 120, 35, 0,
    "Xe", 0.62, 100, -100, 2, 100, 60, 0, 60, 0,
    "Cs", 1.81, 100, -100, 2, 50, 0, 250, 0, 250,
    "Ba", 1.52, 100, -100, 1, 150, 100, 200, 80, 20,
    "La", 1.20, 100, -100, 3, 255, 0, 120, 35, 0 );

```

donde nombre es una cadena de 2 caracteres que definen el símbolo químico de cada átomo. El radio se especifica mediante una cifra en angstroms. `qmax` y `qmin` son los valores de carga extremos que se obtienen al leer el archivo de visualización para cada tipo de átomo. `colorvar` el color que se va a variar (1 rojo, 2 verde y 3 azul), `colormax` y `colormin` son los valores que delimitan el intervalo en el cual varía `colorvar`, y `colorr`, `colorg` y `colorb` que son las constantes de inicio del color de los átomos. Cabe hacer notar que los valores de los elementos faltantes, así como elementos que tienen sus valores en cero, no han sido colocados ya que se han averiguando conforme se han ido necesitando los elementos.

Después de leído el archivo de visualización, el sistema dibuja las partículas. Por cada partícula se dibuja una esfera centrada en las coordenadas señaladas por `x`, `y`, `z` y con un radio que se obtiene en función de su número atómico. El color de la esfera se establece en función de la carga de la partícula.

Para obtener el color de la esfera, se hace una interpolación entre los límites de colores marcados en la estructura que contiene las características de las partículas, luego este valor es asignado a la componente de la terna `rgb` indicada por `colorvar`. Por ejemplo, si `colorvar` es 1 entonces las componentes `colorg` y `colorb` quedarán como constantes, y el valor de `colorr` variará entre las constantes `Cmax` y `Cmin` en función de su carga. Una forma simple de explicar cómo se obtiene el valor del color variable es por medio de la siguiente gráfica.

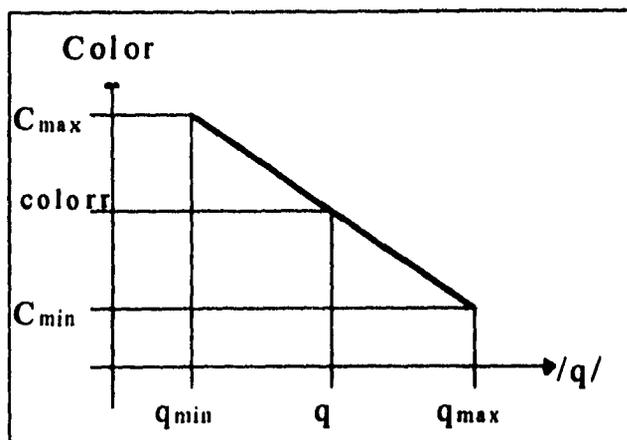


Figura 4.2 Obtención del color variable.

De esta manera, la componente `color_r` se obtendrá por medio de:

$$color_r = \frac{(C_{max} - C_{min}) * (q_{max} - q)}{(q_{max} - q_{min})}$$

El propósito de variar sólo una de las componentes de la terna `rgb` es que las esferas de una misma especie se vean más brillantes o más opacas pero que no tengan un cambio brusco de color, puesto que se confundirían con otras especies de partículas. De esta manera la partícula con más carga (positiva o negativa) tendrá una tonalidad más oscura que una con menor carga.

Obtenidas todas las características de la esfera (posición, radio y color) se dibuja por medio de una función de la biblioteca gráfica, la cual nos permite dibujar las esferas con la definición que queramos. Este procedimiento se repite para todas las partículas y para todos los cuadros de la simulación.

4.3.2 Buscar vecinos

Este módulo genera nueva información a partir del archivo de visualización. Sin embargo, a diferencia de los otros módulos intermedios (hoyos, potenciales y volumen), éste es interactivo y sólo se ejecuta en la estación de trabajo durante el proceso de visualización. Esto es debido a que, por ejemplo, para una estructura promedio sólo existe un conjunto de hoyos que definen su espacio vacío, un conjunto de puntos de potencial electrostático y un volumen. Pero los vecinos

de los átomos varían según el radio de conexión que se desee. El diagrama de flujo de éste proceso se muestra a continuación.

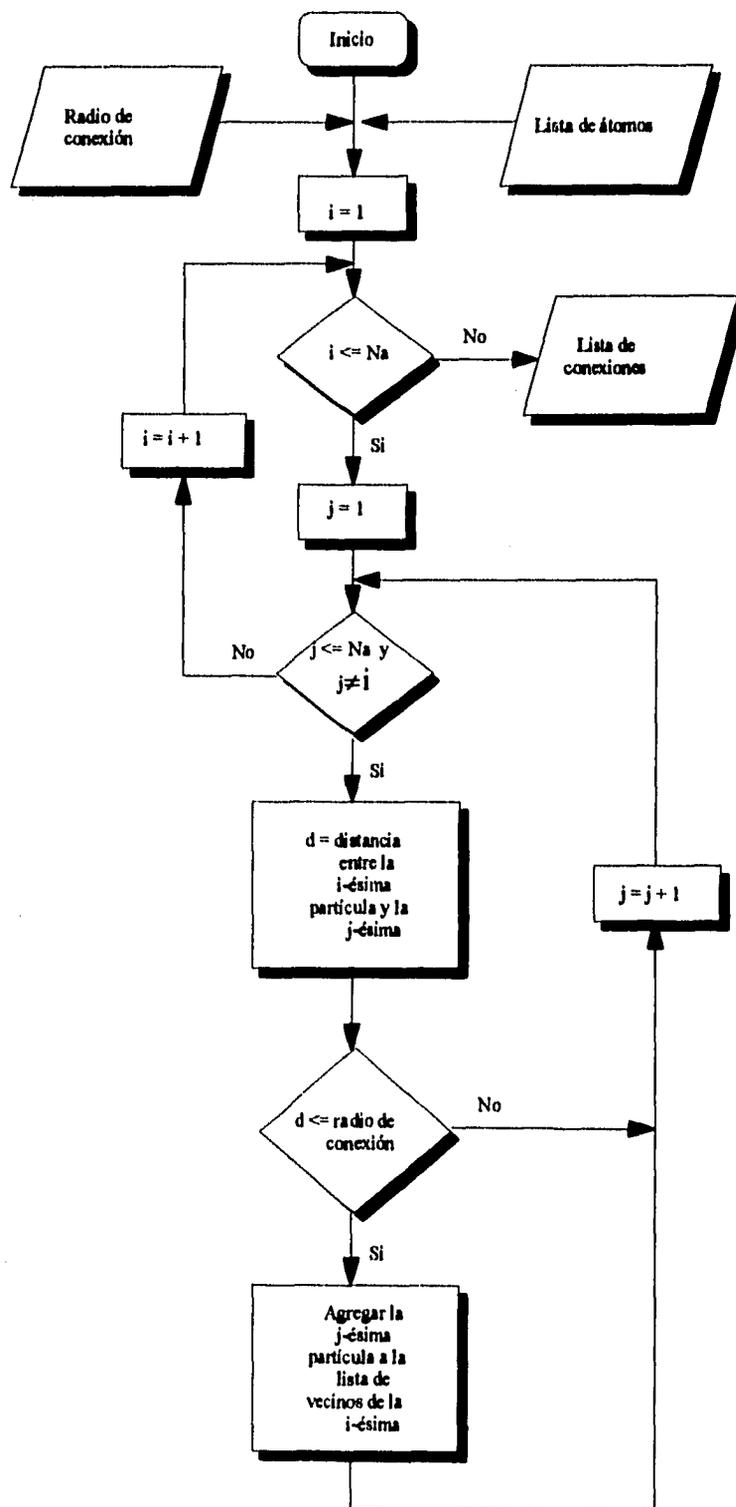


Figura 4.3 Diagrama de flujo del módulo buscar vecinos.

El funcionamiento consiste en posicionarse en cada una de las partículas y en base a una distancia dada, llamada radio de conexión, comparar la distancia con el resto de las partículas de la estructura. De esta comparación, se seleccionan las que se encuentran a una distancia menor o igual al radio de conexión y se forma una lista dinámica para cada partícula. En la lista no se almacenan las posiciones de los vecinos por ser ésta una información redundante, y contiene únicamente las etiquetas de los átomos vecinos.

4.3.3 Dibujar conexiones

Como parte importante del proceso de visualización, se encuentra la capacidad de ver las partículas conectadas entre sí en función de la distancia de enlace químico. De esta manera, podemos ver la “estructura de alambre” que provee valiosa información.

Gracias a este módulo, se puede observar la coordinación de ciertas partículas de interés, las cavidades, verificar distancias, etc. El proceso consiste en tomar cada elemento de la lista y la etiqueta de cada uno de sus vecinos. Conociendo la etiqueta de una partícula y las de sus vecinos, se obtienen las coordenadas, con lo que se procede a dibujar un elemento que conecte cada par de partículas. Dicho elemento es un prisma de color oscuro con base hexagonal el cual produce un efecto agradable en la visualización.

4.3.4 Buscar Hoyos

Frecuentemente en las simulaciones de DM es necesario analizar, no sólo la estructura constituida por átomos y moléculas, sino también el espacio vacío que constituye la porosidad o los espacios intersticiales. Se pueden efectuar diferentes cálculos basados en un conjunto de “hoyos”, como el potencial eléctrico dentro de ellos, su distribución de tamaño, el volumen libre, y la estructura de las cavidades, los cuales son de gran importancia para la comprensión de los fenómenos fisicoquímicos tales como la difusión.

El método para encontrar el conjunto de hoyos de una estructura, se basa en una sucesión de puntos aleatorios uniformemente distribuidos en el espacio, que es generada dentro de la caja computacional. Una esfera es centrada en cada punto y su radio toma el valor de la distancia entre su centro y la superficie de la partícula más cercana. Si la nueva esfera que representa un hoyo está

contenida en otro hoyo entonces se descarta, y si previamente existía un hoyo contenido en la nueva esfera entonces este último se elimina, si ninguno de los casos anteriores sucede, se añade a la colección un nuevo hoyo y se actualiza. El criterio de contención empleado es el siguiente: Dados dos hoyos A y B, A está contenido en B si $d_{AB} < r_B$ y $r_A < r_B$, como se muestra en la figura 4.4.

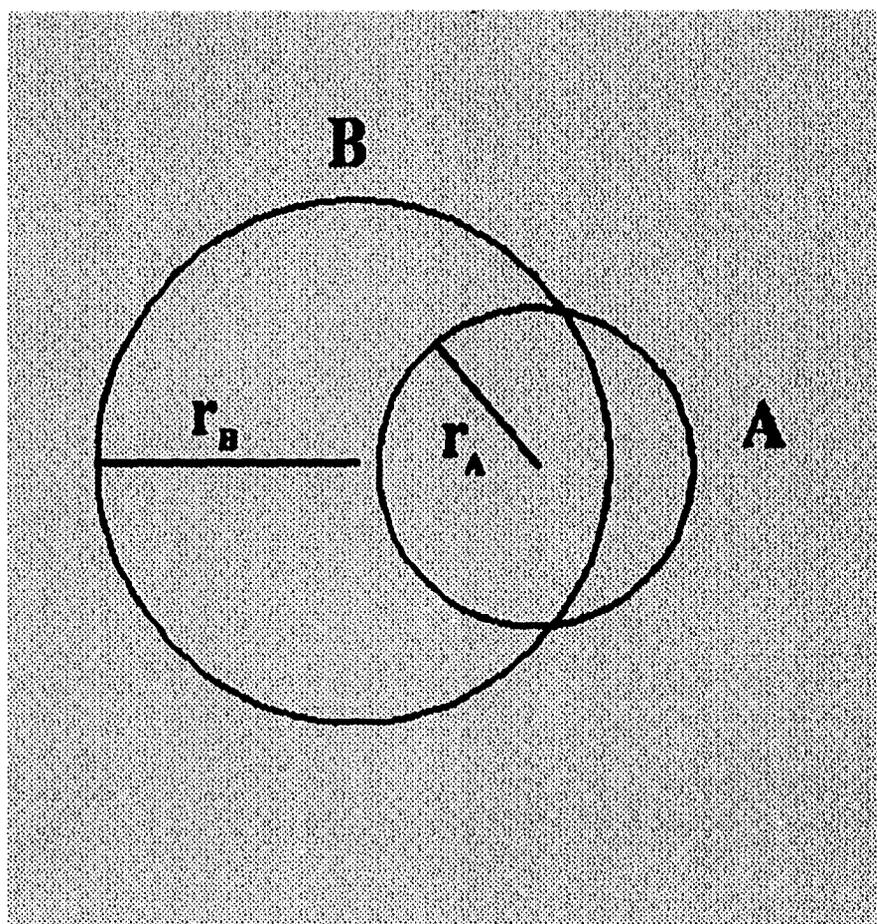


Figura 4.4 Criterio de contención de hoyos.

La generación de puntos aleatorios en el espacio, que cubren densamente el volumen de la caja, se realiza tomando tres números aleatorios sucesivos uniformemente distribuidos sobre el intervalo $[-L, L]$ donde L es el lado de la caja computacional. Si los números aleatorios son independientes, entonces los puntos del espacio constituidos por las triadas así formadas, cubren densamente el volumen de la caja que contiene a las partículas para un número suficientemente grande de triadas. En la figura 4.5 se presenta una gráfica con 1000 puntos aleatorios en tres

dimensiones y sus proyecciones sobre los planos $x=0$, $y=0$ y $z=0$ generados con la función $\text{rand}(\)$ de la biblioteca matemática de "C" de CRAY.

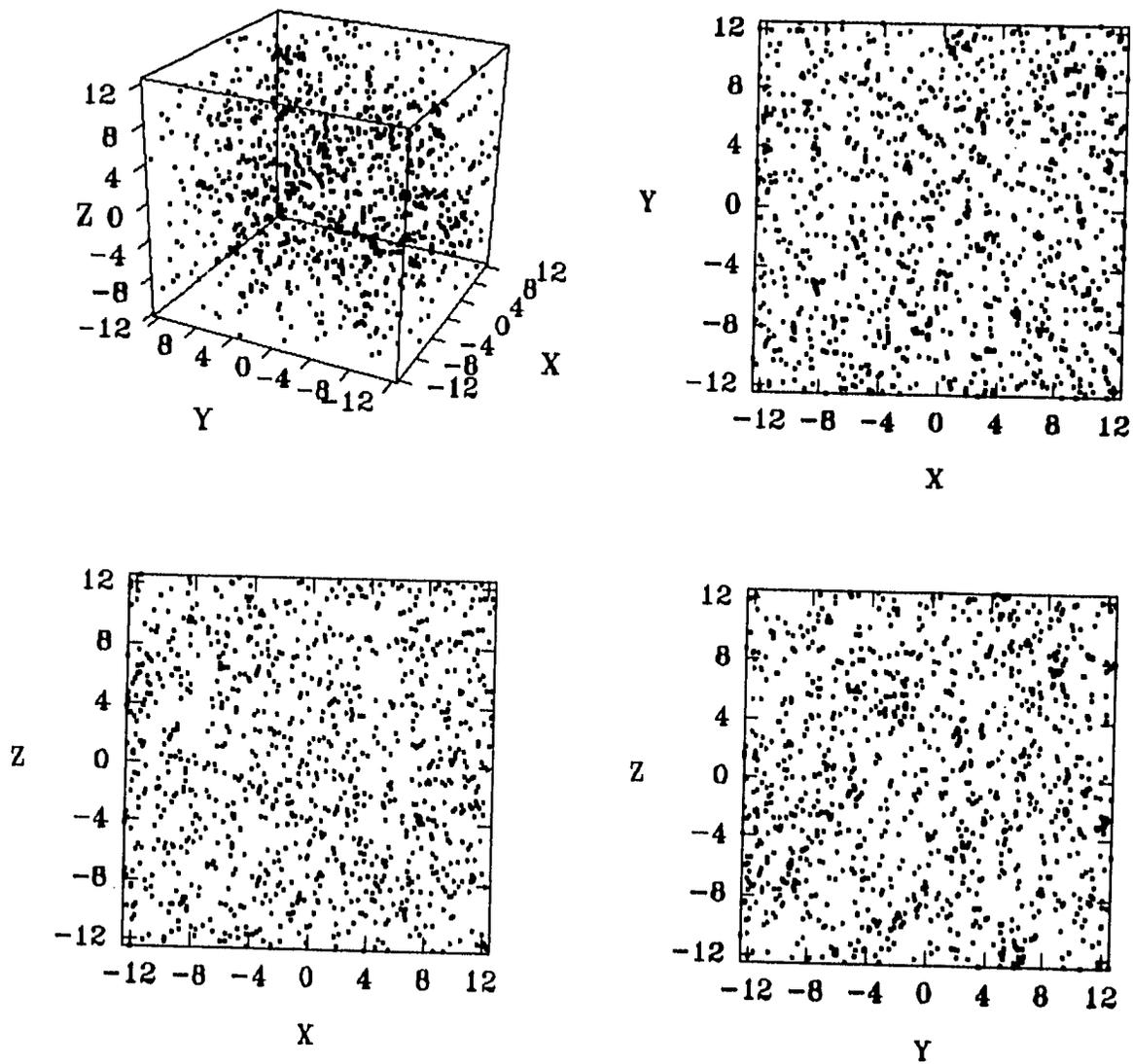


Figura 4.5 Proyección de puntos aleatorios en 3D en los tres planos.

