Enrique Ballesteros Horcajo Práctica 3. Problema 5.

Los comensales

Tenemos una mesa redonda con ‘n’ asientos, en la que se sientan ‘n’ comensales. Cada pareja de comensales comparte una afinidad. En total hay ‘n’ afinidades en la mesa. Se trata de maximizar esta afinidad.

Al ser redonda, comenzamos con el comensal ‘0’ sentado en la posición ‘0’ de la mesa, pues si no, habría n soluciones iguales pero con la mesa girada. A partir de este fijado, cada vez que colocamos a un nuevo comensal actualizamos la afinidad de la mesa (con el que tiene sentado a su izquierda). Está garantizado que puede actualizarse pues siempre se comienza con el comensal ‘0’ ya sentado. Cuando se ha completado la mesa, se actualiza la afinidad del último comensal (el ‘n’) con el primero (el ‘0’), y se compara la solución que se ha obtenido con la mejor hasta el momento.

El caso umbral a partir del cual el tiempo de resolución del problema sin poda comienza a superar los 30 segundos es el de 12 comensales. Por ello será uno de los casos que trataremos.

Las podas usadas se describen en los respectivos archivos de código fuente. Para probarlos se selecciona un número de comensales y se indica un nombre de archivo del que leer la matriz. Se incluye un generador de matrices aleatorias con rango de 1-M y de tamaño N.

La poda 3 es la que mejor funciona, pues su coste por nodo es constante. Resulta curioso, sin embargo, que para funciones de afinidad que agrupan las mayores afinidades entorno a unos pocos valores las podas resultan muy ineficaces. Es el caso de la función afinidad(a, b) = b*a.

Para 13 comensales los tiempos sin poda pasan a ser de 10 minutos. Pero en matrices aleatorias las podas que se actualizan con tiempo constante en cada nodo dan resultados muy buenos. La poda 2 aunque pudiera parecer buena a priori, es menos eficiente en estos casos, aunque tarda menos que sin poda (y visita menos nodos que el resto).

Para 14 comensales la ejecución sin poda ya excede las dos horas.

Con 15 comensales la ejecución sin poda excede las tres horas. Y como los valores de afinidad están en el rango 1-30, se observa que la poda 1 y la poda 3 tienen los mismos tiempos, pues la diferencia entre las ‘n’ mayores afinidades y la mayor de todas es muy pequeña.

Ejecutamos con 16 comensales y un rango de 0-200. Al haber aumentado el rango la poda 3 ahora sí que es mejor que la 1.Para terminar probamos ambas podas con 17.

También ejecutamos para 12 comensales usando la función de afinidad b*a con la que se obtienen datos curiosos.