

Capítulo 6

Programación del método

El objetivo del programa es estudiar los mecanismos que surgen en la deformación de un nanohueco dentro de un cristal cuando se aplica una deformación cíclica. El programa se divide en dos partes, en la primera se determina la configuración de los átomos a temperatura finita permitiendo que el cristal se expanda libremente y en la segunda, a partir de esta configuración, se introduce el número de cargas que se desean caracterizadas por una matriz de deformación.

6.1. Definición del problema

El problema a resolver es un cubo centrado en el origen en el que se pueden diferenciar dos zonas: la atomística que se sitúa alrededor del nanohueco y la zona continua a medida que nos alejamos. Como se acaba de explicar, el nanohueco se sitúa en el centro del cubo (origen de coordenadas) y el tamaño de su radio, al igual que los tamaños de las zonas atomísticas y continua, se definen como números de celdas unidades.

El archivo principal necesita de dos archivos de entrada, en uno de ellos se definen las constantes de cada material necesarias para definir los potenciales interatómicos. El otro se crea cuando ejecutamos un archivo donde se han definido las características de la red (tamaño de las zonas atomísticas y continua) y el tipo de potencial, Lennard-Jones o Embedded-atom method (si es EAM además hay que indicar qué material es), y genera la siguiente información:

- Tipo de red (BCC o FCC) del cristal y la matriz de red que se utiliza para transformar las coordenadas de la red en coordenadas absolutas.

- Posiciones de todos los nodos y tamaño del cubo. En la zona atomística cada nodo representa un átomo, pero a medida que nos alejamos de ella, cada nodo equivale a cada vez más átomos.
- Condiciones de contorno de cada nodo.

6.2. Expansión térmica

Con los archivos de entrada nombrados anteriormente, ya se puede calcular la configuración del cristal tras establecer la temperatura a la que se encuentra. A modo de resumen, se va a exponer la secuencia con la que se resuelve el problema (para un resumen más amplio consultar anexo I):

- Lee y procesa los archivos de entrada descritos anteriormente.
- Realiza el mallado y se crea el primer archivo de salida.
- Crea el hueco (“nanovoid”) sustrayendo los nodos que están dentro del radio que se ha indicado.
- Se introducen las condiciones de contorno en desplazamientos que se indican en el archivo de entrada.
- Crea un árbol de BSP (Binary space partitioning) para localizar los elementos reduciendo al máximo el coste computacional.
- Crea los clusters: a cada nodo se le asignan los vecinos más cercanos (1º y 2º vecinos) y si hay algún conflicto (un vecino asignado a más de un nodo) tiene preferencia el más cercano. Por último, se calcula el peso de cada nodo.
- Fija la temperatura para todos los nodos.
- Para cada nodo calcula la frecuencia y la entropía en función de la traza de la matriz de rigidez. Para calcular la traza se computa el potencial con los átomos que están dentro del radio de corte.
- Para cada nodo se calculan algunas variables como la energía cinética, la aceleración... teniendo en cuenta la vibración de los átomos. Cálcula la energía total (potencial + cinética) del sistema.
- Con la variante Polak-Ribiere del método de los gradientes conjugados calcula el valor de la frecuencia de cada átomo que minimiza la energía del sistema. Con esta frecuencia calcula de nuevo las variables.

- De nuevo con el mismo método calcula el valor de la energía cinética y de la posición de cada nodo que minimiza la energía para obtener la configuración de equilibrio del sistema. Se vuelven a calcular de nuevo las variables con los resultados obtenidos.
- Calcula la matriz de tensión del virial.
- Produce el archivo final de salida. También produce otro archivo (“restart”) donde se graban todos los datos que son necesarios para poder ejecutar el programa desde este archivo.

6.3. Deformación del cristal

Con el archivo restart que se obtiene en la expansión térmica y el archivo “materials” con las constantes de los potenciales, se ejecuta la segunda parte del programa. Al igual que en la expansión térmica, se va a exponer la secuencia con la que se resuelve el problema (para un resumen más amplio consultar anexo I):

- Lee y procesa el archivo restart. Crea el hueco (“nanovoid”), se introducen las condiciones de contorno en desplazamientos y crea el árbol de BSP (Binary space partitioning).
- Crea los clusters: a cada nodo se le asignan los vecinos más cercanos (1° y 2° vecinos) y si hay algún conflicto (un vecino asignado a más de un nodo) tiene preferencia el más cercano. Por último, se calcula el peso de cada nodo.
- Para cada nodo se calculan algunas variables como la energía cinética, la aceleración... teniendo en cuenta la vibración de los átomos. Calcula la energía total (potencial + cinética) del sistema.
- Se introducen las condiciones de contorno en desplazamientos que se indican en el archivo de entrada.
- Comienza un bucle que se repite para cada una de las deformaciones que se implementa. En cada una de las iteraciones:
 - En el primer paso se crean las matrices de la deformación acumulada y del incremento de deformación como matrices unidad. En los pasos posteriores, se actualizan las matrices y se calculan las nuevas posiciones de los nodos como la posición actual por la matriz de deformación.
 - De nuevo con la variante Polak-Ribiere del método de los gradientes conjugados calcula el valor de la energía cinética, de la temperatura y de la posición de cada nodo que minimiza la energía para obtener la configuración de equilibrio del sistema. Se vuelven a calcular de nuevo las variables con los resultados obtenidos.
 - Calcula la matriz de tensión del virial.

- Produce el archivo de salida y el “restart” de esta iteración.
- Si algún elemento se deforma más que un determinado valor divide el elemento creando nuevos nodos.
- Libera la información almacenada sobre los elementos y el árbol BSP.
- Vuelve a mallar y a crear el hueco.
- Se vuelven a introducir las condiciones de contorno y se crea de nuevo el árbol BSP (Binary space partitioning).
- Crea los clusters de nuevo y actualiza el valor de todas las variables.
- Para cada nodo calcula todas las variables: energía cinética, aceleración...