El número de triadas aleatorias que podrán ser centros de hoyos, está dado por la variable entera alea , cuyo valor generalmente se coloca en 20,000. Este valor fue obtenido por ensayo y error ya que, como se verá en el módulo que calcula el volumen vacío de la estructura,

aproximadamente con 15,000 triadas aleatorias el algoritmo converge, es decir, con este número de hoyos, se llega a detectar un alto porcentaje del volumen libre contenido en la estructura.

La figura 4.6 muestra una gráfica con resultados obtenidos con una estructura química llamada faujasita, donde se observa el crecimiento del número de hoyos que el algoritmo encuentra en relación al número de triadas aleatorias generadas.

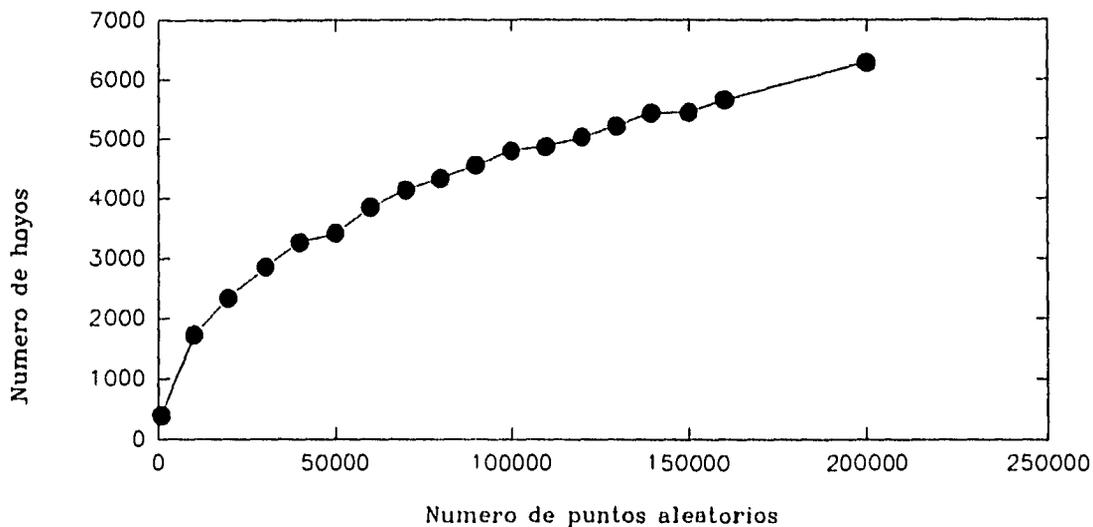


Figura 4.6 Número de hoyos en función del número de triadas.

Aunque el algoritmo continua encontrando hoyos al aumentar el número de triadas aleatorias, los hoyos que se localizan constituyen pequeñas esferas que quedan en los intersticios, son demasiado pequeños y no son de gran interés. En la figura 4.7 se muestra la distribución de tamaños de hoyos.

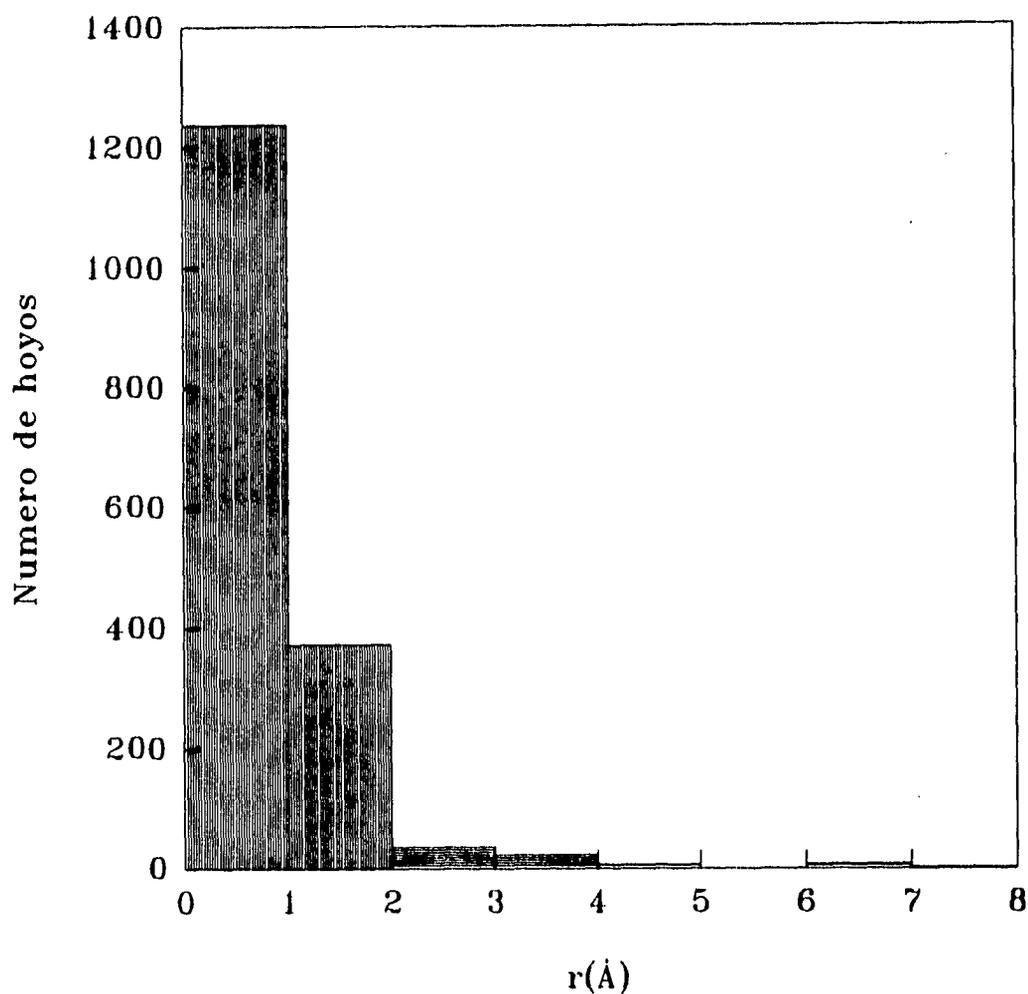


Figura 4.7 Distribución del tamaño de hoyos.

El módulo tiene la opción de obtener el conjunto de hoyos de la estructura con condiciones de periodicidad ya sea en dirección de uno, dos o de los tres ejes. Las condiciones de periodicidad se refieren a tomar a la estructura no como una estructura aislada, sino que se repite infinitamente, lo cual provoca que el criterio para asignarle una longitud a un hoyo cambie. En realidad, para encontrar los hoyos con condiciones de periodicidad en las tres direcciones de una estructura, sólo

importarían las cajas adyacentes, para lo cual se toman las imágenes de las partículas en las cajas vecinas.

El archivo de salida que genera tiene el mismo nombre que el archivo de entrada, más la extensión `.hoyo`. El contenido de éste es una lista de las coordenadas y radios de cada hoyo encontrado en la estructura. El siguiente diagrama muestra el funcionamiento de este módulo.

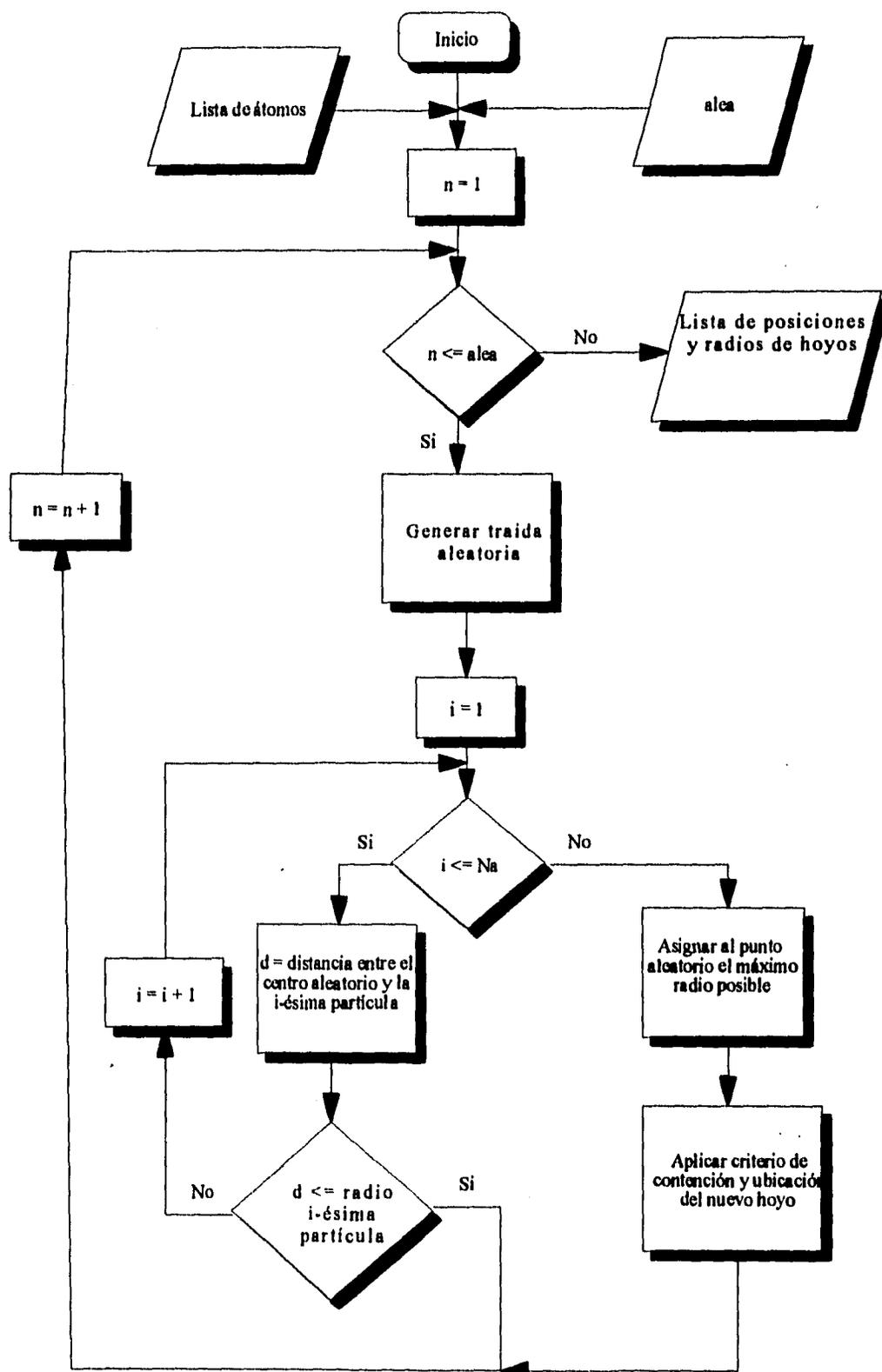


Figura 4.8 Diagrama de flujo del módulo buscar hoyos.

4.3.5 Dibujar hoyos

Consiste en tomar la información generada por el módulo **Buscar hoyos**, y dibujar esferas en la posición y del tamaño indicados. Los hoyos se dibujan de color blanco con el objeto de distinguirlos de los átomos.

Dependiendo de las características de las estructuras a analizar, se observa una diferente cantidad de hoyos entre las partículas. Si los hoyos generados en una estructura son demasiados, entonces al momento de visualizarlos junto con la estructura, se perderían detalles importantes. Es por esto que se hace uso de las propiedades de la esfera y se altera su transparencia para permitir seguir viendo en su totalidad la estructura y tener bien localizados los hoyos. Se controla la transparencia de la esfera por medio de una perilla, en donde se asocia la posición angular del punto de referencia de la perilla con el valor de transparencia de la esfera.

4.3.6 Calcular volumen y densidad

Si se deseara calcular el volumen y la densidad de un conjunto de átomos encerrados en una caja imaginaria, y teniendo tanto la información del radio de cada uno de esos átomos como las dimensiones de la caja, el cálculo parecería sumamente sencillo y sin complicación. Pero en la realidad, es decir, en un laboratorio experimental, el volumen se obtiene midiendo el volumen de una sustancia que cabe dentro de la estructura, es decir, que experimentalmente se mide el volumen vacío de la estructura para conocer el volumen que cubren las partículas y la densidad de la estructura. Con el fin de poder comparar los resultados de volumen y densidad experimentales, con los obtenidos en la simulación, es que se hace el cálculo sobre la estructura de poros, pretendiendo que los hoyos representen las especies absorbidas que pueden ser argón, nitrógeno o mercurio.

El procedimiento que se realiza para obtener el volumen vacío de la estructura consiste en partir la caja computacional en miles de pequeños prismas rectangulares, los cuales son considerados e incrementados al volumen vacío cuando su centro se encuentra dentro de al menos un hoyo. El criterio de convergencia es el siguiente:

$$\lim_{n \rightarrow \text{alea}} V''_{\text{hoyos}} = V_{\text{vacío}}$$

De esta forma, con el objeto de probar la convergencia del algoritmo buscador de hoyos se muestra la figura 4.9 donde la parte superior indica el volumen representado por el volumen de los hoyos más el volumen de las partículas. Después de la generación de 200,000 puntos aleatorios el volumen no aumenta substancialmente no obstante que se encuentran más hoyos. Esto se debe a que los hoyos encontrados son muy pequeños y su volumen es casi despreciable.

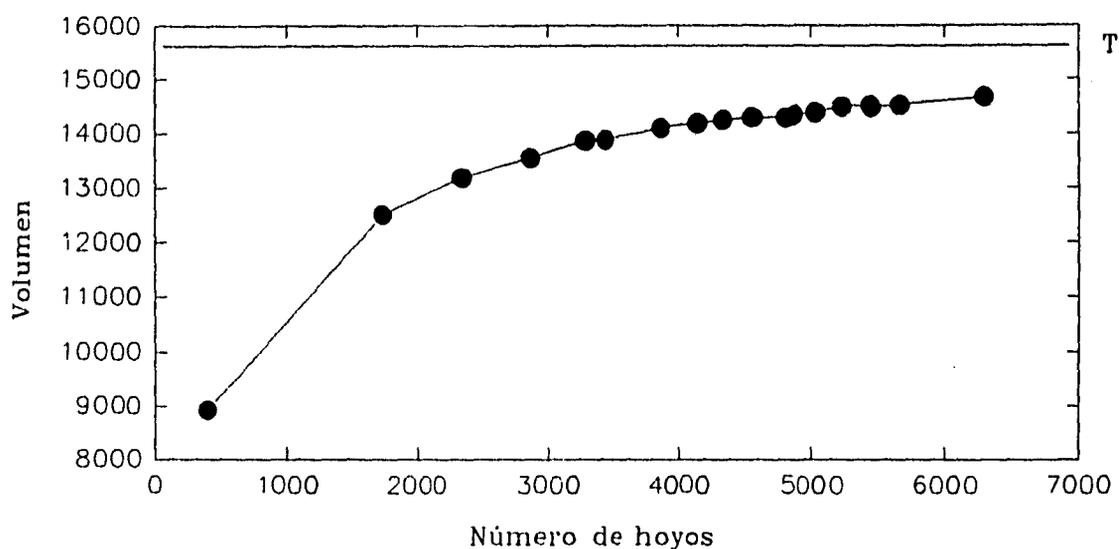


Figura 4.9 Aproximación de volumen.

