

Una Interfaz Grafo-Matriz

R. Carballo, C. Escribano, M.A. Asunción Sastre

Dept. Matemática Aplicada
F.Informática. U.P.M.
Boadilla del Monte Madrid, 28660-Madrid
e-mail: cescribano@fi.uib.es

Resumen.

El contexto general es el desarrollo herramientas, con las que el profesor cuente para presentar numerosos *ejemplos* y *algoritmos animados* en el *aula* y a los que el alumno pueda acceder a través de la *página Web* del departamento para experimentar en tiempo real por su cuenta.

El trabajo que presentamos consiste en la elaboración de una aplicación en forma de Applet Java destinada a servir de apoyo para la *enseñanza* en *Teoría de Grafos* basada en una doble visualización de la estructura. Este trabajo forma parte de un objetivo común que varios profesores del Departamento de Matemática Aplicada de la Facultad de Informática en la U.P.M. han puesto en marcha.

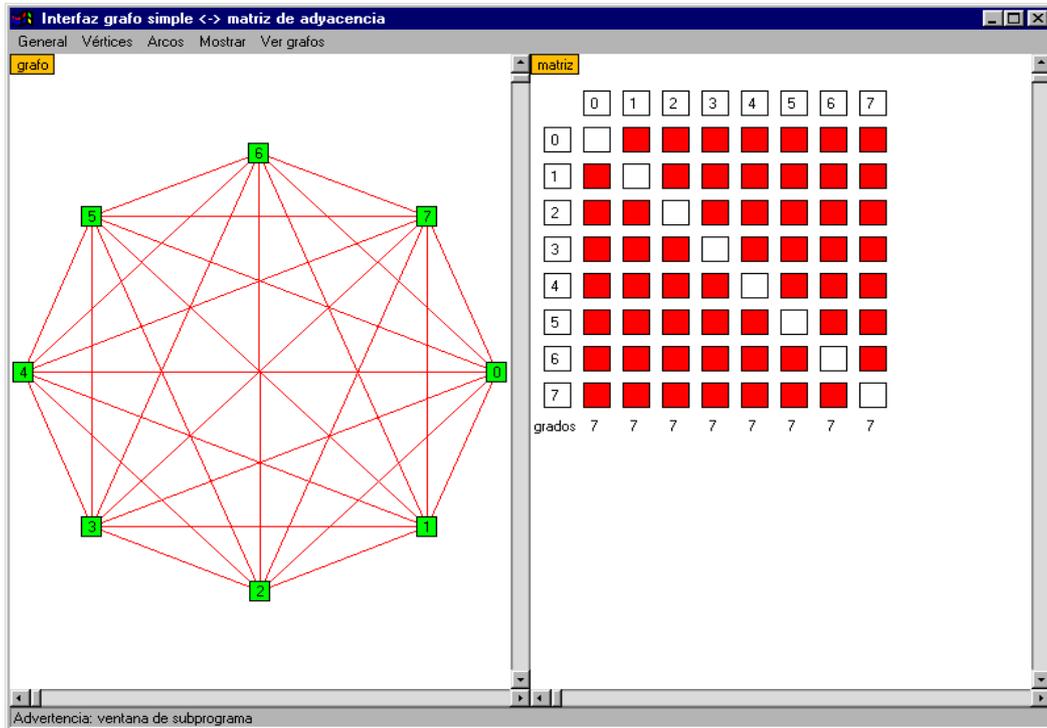
1. Objetivos:

- *Diseño de una interfaz* gráfica de fácil manejo que permite la visualización y manipulación de la estructura combinatoria a través de dos representaciones distintas que *interactúan* mutuamente: la *gráfica* y la *matriz de adyacencia*. La razón es el hecho de que la reordenación de los vértices en la matriz de adyacencia permite visualizar muchas propiedades estructurales del grafo.
- *Implementación de algoritmos* que resuelvan algunos problemas fundamentales en *Teoría de Grafos* como la búsqueda de subgrafos completos, dentro de un grafo dado y estudiar si dos grafos son isomorfos.
- *Facilidad* para que en el futuro se incorporen nuevas funcionalidades y algoritmos.
- *Accesibilidad* desde la *página Web* del departamento.

2. Manual de Funcionalidades.

A continuación se van a explicar todos las ventanas que corresponden al los menús desplegables que hay en ventana principal de la aplicación, Figura 1.

- *Vértices: añadir, borrar o permutar* vértices. A esta ventana hay que añadirla un campo de texto para que el usuario meta la etiqueta del vértice o vértices que desea borrar o permutar. En este menú hay atajos para manipular desde la de zona de dibujo o sobre la matriz. Además se pueden *agrupar circularmente* de manera automática o manual.
- *Arcos: añadir, borrar*. A esta ventana se le añaden dos campos de texto para que el usuario introduzca los dos vértices entre los que desea añadir un arco. También aquí con atajos.
- *Mostrar*: Permite la posibilidad de trabajar con una única ventana que opcionalmente corresponde a: el *grafo*, la *matriz* o sus *complementarios*. También tiene las opciones de trabajar con dos o cuatro ventanas, según se quiera visualizar grafo y matriz, grafo y matriz complementarios o todas las opciones a la vez. Dentro de este desplegable tenemos la opción *grafos independientes* que sirve para introducir dos grafos independientes y ejecutar el algoritmo que resuelve si son o no isomorfos desde el menú *General*.
- *Ver grafos*: despliega una serie de grafos y familias de grafos básicas prediseñadas como: *ciclos*, *ruedas*, *completos*, *bipartidos completos*, etc, con las que el alumno puede familiarizarse enseguida y empezar a experimentar con ellas.