El algoritmo encuentra entre el 85 y 90 por ciento del volumen total dependiendo de la porosidad de la estructura. Esto es debido al empaquetamiento de esferas dentro de la caja, y al hecho de que crece el número de hoyos encontrados, lo cual dificulta encontrar nuevos hoyos de radio grande. Las esferas representan tanto átomos como hoyos y naturalmente dejan meniscos entre ellas, cuyo volumen es impráctico de estimar. Sin embargo, como se ve de la figura 4.9 para todos fines prácticos, el volumen encontrado con este método representa, con un buen grado de realismo, el volumen libre de la estructura. No hay que olvidar, que en la realidad, el

empaquetamiento, por ejemplo de moléculas de nitrógeno dentro de una estructura molecular, deja también espacios intersticiales sin llenar. De hecho, en cálculos de microporosidad realizados en la fase amorfa de la γ alúmina se encontraron resultados muy similares a los experimentales obtenidos con absorción de nitrógeno.

4.3.7 Calcular potencial electrostático

En este módulo se calcula el valor del potencial electrostático en todos los puntos de una partición en 3D que están contenidos en los hoyos.

La motivación para calcular el potencial eléctrico dentro de las cavidades de las estructuras moleculares que se estudian en simulación numérica tiene dos vertientes. Por un lado, la necesidad de caracterizar los sitios ácidos, responsables de los fenómenos de catálisis, que se ha observado que no son propiedades puntuales ni en cavidades ni en superficies libres. Por otro lado, el interés de correlacionar heterogeneidades de potencial eléctrico dentro de cavidades y conductos en materiales amorfos basados en dióxido de silicio y los mecanismos de difusión de cationes pequeños. Esta última aplicación tiene también interés en geofísica pues estos mecanismos de difusión son los responsables de la conductividad eléctrica en rocas de la corteza y el manto superior de la tierra.

El cálculo eficiente de las contribuciones de largo alcance a la energía coulombiana es bastante difícil (Sangster *et al.*, 1976). Para obtener la energía coulombiana de cada partícula cargada del sistema molecular de N partículas, es necesario considerar su interacción con todas las $N-1$ restantes. Existen varios métodos para abordar este cálculo, sin embargo, el más eficiente está constituido por las llamadas sumas de Ewald (Ewald *et al.*, 1921) y es el que se incorporó al sistema de análisis de la dinámica molecular que aquí se presenta.

La energía coulombiana total de un sistema de N partículas confinadas en una caja cúbica de lado L está dada por:

$$V_c = \frac{1}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \sum_n \frac{z_i z_j}{r_{i,jn}}$$

donde z_i y z_j son las cargas de los iones i y j . El vector $r_{i,jn}$ es el desplazamiento de la partícula i a la partícula jn , la cual es la j ésima partícula en la caja imagen la cual se encuentra desplazada de la caja origen por Ln .

Sin embargo esta serie es condicionalmente convergente. El método de Ewald es en realidad una transformación de esta serie en una suma de tres series absolutamente convergentes

$$V_c = V_1 + V_2 + V_3$$

en donde:

$$V_1 = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \sum_{\substack{n \\ j \neq i \text{ si } n=0}} z_i z_j \frac{\text{ferc}(\alpha r_{i,jn})}{r_{i,jn}}$$

$$V_2 = -\sum_{i=1}^N z_i^2 \alpha \pi^{-1/2}$$

y

$$V_3 = \frac{1}{2\pi L} \sum_{i=1}^N \sum_{j=1}^N \sum_{h \neq 0} z_i z_j \frac{\exp\left(-\frac{\pi^2 |h|^2}{\alpha^2 L^2}\right)}{|h|^2} \cos\left(\frac{2\pi}{L} h r_{ij}\right)$$

La primera serie se identifica en Física del Estado Sólido con una suma en el espacio real. La segunda es un término que en general es constante para cada sistema y se puede interpretar como la autoenergía de las partículas. El tercer término constituye una suma en el espacio recíproco. Donde ferc es la función de error complementaria dada por

$$\text{ferc}(z) = \frac{2}{\pi} \int_0^{\infty} e^{-t^2} dt = 1 - \text{fer}(z)$$

$$f_{er}(z) = \frac{2}{\sqrt{\pi}} \int_0^{\infty} e^{-t^2} dt$$

y el término h es un vector de la red recíproca en unidades tales que sus componentes son enteros, α es un parámetro arbitrario y el potencial V_c es independiente del valor de α aunque claramente se observa que V_1 , V_2 y V_3 dependerán de su valor. En la práctica, las sumas infinitas sobre n y h son remplazadas por sumas de un número finito de términos y el número de términos el cual debe ser considerado con el fin de proporcionar una convergencia satisfactoria dependerá de la elección echa para α , y habrá una óptima elección de en términos de la eficiencia.

El potencial eléctrico se calcula utilizando solamente los dos primeros términos de la transformación de Ewald en virtud de que cierta selección de α hace que la contribución de la suma en el espacio recíproco contribuya a la energía potencial eléctrica total en menos del 10 por ciento. Sin embargo, en una versión posterior del programa de análisis se incorporará su cálculo.

El funcionamiento de este módulo consiste básicamente en partir la caja que encierra a las partículas en miles de pequeños prismas en cuyos centros se coloca una carga unitaria y puntual de prueba y se calcula su energía potencial. El cálculo del potencial se representa en el siguiente diagrama:

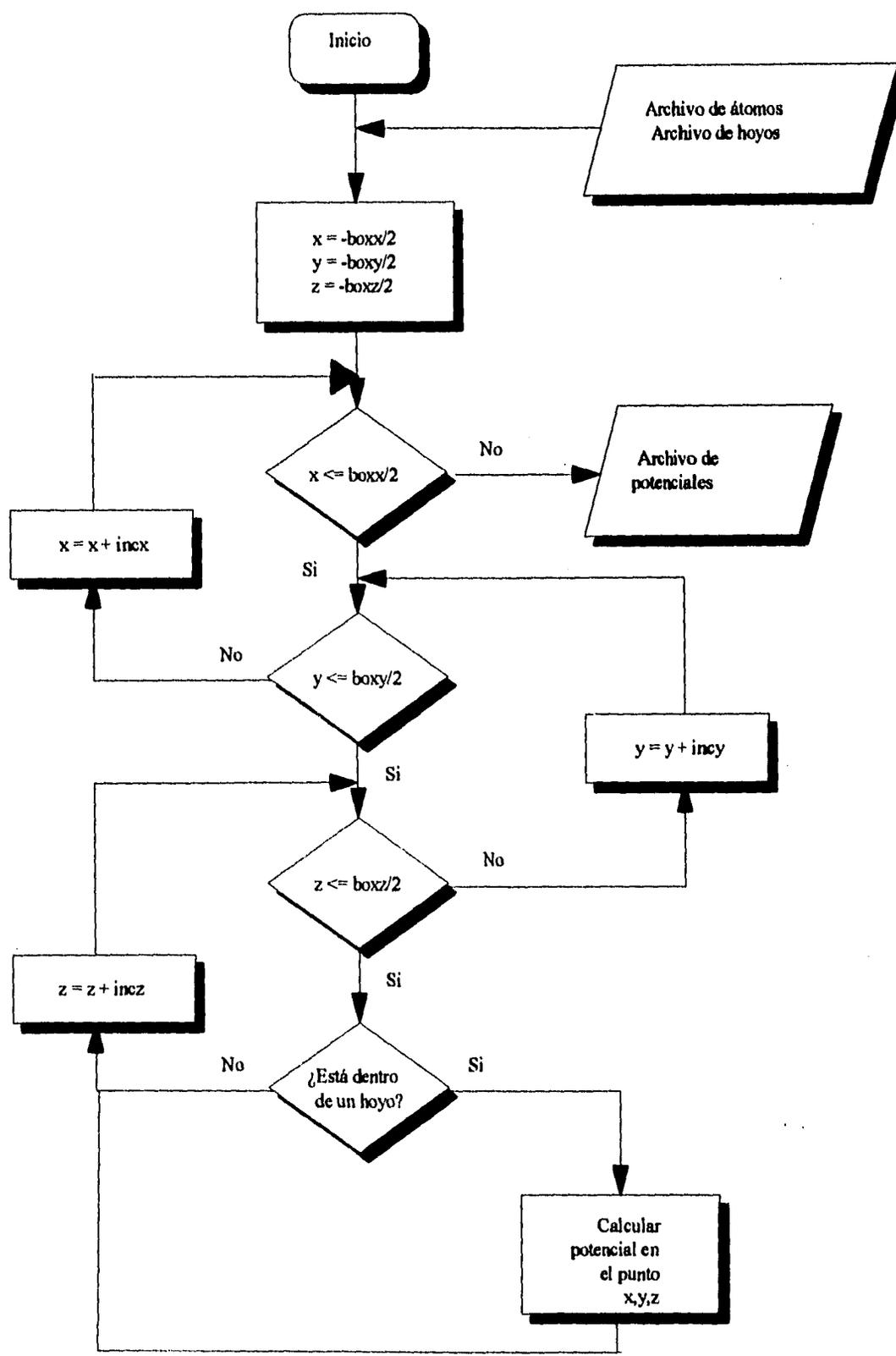


Figura 4.10 Diagrama de flujo del módulo calcular potencial electrostático.

Los archivos de visualización y hoyos son las entradas que necesita éste módulo para obtener los resultados.

En primer lugar se divide la caja computacional en pequeños prismas rectangulares. El algoritmo inicia en una arista del cubo, revisa si el centro del prisma está incluido en algún hoyo, si es así, entonces se calcula el potencial en ese punto, en caso contrario, se asume que cayó en una partícula o simplemente en un intersticio sin interés, asignándole a ese punto un valor arbitrario de potencial que se utiliza en la rutina de graficación. Los puntos en donde se calcula el potencial deben estar encerrados por los hoyos de la estructura ya que de esta manera manipulando tal archivo se pueden seleccionar las regiones en donde se desean saber los valores del potencial. El centro del pequeño prisma se va moviendo a través de todo el espacio encerrado por la caja computacional, de tal forma que se obtiene un valor de potencial para cada punto. El archivo de salida es un archivo (<nombre>.vis.pot) que contiene las coordenadas de cada punto y su potencial.

4.3.8 Dibujar curvas de potencial

El archivo generado por el módulo "Calcular Potenciales", se utiliza para unir los puntos formando una malla de diferentes tonos, de esta forma se pueden ver planos con regiones que, dependiendo del color, darán información visual del valor del potencial electrostático dentro de la estructura. Los valores de potencial se transfieren a una matriz dinámica tridimensional con el propósito de tener acceso a la información identificando los planos de potencial por medio de índices. Debido a que las ternas del archivo .pot van de la arista de coordenadas negativas a la arista con coordenadas positivas y a que el número de puntos de potencial es constante, es que en la matriz sólo se almacenan los valores de potencial sin sus coordenadas. Como se conocen las longitudes de la caja y el número de particiones que se le hicieron en cada lado para obtener los puntos de cálculo de potencial, se puede acceder a cualquier plano de potencial encontrando los índices correspondientes. Por ejemplo, si se desea observar el plano de potencial en $x = 6.5$, entonces el índice x de la matriz se calcula por:

$$indx = \frac{(NPX * 6.5) + \frac{boxx}{2}}{boxx}$$

donde NPX es el número de particiones sobre $boxx$ e $indx$ el redondeo del resultado de la ecuación.

El siguiente paso es hacer una "malla" donde se varían tanto la y como la z desde $-boxy/2$ a $boxy/2$ y desde $-boxz/2$ a $boxz/2$ respectivamente con x fija en 6.5. Se utiliza una malla triangular que es un conjunto de triángulos formados por una serie de puntos. Por ejemplo, si tenemos un conjunto de 8 puntos y el punto 4 está marcado como punto sin potencial por haber caído dentro de una partícula, entonces la unión de los puntos sería la siguiente:

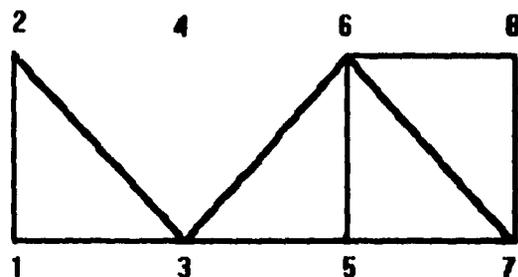


Figura 4.11 Formación de planos por mallas.

El color de la malla varía según el potencial en cada punto. El color para cada punto es evaluado en función de su potencial. Si el potencial es positivo entonces se interpola entre el azul (b) más oscuro para el potencial menos positivo y el azul más claro para el mayor potencial. En caso de ser negativo el potencial, entonces el color se interpola entre rojo (r) oscuro para los menos negativos y rojo brillante para los más negativos. A continuación se resume este proceso:

$$v = 0$$

Si $pot > 0$ entonces

$$r = 0$$

$$b = \frac{255}{q_{\max} - q_{\min}} * (m[indx][y][z] - pot_{\min})$$

en caso contrario

$$b = 0$$

$$r = \frac{255}{q_{\max} - q_{\min}} * (pot_{\max} - m[indx][y][z])$$

Al tener el valor del color en cada uno de los puntos, la función de la biblioteca gráfica que une los puntos, automáticamente llena el triángulo formado interpolando el color interno entre los tres colores límites. Es decir, que automáticamente desvanece el color interno, pixel por pixel desde un extremo a los otros dos.

5. FUNCIONAMIENTO Y RESULTADOS

Este capítulo muestra los resultados al visualizar una estructura producto de una simulación de DM, y debido a que se presentan cada una de las opciones del sistema, es que se puede tomar como referencia para la guía del usuario.

Para exponer el funcionamiento del sistema, se trabajó con una estructura de faujasita simulada en la supercomputadora. La faujasita es un material inorgánico de la familia de las zeolitas cuyas propiedades más importantes son sus formaciones porosas y adsorbentes, que permiten usarla como catalizador, separador de gases, etc.

5.1 Procesos subsidiarios

El archivo que contiene los resultados de la simulación de la faujasita se llama **fau12.vis**. Un fragmento de éste se muestra a continuación:

```
12/1 estadísticas
 25.0280      25.0280      25.0280 90.0
 1
591
 1      -1.39058      3.14181      0.89934      1.39373      14
 2      7.43881      3.19104      0.99176      1.38752      14
 3      7.33563      3.15246      5.23171      1.65000      13
 4     -1.22582      2.98466      5.12917      1.65000      13
 5      0.82253     -1.44156      3.08307      1.37775      14
 6      0.83401      7.59861      3.07380      1.40347      14
 7      5.20122      7.68362      3.15029      1.38830      14
 8      5.36046     -1.38472      2.96126      1.39978      14
 9      2.95078      0.96923     -1.40945      1.39498      14
10      3.14127      0.76813      7.59151      1.38892      14
11      3.07295      5.27198      7.63044      1.39296      14
```

5.1.1 Buscar hoyos

Como se mencionó anteriormente, en este tipo de estructuras es importante el análisis no sólo de las posiciones de los átomos, sino también la formación de las cavidades dentro de ella. Es por esto que el proceso buscador de hoyos juega un papel importante en el análisis. El proceso funciona como un programa independiente que recibe como entrada el archivo de visualización y arroja otro con una lista de la posición y radio de cada uno de los hoyos encontrados ordenados del de mayor radio al de menor.

El proceso buscador de hoyos, junto con los otros procesos subsidiarios, se ejecutó en la supercomputadora mediante la siguiente instrucción:

```
% buscah fau12.vis <entrar>
```

El proceso generó un archivo de nombre `fau12.vis.hoyo` con un tamaño de 331,408 bytes. Asimismo, para efectos de demostración, este programa se ejecutó con una herramienta de CRAY, llamada `hpm` (del inglés *hardware performance monitor*), que muestra el rendimiento de la máquina cuando ejecuta un programa. El resultado de este monitoreo arrojó la siguiente información:

| | | | | | |
|-------------------------|---|--------|-------------------|---|-------------|
| Group 0: CPU seconds | : | 279.09 | CP executing | : | 46515222334 |
| Million inst/sec (MIPS) | : | 50.15 | Instructions | : | 13997575065 |
| Avg. clock periods/inst | : | 3.32 | | | |
| % CP holding issue | : | 50.98 | CP holding issue | : | 23712889288 |
| Inst.buffer fetches/sec | : | 0.51M | Inst.buf. fetches | : | 141755065 |
| Floating adds/sec | : | 6.51M | F.P. adds | : | 1817783528 |
| Floating multiplies/sec | : | 4.42M | F.P. multiplies | : | 1232215514 |
| Floating reciprocal/sec | : | 1.26M | F.P. reciprocals | : | 351284720 |
| I/O mem. references/sec | : | 0.17M | I/O references | : | 46848125 |
| CPU mem. references/sec | : | 6.01M | CPU references | : | 1678522555 |
| Floating ops/CPU second | : | 12.19M | | | |

Del reporte, la información que se utilizó fue el tiempo de procesador, el cual se comparó con la información del reporte de las ejecuciones del mismo programa pero en la estación *Silicon Graphics*, con lo cual se observó que el ahorro en tiempo de procesador en la CRAY con respecto a la estación era de un 70%. Este ahorro se debe principalmente a que el programa fue compilado con directivas de vectorización y a la rapidez de proceso que caracteriza a una supercomputadora.

5.1.2 Calcular potencial electrostático

El procedimiento para calcular el potencial electrostático dentro de la caja computacional es muy similar. Se ejecuta la instrucción:

```
% potencial fau12.vis <entrar>
```

El programa toma como parámetro `fau12.vis` y busca en el mismo directorio el archivo con nombre `fau12.vis.hoyo` ya que estos dos son los que necesita como entrada para realizar los cálculos. Actualmente el programa que calcula el potencial en las estructuras, lo hace para 51 particiones en cada eje, es decir, calcula el potencial electrostático en 51^3 puntos dentro de la caja imaginaria que encierra a la estructura. Este número de particiones fue fijado empíricamente, ya que las curvas de potencial se generaban con buena definición con los puntos de potencial obtenidos. Al terminar el proceso se obtuvo un archivo de nombre `fau12.vis.pot` con una longitud de 6,053,775 bytes, además del reporte de rendimiento:

```

Group 0: CPU seconds      :    304.31      CP executing       :    50718317056
Million inst/sec (MIPS)  :    47.50      Instructions       :    14454157186
Avg. clock periods/inst :     3.51      CP holding issue   :    24945034598
% CP holding issue      :    49.18      Inst.buf. fetches  :    296176799
Inst.buffer fetches/sec  :     0.97M     F.P. adds          :    3687028633
Floating adds/sec       :    12.12M     F.P. multiplies    :    2799266815
Floating multiplies/sec  :     9.20M     F.P. reciprocals   :    228657207
Floating reciprocal/sec  :     0.75M     I/O references     :     61650631
I/O mem. References/sec :     0.20M     CPU references     :    3416550855
CPU mem. references/sec  :    11.23M
Floating ops/CPU second :    22.07M

```

El porcentaje de tiempo ahorrado con respecto a las ejecuciones realizadas en la estación de trabajo rebasa el 50%.

5.1.3 Calcular volumen y densidad

La manera de ejecutar este proceso es:

```
% volumen fau12.vis <entrar>
```

Dando como resultado dos cifras, volumen libre y densidad de la estructura

```
Volumen: 9815.18654142      Densidad: 0.49819785
```

Este volumen encontrado no es exacto ya que si lo comparamos con el volumen obtenido al calcular el volumen de las partículas y restándoselo al volumen que encierra la caja

computacional, se observa un error de aproximadamente el 9%. Pero como se mencionó en el capítulo anterior, es un valor que se acerca más al experimental. El siguiente es el reporte de rendimiento de la corrida de búsqueda de volumen:

| | | | | | |
|-------------------------|---|---------|-------------------|---|-------------|
| Group 0: CPU seconds | : | 173.73 | CP executing | : | 28955300802 |
| Million inst/sec (MIPS) | : | 14.20 | Instructions | : | 2467523579 |
| Avg. clock periods/inst | : | 11.73 | | | |
| % CP holding issue | : | 88.29 | CP holding issue | : | 25563756415 |
| Inst.buffer fetches/sec | : | 0.01M | Inst.buf. fetches | : | 2184370 |
| Floating adds/sec | : | 100.56M | F.P. adds | : | 17470634986 |
| Floating multiplies/sec | : | 67.04M | F.P. multiplies | : | 11646116432 |
| Floating reciprocal/sec | : | 0.00M | F.P. reciprocals | : | 43 |
| I/O mem. references/sec | : | 0.06M | I/O references | : | 10618204 |
| CPU mem. references/sec | : | 67.95M | CPU references | : | 11804328758 |
| Floating ops/CPU second | : | 167.60M | | | |

Cabe aclarar que los programas sólo fueron compilados con la directiva de vectorización del compilador de "C" de CRAY, la cual automáticamente intenta optimizar el código para obtener un mayor rendimiento, pero los programas no fueron diseñados para ser ejecutados en una computadora vectorial, lo que significa que aún se pueden optimizar los códigos y así obtener un rendimiento mayor por parte de la supercomputadora.

5.2 Opciones del sistema

Para el efecto de esta demostración se tomaron, colocados en el mismo directorio, los archivos:

```
fau12.vis
fau12.vis.hoyo
fau12.vis.pot
```

previamente obtenidos por los módulos buscador de hoyos y calcular potenciales respectivamente.

El programa de visualización verifica la existencia de los archivos producidos por los módulos subsidiarios con el efecto de agregar las respectivas opciones al menú, es decir, si por ejemplo no existe el archivo con extensión "hoyo", entonces en el menú principal no aparecerá la opción *hoyos*, y por consiguiente tampoco se podrán realizar funciones que requieran información de dicho archivo. Lo mismo sucede para el archivo con extensión "pot".

El procedimiento para ejecutar el programa es tecleando:

```
% visualiza fau12.vis <entrar>
```

lo que implica que el programa tome como entrada el archivo `fau12.vis` y comience la visualización. En caso de omitir el archivo de visualización, el sistema presenta una caja de diálogo con el directorio de donde se puede seleccionar el archivo de entrada deseado.

Una vez que se le ha proporcionado al sistema el archivo de entrada, aparece en la pantalla un pequeño recuadro que se puede mover con el ratón. Colocándolo en una posición deseada y presionando el botón derecho, podemos fijar el tamaño de la ventana de visualización. Después de colocar la ventana del tamaño deseado, es dibujada la estructura por el sistema, tal y como se muestra en la figura 5.1.

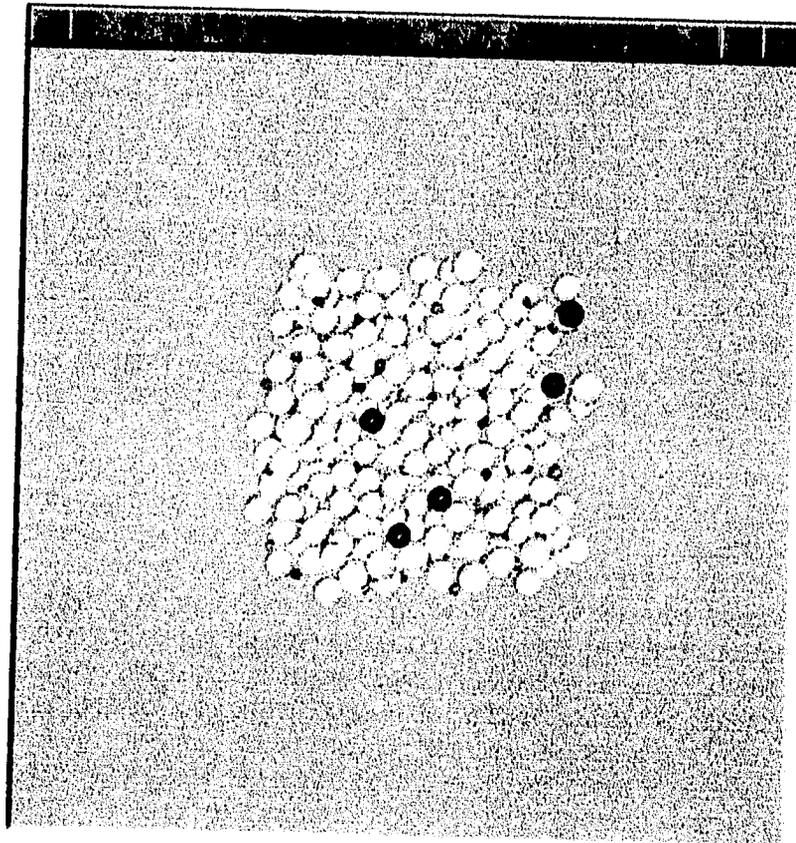


Figura 5.1 Pantalla inicial.

Esta estructura contiene partículas amarillas brillantes y amarillas opacas que representan los oxígenos con diferentes cargas, las partículas verdes son los sodios, las azul cielo los aluminios, y las azul marino representan a los silicios.

El menú de opciones del sistema sólo aparecè al mantener oprimido el botón derecho del ratón, y siempre y cuando el cursor se encuentre dentro de la ventana del sistema, lo cual permite tener un mayor campo de visión sobre la estructura (fig. 5.2); además de que se pueden tener habilitadas otras ventanas de trabajo o incluso otras ventanas del sistema de visualización. Una parte de las opciones funcionan como interruptores, es decir, habilitan la operación al seleccionarla, y la deshabilitan al seleccionarla por segunda vez. El resto de las opciones son aquellas donde se necesita un valor de entrada por parte del usuario, en este caso aparecen ventanas de diálogo donde el usuario puede introducir un valor específico.

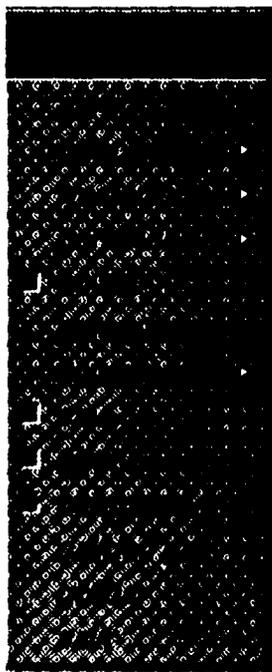


Figura 5.2 Menú principal del sistema.

La flecha horizontal que aparece al final de algunas opciones, es una liga que nos permite acceder a otro submenú con nuevas opciones. Algunas de las opciones del menú pueden aparecer en un tono más claro, lo que indica que esa opción no se encuentra habilitada por no contar con ese tipo de información, esto significa, por ejemplo, que si el sistema no encontró el archivo con

extensión .hoyo, entonces la opción que despliega los hoyos estará desactivada. El funcionamiento y utilidad de cada una de las opciones, tanto del menú principal como de los submenús, se describe a continuación sin seguir el orden indicado por el menú principal, esto con el objeto de hacer más entendible el funcionamiento del sistema.

5.2.1 Abrir

Muestra una caja de diálogo de donde se puede seleccionar un nuevo archivo para visualizar. La nueva información será desplegada en la misma ventana que ocupaba la imagen inicial.

5.2.2 Ejes

Cuando se efectúan movimientos o rotaciones de la estructura, llega el momento en que el usuario pierde la noción de la orientación de la estructura, es por esto que esta opción conmuta entre mostrar o quitar los ejes de referencia de la estructura (fig. 5.3).

5.2.3 Caja

La caja computacional que encierra a las partículas se muestra en esta opción como un prisma rectangular o romboide para el caso de que el ángulo de la caja sea diferente de 90° . Dicho prisma sirve como referencia de las características de la estructura que se despliegan (fig. 5.3). Esta opción funciona también como interruptor.

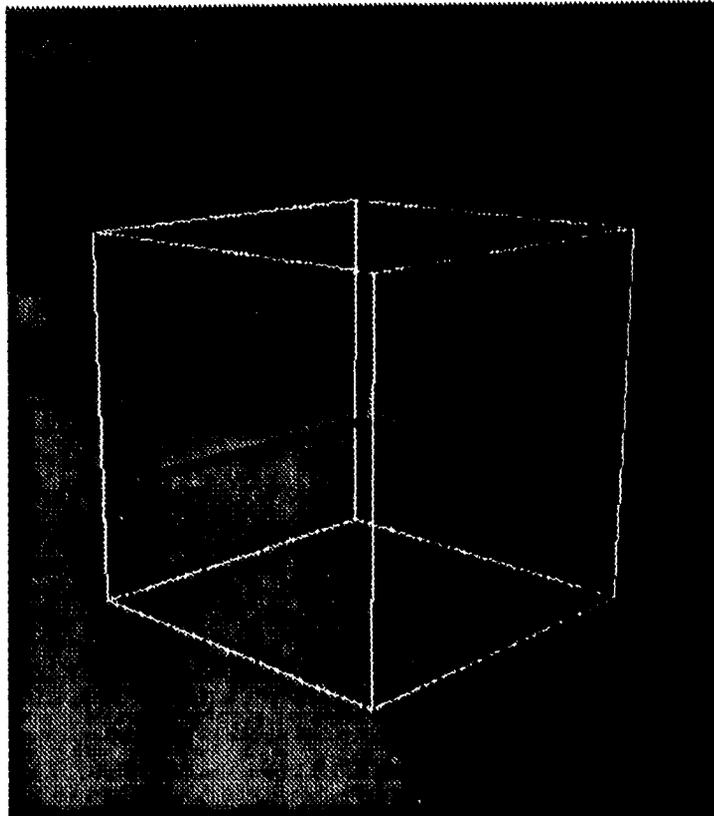


Figura 5.3 Caja y ejes de referencia.

5.2.4 Rotaciones

Al iniciar el funcionamiento del sistema, la orientación de los ejes de referencia presentan al eje *x* horizontalmente y en sentido derecho, al eje *y* verticalmente con sentido hacia arriba, y el eje *z* saliendo de la pantalla con sentido hacia el usuario. Esta orientación cambia gracias a la opción *rotaciones*. Existen tres formas de rotar la estructura. La primera es mediante el movimiento del ratón, efectuando una relación de la distancia que se mueve el ratón con respecto del ancho de la ventana de visualización, para transformarlo en grados. Esto significa que por ejemplo, si movemos el ratón hasta el extremo derecho de la ventana, entonces la estructura girará 180° alrededor del eje *y* (Fig. 5.4), y si lo movemos hasta el extremo inferior, entonces girará -180° alrededor del eje *x* (Fig. 5.5). De igual forma, si el ratón se mueve en sentido diagonal, entonces girará la estructura alrededor de los ejes *x* y *y* en proporción a la distancia con las paredes de la ventana (Fig. 5.6).