Figura 1. Grafo completo K_7

- General:* en este menú se accede los algoritmos básicos mencionados antes: *Reordenar matriz según grados:* una lista desplegable ofrece al usuario seleccionar un grafo sobre el cual se reordenará la matriz según la lista de grados. *Crear grafo usando la lista de grados:* una lista desplegable ofrece al usuario seleccionar el lugar donde se dibujará el grafo. A continuación se añade un campo de texto para poder introducir la lista de grados. Si existe un grafo simple que la cumpla, aparece en pantalla una solución. Un mensaje de error: se muestra indicando que la lista de grados introducida por el usuario no es gráfica en el caso en que no exista solución. *Crear grafo aleatorio:* se muestra una lista desplegable para que el usuario decida en qué ventana desea dibujar el grafo y se introduce el número de vértices del grafo aleatorio. *Buscar grafo completo:* en una estructura ya

construida se busca el subgrafo completo maximal. El resultado de la búsqueda se muestra en la ventana de grafo completo, Figura 2.. Una ventana de aviso: muestra un texto indicando que no se ha encontrado un grafo completo (el mínimo que se puede encontrar es un K_3) y añade un botón de Aceptar para que una vez pulsado se pueda borrar. *Comprobar isomorfismo:* este algoritmo se ejecuta sobre dos estructuras diferentes que el usuario previamente ha introducido en las ventanas de grafo y grafo independiente. Una ventana de aviso: muestra un texto indicando si los grafos que ha introducido el usuario son isomorfos y añade un botón de Aceptar para que una vez pulsado se borre la ventana *Borrar, guardar, abrir e imprimir:* permiten que el usuario seleccione el grafo que desea eliminar, guardar, abrir e imprimir.

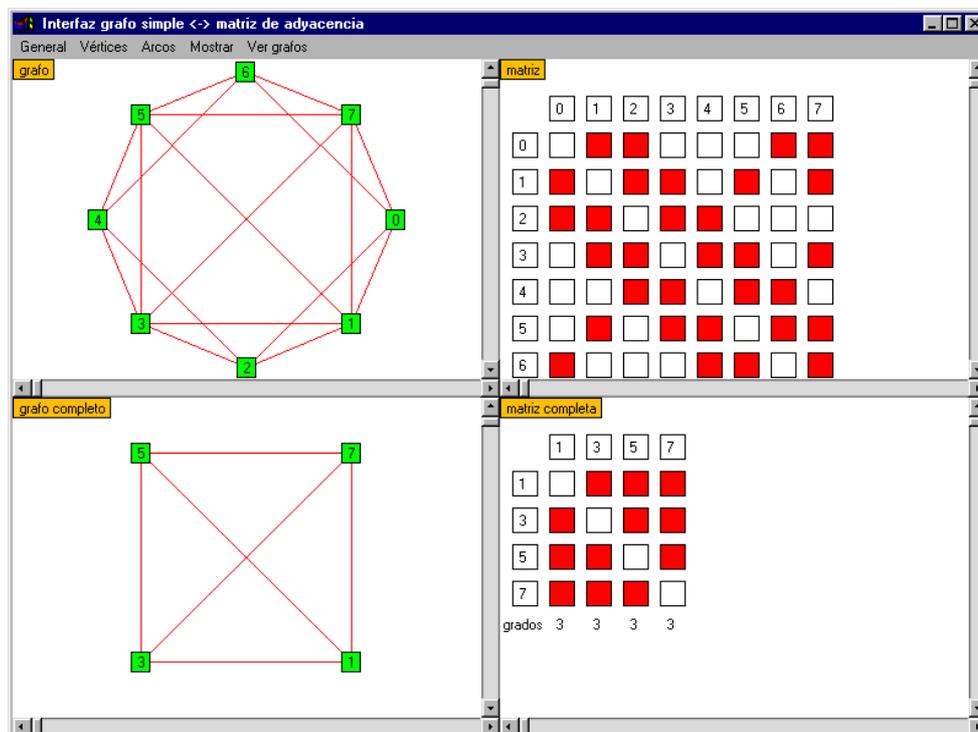


Figura 2. Resultado de la búsqueda del subgrafo completo maximal.

3. Seguridad en las Applets.

Dado que los residentes en un servidor se ejecutan en la máquina cliente, son necesarias unas restricciones de seguridad para prevenir que un applet pueda producir efectos no deseados (virus, caballos de Troya...).

Por tanto, al desarrollar esta aplicación nos encontramos con estos problemas de seguridad a la hora de manejar ficheros y enviar gráficos a la impresora. Las soluciones que hemos considerado para hacer que un applet sea seguro son:

- El JDK 1.3 tiene una aplicación llamada "Policy Tool" que crea un fichero de policía donde debemos indicar los permisos que deseamos conceder al applet. A continuación en el directorio /lib/security hay un fichero de seguridad donde debemos añadir una línea para que tenga en cuenta la política de seguridad que hay que seguir cuando se ejecute nuestro applet y que quedó definida

en el fichero de policía. Este método sólo nos permite ejecutar el applet de manera local.

- Otra opción consiste en obtener un certificado con el que firmar digitalmente el applet.
- Usar un navegador como HotJava que permite definir al usuario la política de seguridad a seguir.

En resumen, Java parece una herramienta adecuada para mejorar las capacidades que anteriormente nos ofrecían las páginas HTML estáticas, facilitando la interacción.