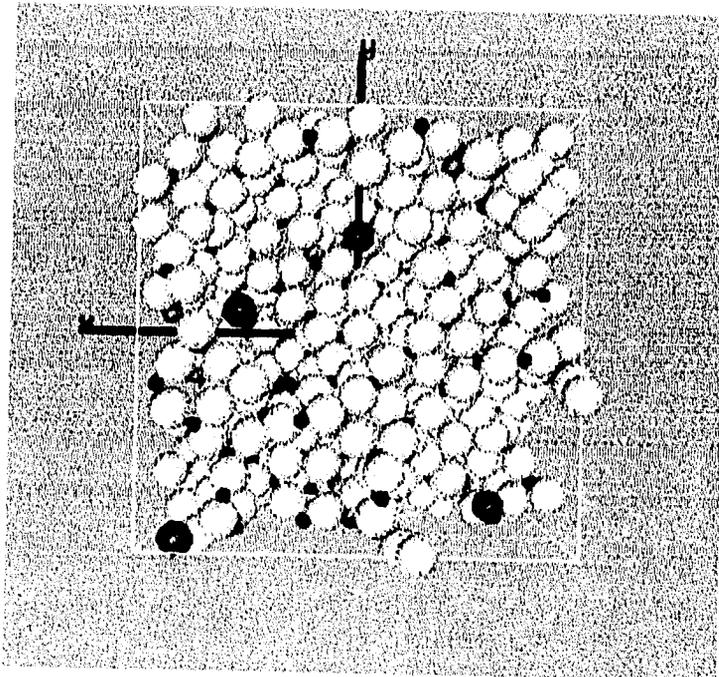


Figura 5.4 Rotación alrededor del eje y .

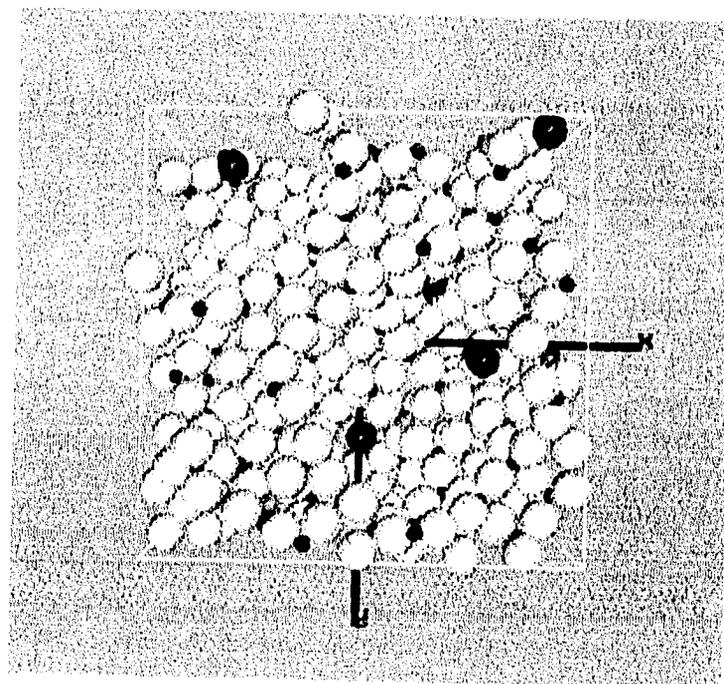


Figura 5.5 Rotación alrededor del eje x .

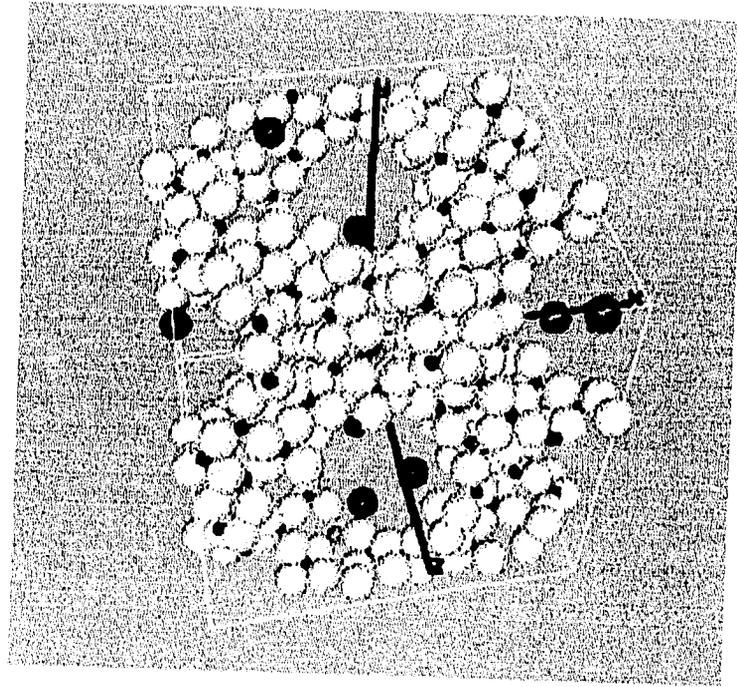


Figura 5.6 Rotación alrededor de los ejes x e y .

La segunda forma se vuelve necesaria ya que en ocasiones se requiere predefinir la orientación de la estructura respecto de su posición original, y es por esto que en esta opción se permite introducir los grados de rotación directamente con el teclado, con la ventaja que aquí se puede igualmente rotar alrededor del eje z . Al seleccionar esta opción, se presenta una ventana con tres campos de entrada, en los cuales se piden los grados que se desea que gire la imagen. De esta manera, si se introducen los valores mostrados en la figura 5.7 y se oprime el botón *Aceptar*, entonces la imagen resultante será como la mostrada en la figura 5.8.

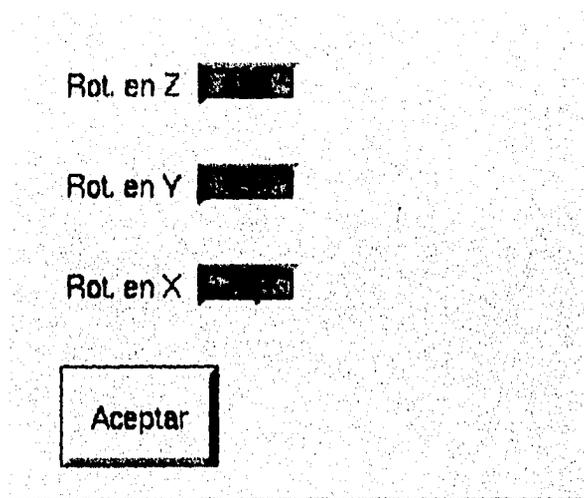


Figura 5.7 Predefinir rotación.

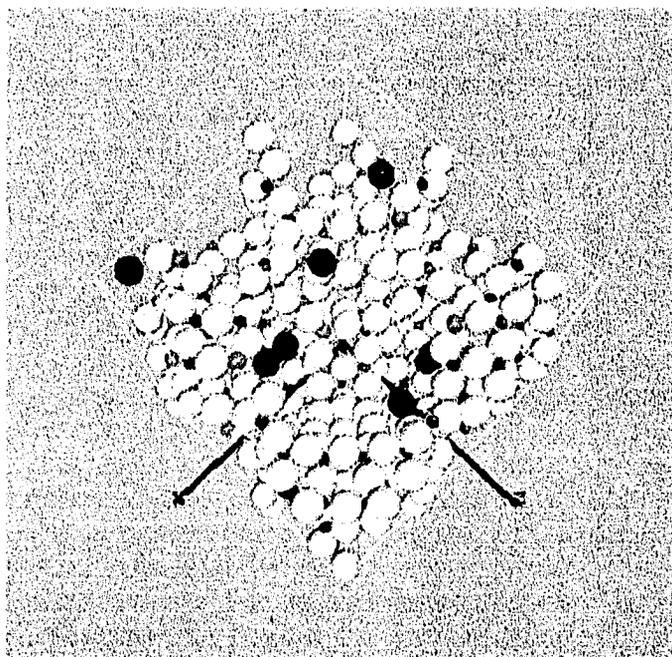


Figura 5.8 Rotación en los tres ejes de referencia.

Asimismo, como herramienta adicional y siempre y cuando exista la caja de perillas, las perillas 1, 2 y 3 rotan la estructura alrededor de los ejes x , y y z respectivamente. Al girar las perillas en sentido horario, la estructura gira en sentido positivo alrededor del eje seleccionado, y por el contrario, si se gira la perilla en sentido antihorario, la estructura será rotada en sentido negativo.

5.2.5 Traslaciones

En la labor de análisis no basta sólo con rotar la estructura, ya que para poder observarla a más detalle son necesarios también los movimientos de traslación. Esto es, mover la estructura a la izquierda o derecha, arriba o abajo, y tal vez la más importante de éstas, la que nos permite viajar al interior de la estructura. Para trasladar la estructura en cualquiera de las direcciones antes mencionadas, se selecciona la opción *traslaciones* y dentro de ésta se hace la elección del eje. Una vez seleccionado el eje en el cual se va a trasladar la estructura, se tiene el control del movimiento sobre ese eje en el ratón. El botón del centro realiza una traslación en el sentido positivo, es decir, mueve la estructura hacia la derecha, hacia arriba, o hacia el usuario según haya sido la elección del eje. El botón izquierdo realiza los movimientos contrarios, hacia la izquierda, hacia abajo y alejar la estructura. Las figuras 5.9 y 5.10 muestran algunas traslaciones de la estructura.

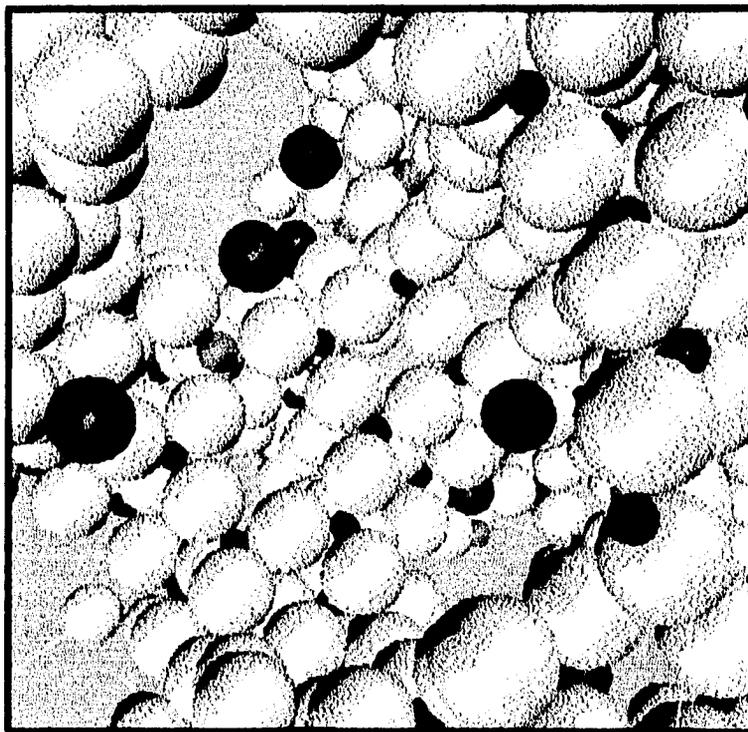


Figura 5.9 Traslación en el eje z de la estructura.

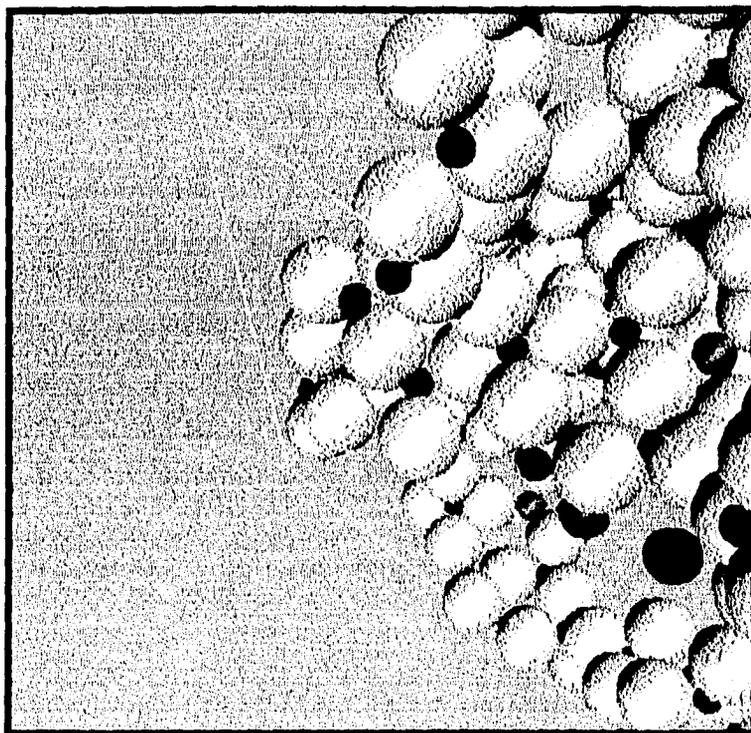


Figura 5.10 Traslación en los tres ejes de la estructura.

5.2.6 Partículas

La opción *partículas* contiene cinco opciones: *enc. apag.*, *seleccionar*, *etiquetas*, *info.* y *radio estandar*. La opción *enc. apag.* trabaja como interruptor, ya que al seleccionarla, el sistema deja de dibujar todas las partículas de la estructura, y al seleccionarla nuevamente las vuelve a dibujar.

Seleccionar permite dibujar los átomos de determinadas especies. Al elegir esta opción, aparece una ventana con un número de botones igual al número de especies. Cada uno de estos botones tiene el nombre del átomo a controlar. Estos botones, a su vez, funcionan como interruptores donde al pulsar el botón izquierdo del ratón sobre el botón de una especie en particular, activa o desactiva su despliegue en pantalla. Por ejemplo, el procedimiento a seguir para que el sistema sólo dibuje los átomos de sodio y de aluminio es, en primer lugar, seleccionar la opción *seleccionar* y al mostrar el sistema la ventana de selecciones, pulsar los botones de las partículas de oxígeno y silicio, de tal manera que estos botones estén hacia afuera mientras que

los de aluminio y sodio permanezcan oprimidos (fig. 5.11). Al aceptar esta operación, la estructura se verá como se muestra en la figura 5.12.

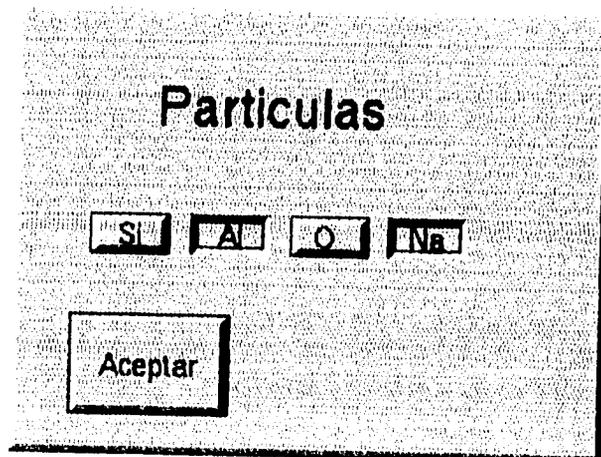


Figura 5.11 Selección de partículas a desplegar.

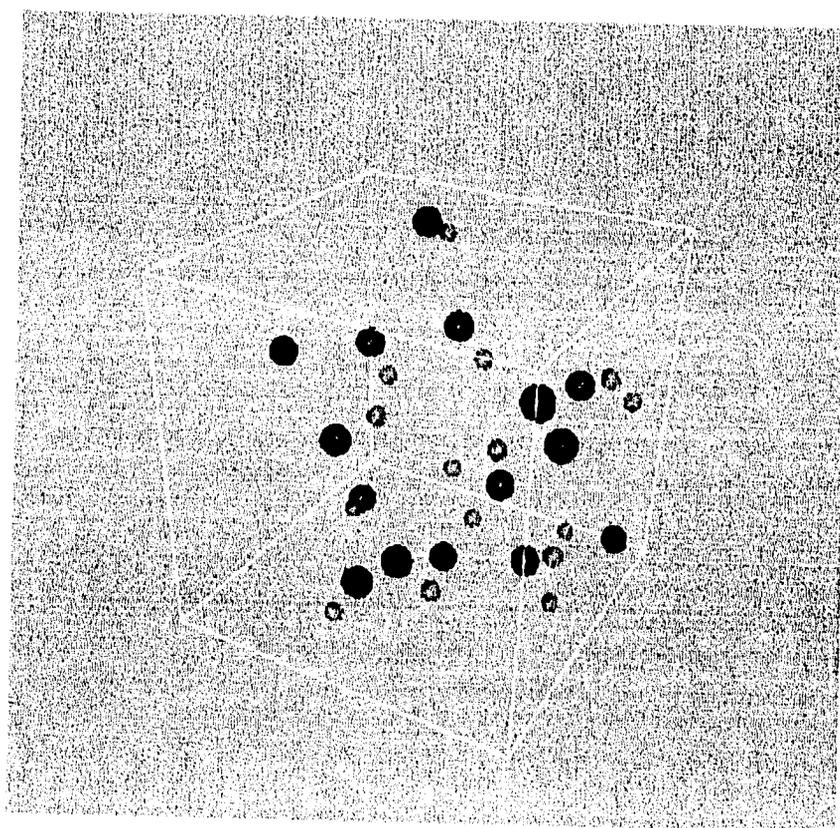


Figura 5.12 Despliegue de átomos de sodio y aluminio.

Como se mencionó anteriormente, en algunos casos es necesario distinguir los átomos de una misma especie o para seguir el comportamiento de algún átomo en particular. En este caso la opción *etiquetas* dibuja el número que etiqueta a cada una de las partículas de la estructura. Esta opción muestra las etiquetas de las partículas que en ese momento se encuentran activas para ser dibujadas (fig. 5.13).

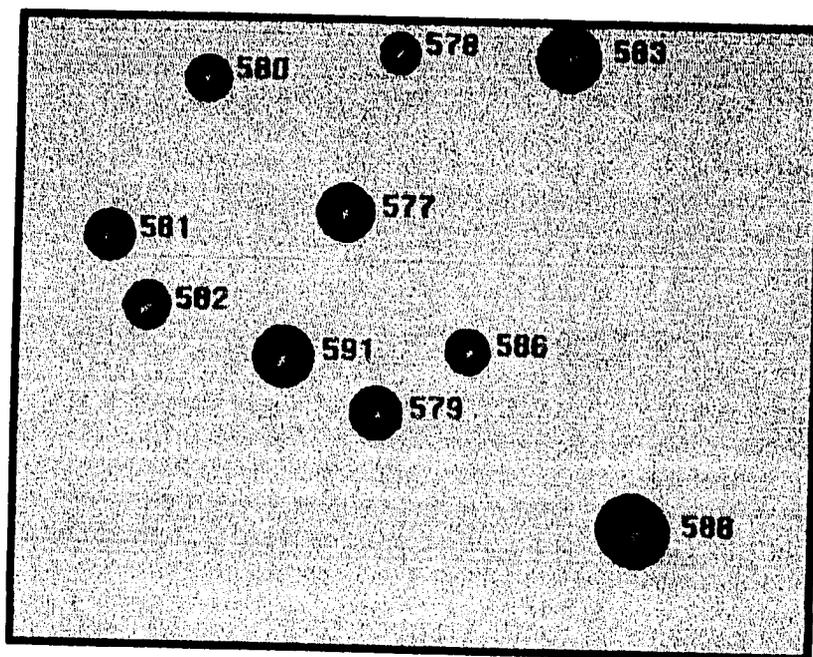


Figura 5.13 Etiquetas de partículas.

Al seleccionar *info*, se crea una nueva ventana que contiene información de la estructura. Esta ventana contiene una esfera con su respectivo color, nombre de la especie y el número de átomos por cada especie en la estructura (fig. 5.14). La ventana de información se puede manipular como cualquier ventana independiente, es decir que, se puede mover, ocultar o achicar. Funciona como interruptor, creando y borrando la ventana de información cada vez que es invocada.

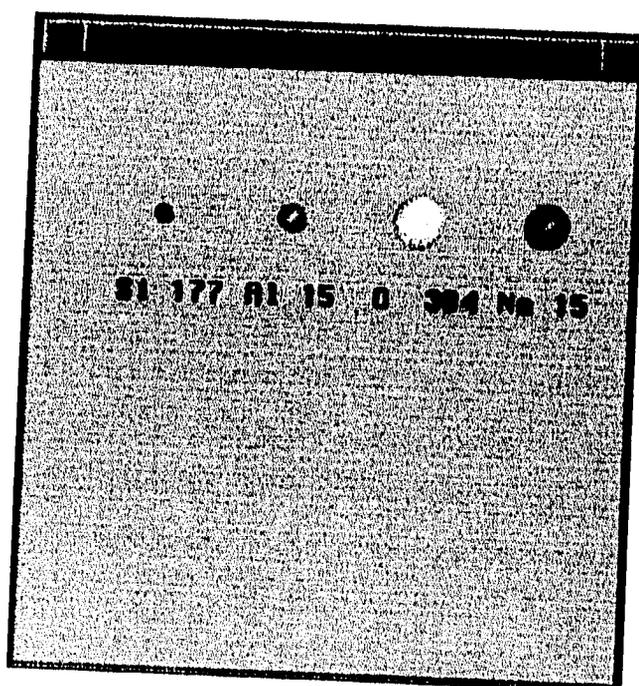


Figura 5.14 Ventana de información.

5.2.7 Conexiones

Dentro del análisis, es necesario estudiar el armazón de la estructura que permite apreciar características físicas como las coordinaciones de las partículas, la forma de la estructura, los poros que se forman dentro de ella, etc. El armazón de la estructura se observa a partir de la unión de determinadas partículas mediante un radio de alcance. Al seleccionar la opción *conexiones*, aparece una ventana de diálogo muy similar a la de elección de partículas, a diferencia que ésta tiene un campo de entrada donde el usuario introduce el radio de conexión deseado (fig. 5.15). Por ejemplo, como en el caso de la faujasita, las partículas de silicio y oxígeno son las que le dan forma al material, por lo que el procedimiento para analizar el armazón se lleva a cabo oprimiendo los botones de dichas partículas e introduciendo el radio de alcance, en este caso 1.7 \AA , con lo que el sistema unirá silicios con silicios, silicios con oxígenos y oxígenos con oxígenos, siempre y cuando se encuentren a una distancia menor o igual a 1.7 \AA .

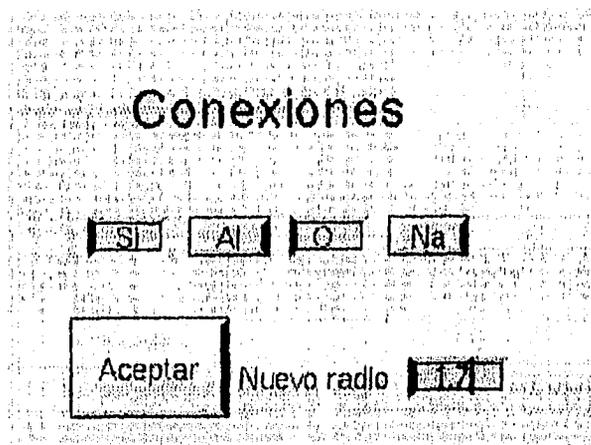


Figura 5.15 Selección de partículas a conectar.

Dibujando sólo la estructura de alambre y las partículas que no fueron unidas, el despliegue es el mostrado en la figura 5.16.

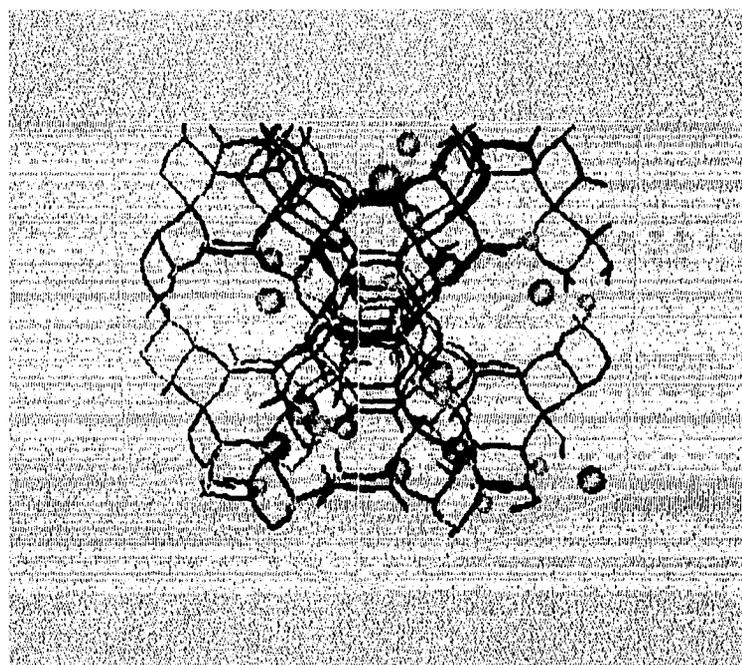


Figura 5.16 Estructura de alambre.

radio estandar es una opción que pertenece al menú de *partículas* pero generalmente se utiliza junto a la estructura de alambre anteriormente vista. Esta opción dibuja todas las especies de átomos de la estructura con un radio muy pequeño (fig. 5.17), esto es debido a que cuando

observamos sólo la estructura de alambre, perdemos la noción de la ubicación de las especies, y cuando dibujamos tanto la estructura de alambre como las partículas con sus dimensiones originales, no se puede distinguir la estructura de alambre.

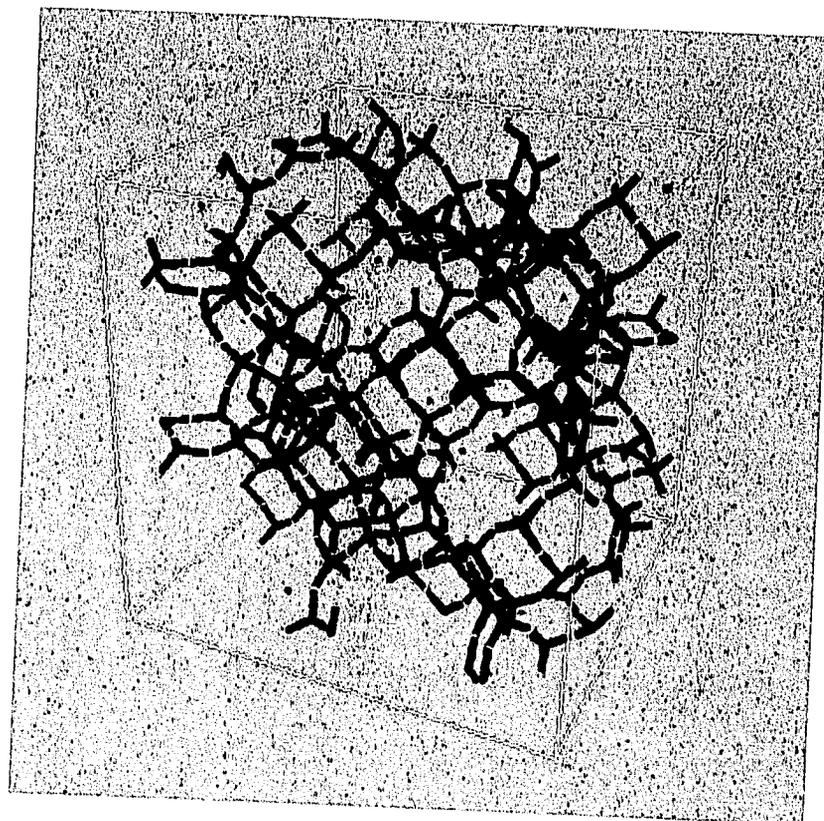


Figura 5.17 Radio estándar y estructura de alambre.

5.2.8 Hoyos

Con esta opción el sistema dibuja el conjunto de esferas que se encuentran en las cavidades de la estructura. Como se muestra en la figura 5.18, las esferas de color blanco de diferentes tamaños constituyen el sistema de poros de la estructura.

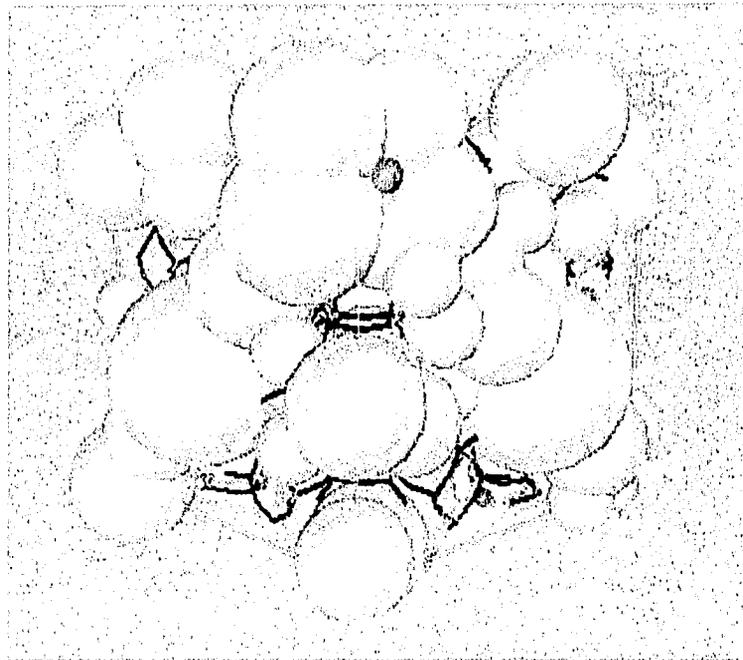


Figura 5.18 Hoyos de la estructura.

En este momento, la perilla 0 controla la transparencia de los hoyos con lo cual éstos no obstruyen la visión general de la estructura, permitiendo observar a las partículas y a las demás características de la estructura (fig. 5.19).

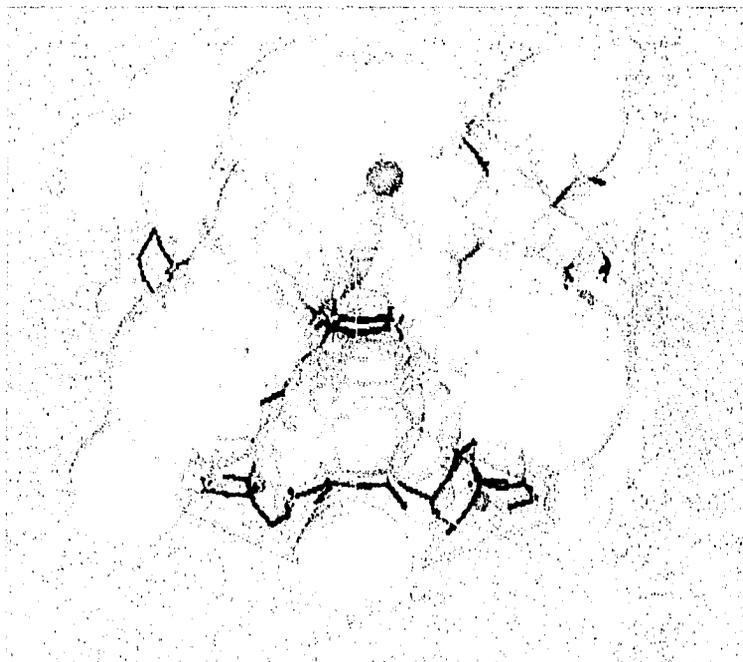


Figura 5.19 Transparencia de los hoyos.

5.2.9 Potenciales

Por medio de *potenciales* se puede observar el potencial electrostático de la estructura. Aquí se puede elegir el plano en el cual se desea observar el potencial, por ejemplo, si se selecciona la opción $y = cte.$ dentro de *potenciales*, e introduciendo un valor de 3.5 en la ventana que se muestra en la figura 5.20, el sistema mostrará el plano $y = 3.5 \text{ \AA}$. El plano presenta regiones con diferentes tonos de color azul y rojo. Las regiones de color azul claro representan potenciales positivos elevados, mientras que las de azul oscuro indican potenciales positivos pero cercanos a cero. Lo mismo sucede para las regiones de color rojo excepto que éstas indican potenciales negativos (fig. 5.21).

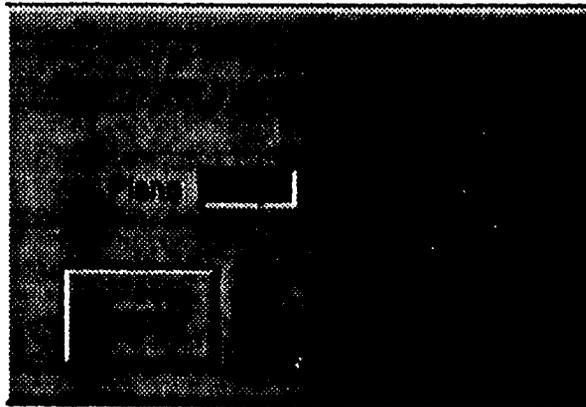


Figura 5.20 Selección del plano de potencial.

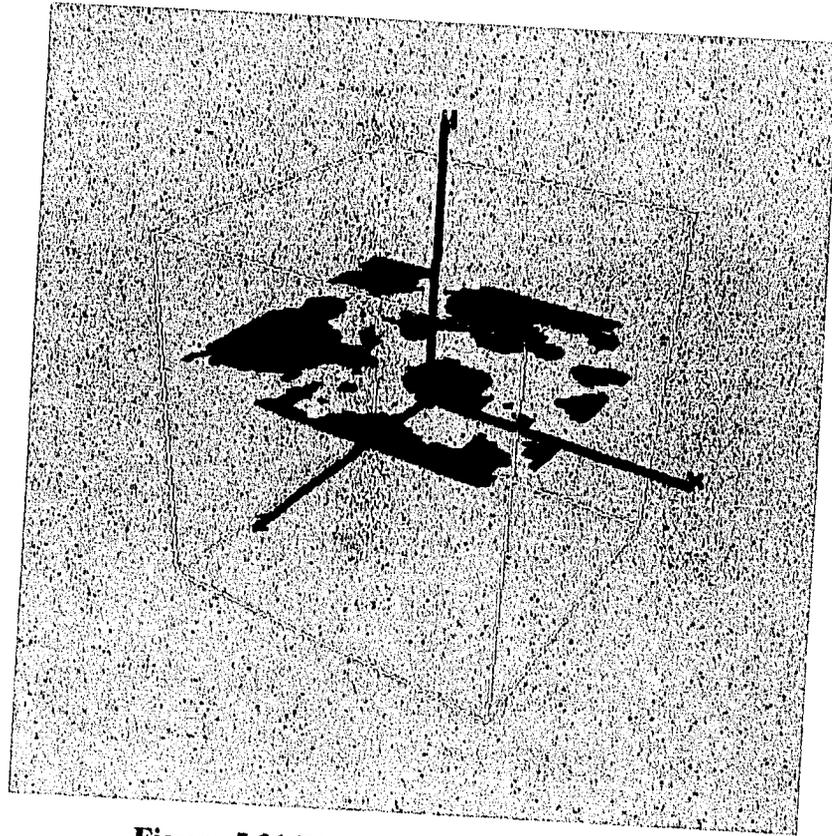


Figura 5.21 Plano de potencial $y=3.5 \text{ \AA}$.

En la figura 5.22 se observa el mismo plano pero con una altura asignada en función del valor del potencial, es decir, las regiones azul claro tendrán la mayor altura en el sentido positivo del eje, y las rojo claro con altura mayor pero en el sentido negativo del eje.

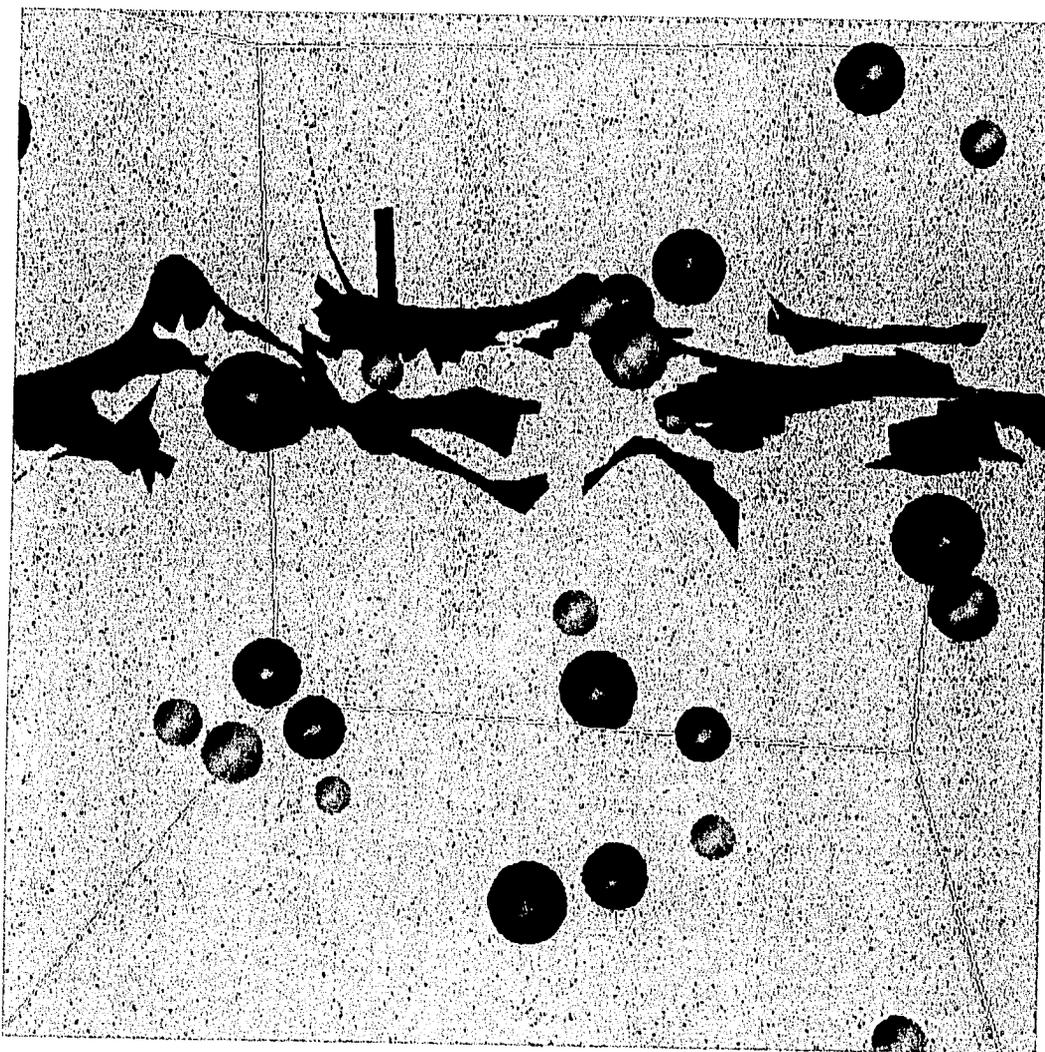


Figura 5.22 Distorsión del plano de potencial.

Asimismo, la opción *todos* despliega todos los planos de potencial de la estructura, mostrándose en pantalla volúmenes de potencial (fig. 5.23).

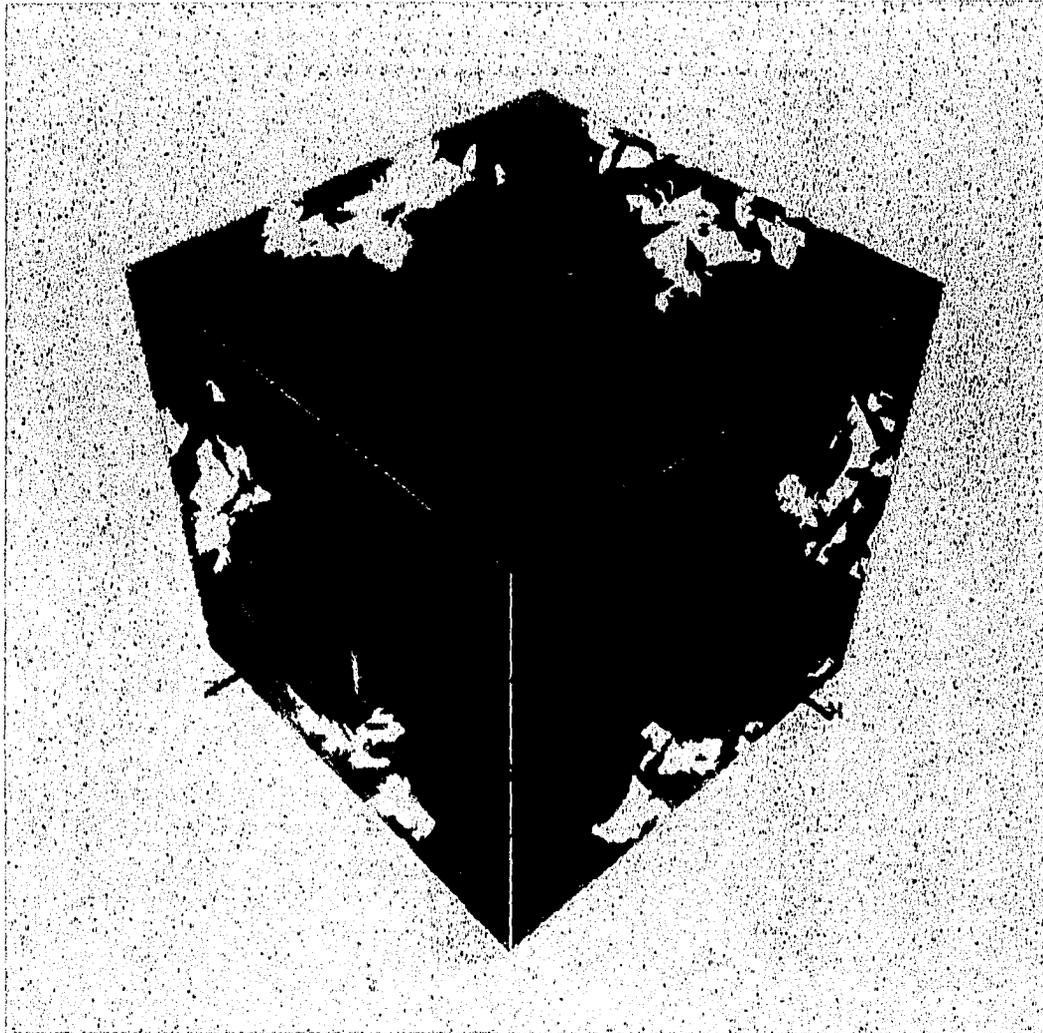


Figura 5.23 Potencial electrostático en toda la estructura.

Los espacios sin colorear indican la presencia de partículas, ya que el potencial electrostático solo se calcula y dibuja en el espacio encerrado por los hoyos. En la figura 5.24 podemos observar la tendencia del potencial electrostático junto con las partículas que ocasionan dichas variaciones de potencial.

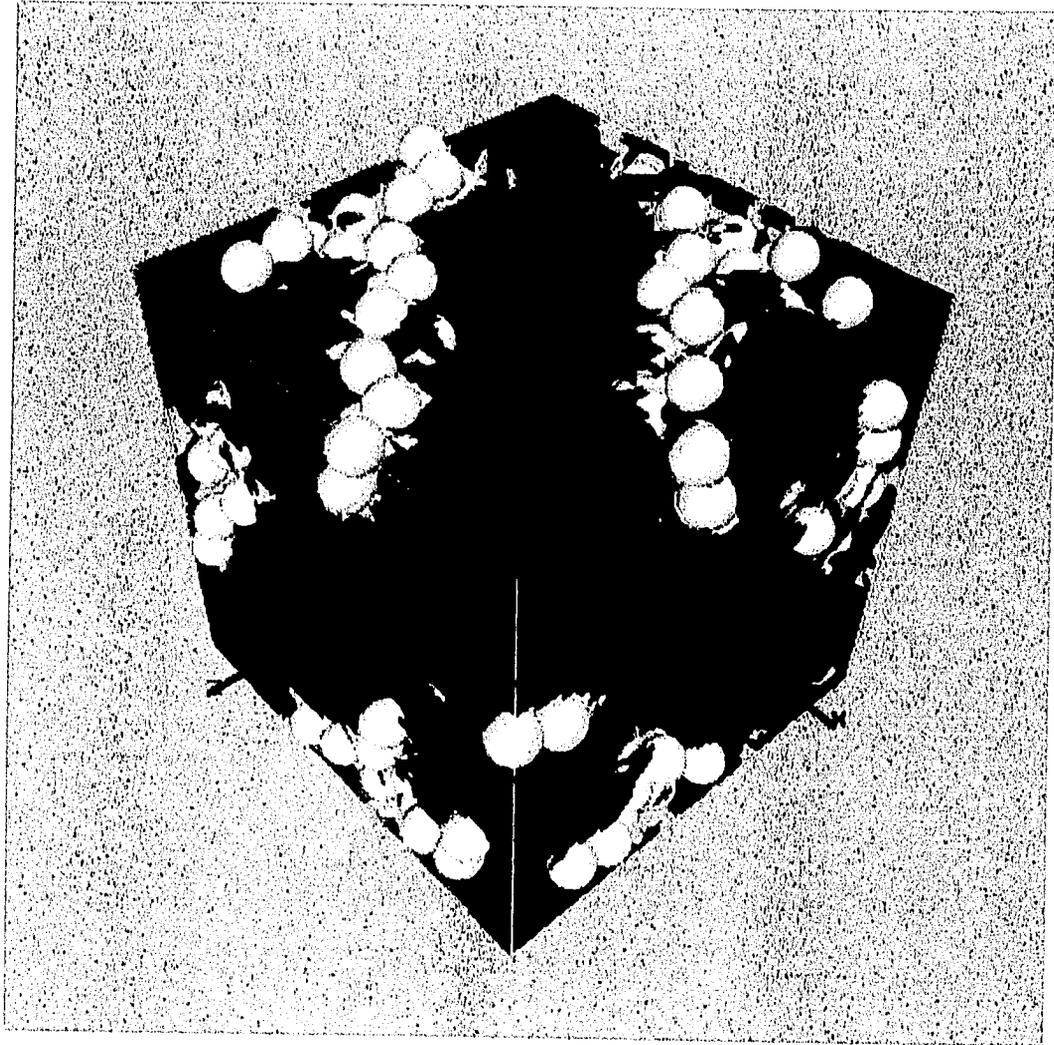


Figura 5.24 Potencial y partículas de la estructura.

5.2.10 Calidad de esferas

Las esferas que representan tanto a las partículas de la estructura como las que representan a los poros, se forman por medio de polígonos. Se puede definir el número de polígonos que forman a las esferas, con lo que se cambia su aspecto. Para el sistema se definieron valores entre 1 y 15, lo que implica que una esfera de calidad 1 será la más imperfecta, y por el contrario, el 15 indica mucho más definición en el despliegue. Por supuesto que entre mejor sea el despliegue de las esferas, mayor será el tiempo que tomará la máquina para dibujarlas. El sistema dibuja al inicio esferas con resolución 5. Para mostrar el contraste, si

seleccionamos esta opción aparecerá una ventana preguntando el nuevo valor de resolución (figura 5.25), es aquí donde se altera el valor inicial por una resolución de 2 (figura 5.26).

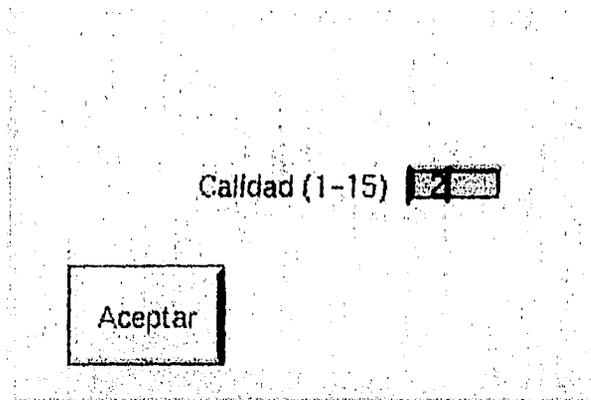


Figura 5.25 Elección de la resolución de las esferas.

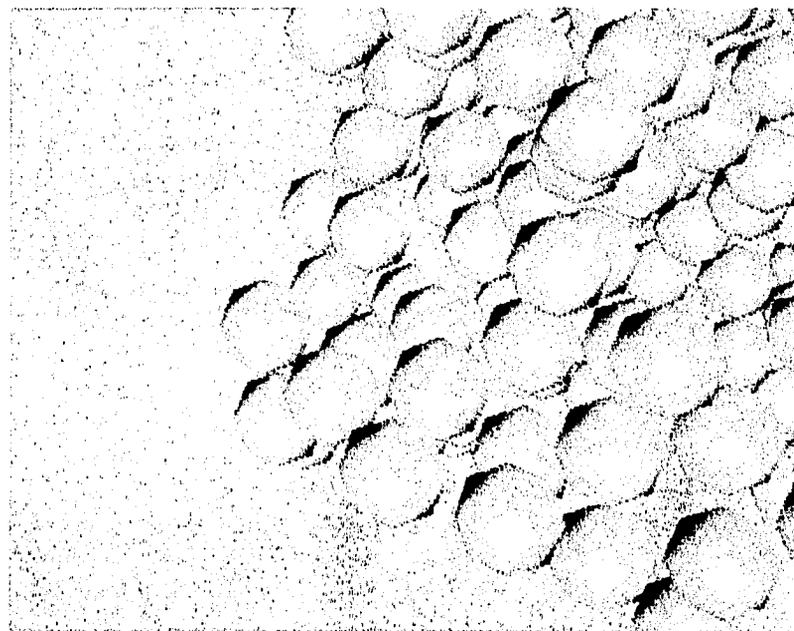


Figura 5.26 Resolución baja de esferas.

5.2.11 Grabar

Esta opción sólo se habilita cuando existe un equipo de grabación y edición de video conectado a la máquina. Al seleccionar *grabar* el sistema ejecuta, al terminar de desplegar cada

cuadro, un programa que envía la imagen de la pantalla a una cinta de video. Externamente, se programa el equipo de video para que cada cuadro de la simulación sea grabado en uno o más cuadros en la cinta de video.

5.2.12 Salir

Esta opción termina la ejecución del programa de visualización.

6. APLICACIONES

El sistema de visualización que aquí se describe ha sido utilizado, en cada una de sus fases intermedias de desarrollo y en su configuración final, por el grupo de trabajo del Laboratorio de Simulación de Materiales de la DGSCA, en la visualización y análisis de un número de sistemas simulados. De estos, quizás el más importante sea el de la γ -alumina, en el que gracias a su utilización ha sido posible observar por primera vez fenómenos de reconstrucción de superficies. También se han utilizado modificaciones del sistema para realizar otro tipo de tareas, como la de visualizar simulaciones del sistema planetario de la tierra y en la puesta en marca de simulaciones de sistemas complejos como el óxido de cromo y el dióxido de silicio amorfo, entre otros. Dos de estas aplicaciones se describen a continuación:

a) El sistema fue utilizado con fines didácticos de materias de la licenciatura de Física y de la maestría de Ciencias de la Computación de la Facultad de Ciencias y del Instituto de Investigación en Matemáticas Aplicadas y Sistemas de la UNAM, donde los alumnos realizaron un programa que simulaba el movimiento del sol y los planetas del sistema solar por un periodo de 100 años. Debido a que el sistema despliega partículas representadas por esferas, se logró adaptarlo al despliegue de dicha simulación (figura 6.1). Los únicos cambios que se le hicieron al sistema fueron el obvio cambio de escala en la medición de distancias, de angstroms a distancias tierra-sol y en el archivo de inclusión `atomo.h`, donde fueron cambiadas las características de los átomos por las de los planetas, es decir, se cambiaron nombres y radios de la estructura haciendo que el tipo de planeta coincidiera con el del resultado de la simulación.

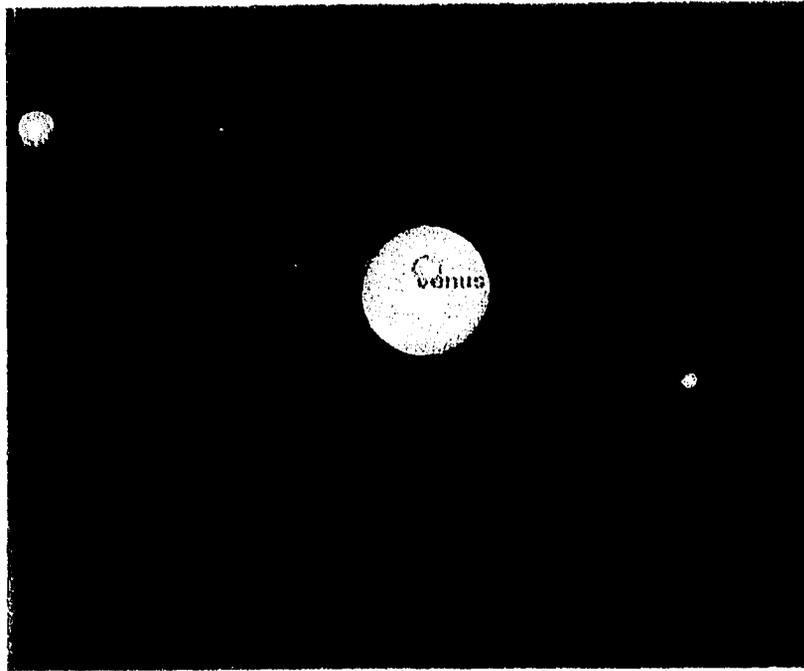


Figura 6.1 Simulación del sistema solar.

b) La figura 6.2 muestra 20 configuraciones finales de igual número de simulaciones a diferentes densidades y temperaturas de una estructura de gama alúmina. La importancia del estudio en este tipo de material radica en el uso de su superficie como catalizador. Con la ayuda del sistema de visualización se observó que la estructura se esponjaba en su superficie, donde la configuración de mayor densidad y sometida a mayor temperatura, resultó la más propicia para la formación de poros. Asimismo, se pudo observar que la formación del sistema de poros seguía una orientación paralela al plano (110) de la estructura, característica prácticamente imposible de descubrir sin la ayuda gráfica.

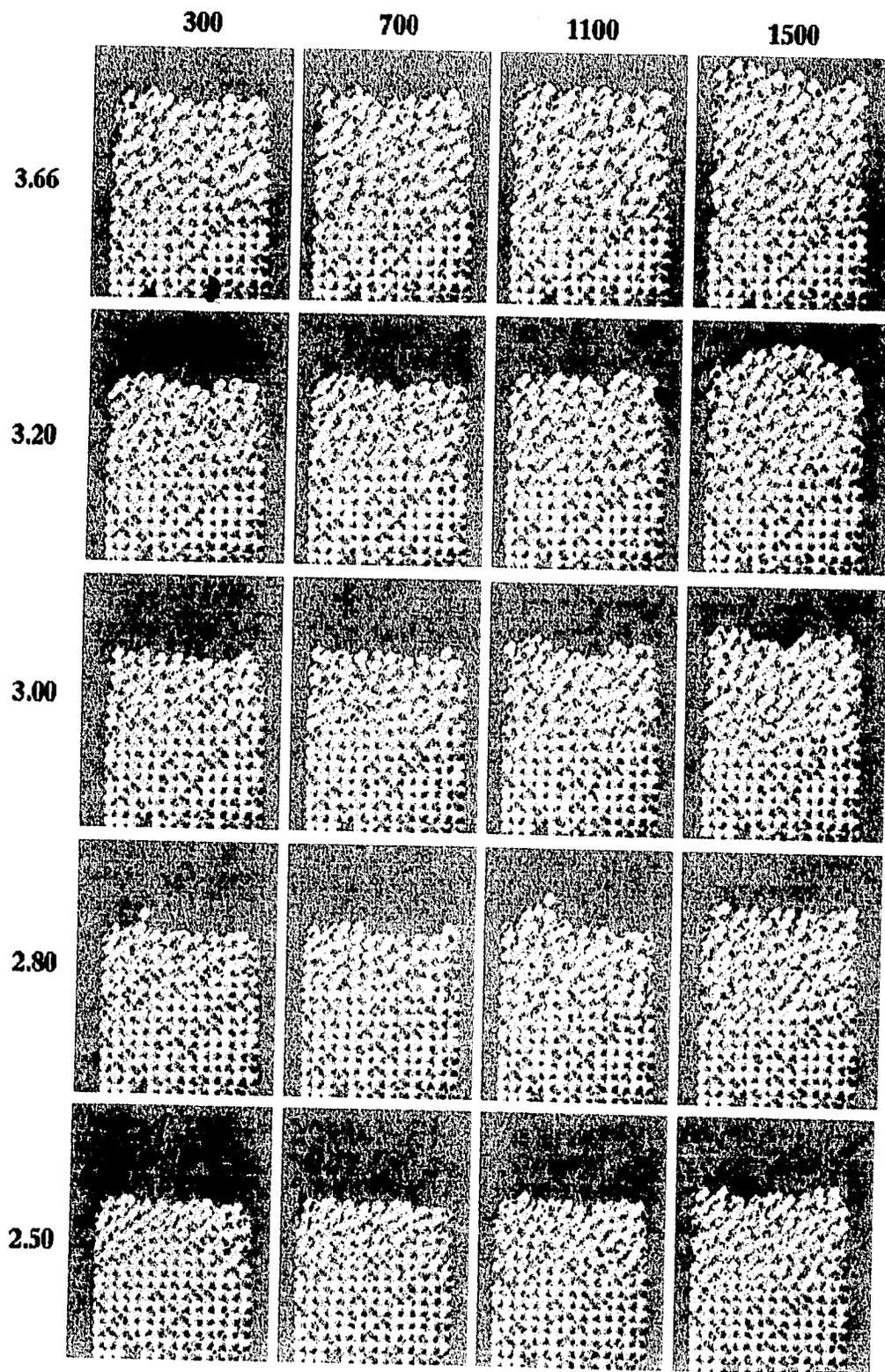


Figura 6.2 Configuraciones de gama alúmina para cinco densidades de 3.66, 3.20, 3.00, 2.80 y 2.50 g cm⁻³ a cuatro temperaturas de 300, 700, 1100 y 1500 K.

7. CONCLUSIONES

Gracias a los avances que las ciencias de la computación han tenido en los últimos años, en cuanto a circuitería y programas se refiere, es que hoy en día se pueden realizar experimentos numéricos complejos que permiten un estudio más amplio de la naturaleza. Gracias a esto, las actuales supercomputadoras arrojan grandes cantidades de información en tiempos muy cortos, lo que trae como consecuencia la búsqueda de herramientas que simplifiquen la tarea de análisis y comprensión de los resultados.

Las estaciones de trabajo han tenido un desarrollo también acelerado en la velocidad de procesamiento de información y en la capacidad gráfica. La necesidad de desplegar grandes cantidades de información con una alta calidad gráfica, son los factores fundamentales por lo cual se desarrolló el sistema en una plataforma de este tipo. *Indigo 2* de *Silicon Graphics* fue la estación de trabajo seleccionada para este propósito, ya que actualmente es una de las mejores estaciones de trabajo para fines de visualización científica, además de que ya se contaba con un número considerable de las mismas.

El lenguaje utilizado fue el "C", debido tanto a sus propias características como a la existencia de bibliotecas y rutinas escritas en este lenguaje.

La visualización se ha convertido en una herramienta necesaria en el campo de la simulación por computadora. Este sistema aporta a los experimentos numéricos de simulación de materiales la facilidad de ver a través de un microscopio imaginario los resultados, con lo cual el investigador no analiza números sino formas y colores, las cuales dan una mayor idea del comportamiento del material final o bien de las etapas intermedias hasta llegar a un estado de interés.

Los resultados mostrados en este trabajo permitieron explicar fenómenos de diferentes estructuras a nivel molecular, sirviendo de apoyo para la sustentación de publicaciones y trabajos de tesis. Sin embargo el camino por recorrer es aún largo, ya que como proyecto a futuro se pretende hacer de este sistema de visualización un sistema que no solo muestre los resultados de las simulaciones, sino que también sea el encargado de enviar procesos a la supercomputadora y recibir los resultados en forma automática, es decir, que el sistema funcione como monitor de la DM y que se pueda "ver" en tiempo real los cambios que va sufriendo la estructura. Esto todavía

no es posible, ya que aunque se posee equipo muy poderoso, es aún insuficiente puesto que una sola configuración de dinámica molecular puede tardarse varios segundos. Pero con las nuevas tecnologías, como por ejemplo las supercomputadoras masivamente paralelas, se puede lograr en un futuro no muy lejano.

El sistema de visualización y análisis desarrollado en esta tesis se ha vislumbrado que puede fácilmente evolucionar hacia un sistema de graficación más sofisticado de aplicación múltiple, utilizando las rutinas ya escritas en un sistema cuya arquitectura esté basada en conceptos de teoría de gráficas y teoría de funciones.

8. BIBLIOGRAFÍA

1. ABRAMOWITZ, M. y STEGUN, I. A. "Handbook of Mathematical Functions". U. S. Department of Commerce. National Bureau of Standards (1964).
2. ÁLVAREZ, L. J., BOSCH, P. y VALENZUELA, M. A. "Molecular dynamics studies of $ZnAl_2O_4$ ". *Catalysis Letters* 22 (1993) 361-372.
3. ÁLVAREZ, L. J., FERNÁNDEZ SANZ, J., CAPITÁN, M. J. y ODRIOZOLA J. A., "Onset of perovskite formation in the catalytic system $La_2O_3/\gamma-Al_2O_3$ ". *Catalysis Letters* 21 (1993) 89-97.
4. ÁLVAREZ, L. J., LEÓN, L. E., FERNÁNDEZ SANZ, J., CAPITÁN, M. J. y ODRIOZOLA, J. A., "Micropore formation mechanisms in $\gamma-Al_2O_3$ ". *Surf. Science* 322 (1995) 185-192.
5. ÁLVAREZ, L. J., VEGA, G. y CORREA, R. "Interstice and cavity analysis algorithm for molecular systems". Artículo sin publicar.
6. CATLOW, C. R. "A. Zeolites: Structure, Synthesis and Properties -An Introduction-". Academic Press, London (1992).
7. CHAN, S. Y. y ELLIOTT, S. R. *Jour. Non-Crystalline Solids*, 124, 22-33 (1990).
8. CRAY RESEARCH. "CF77 & SCC Features and Optimization". Cray Reseach Inc, USA (1992).
9. D. E. KNUTH, "The Art of Computer Programing", Addison-Wesley Publishing Company, Reading Massachusetts (1981).
10. EWALD, P. *Ann Phys.* 64, 253 (1921).
11. FAIRLEY, RICHARD. "Ingeniería de Software". Mc Graw Hill, México (1987).

12. HUEHEE, JAMES E. "QUÍMICA INORGÁNICA principios de estructura y reactividad". Harla, México (1981).
13. KERNIGHAN, BRIAN y RITCHIE, DENNIS. "El lenguaje de programación C". Prentice Hall, México (1991).
14. LEÓN BÁRCENAS, L.E.. "Dinámica molecular de gama alúmina". Tesis de licenciatura. Facultad de Ciencias, UNAM, México, D.F (1995).
15. OVERMARS, MARK. "Forms Library, A graphical user interface toolkit for Silicon Graphics workstations". Department of Computer Science, Utrecht University, the Netherlands (1991).
16. PRESSMAN, ROGER. Ingeniería de Software. Mc Graw Hill, México (1988).
17. SANGSTER, M. J. y DIXON, M. "Interionic potentials in alkali halides and their use in simulations of the molten salts". *Advances in Physics* 1976, Vol. 25, No. 3, 247-342.
18. SILICON GRAPHICS. "Graphics Library Programming Guide". Silicon Graphics Inc, Mountain View, California (1990).
19. YAKOWITS, S. J. "Computational Probability and Simulation". Addison-Wesley Publishing Company, Reading Massachusetts (1977).