

Introducción

La capa de enlace de la red tiene una de las funciones más importantes de las redes universales. En primer lugar, se encarga del transporte de paquetes de un enrutador de origen a un enrutador de destino, y en segundo lugar, debe realizar la asignación de una ruta óptima para cada uno de los paquetes, tarea en la cual este proyecto se enfoca. Para realizar el proceso anteriormente nombrado, la capa de enlace de la red utiliza algoritmos de enrutamiento tradicional, los cuales están implementados en todos los routers de la red. El propósito de estos algoritmos no es solo encontrar una ruta entre dos enrutadores, pues también intenta recomendar la ruta menos costosa y con poca congestión, para que el envío de paquetes se realice de forma más ágil.

Uno de los algoritmos más destacados, es el algoritmo de Dijkstra, o algoritmo de la ruta mínima, el cual es muy utilizado en diferentes áreas y es un concepto básico en la Teoría de Grafos. La capa de enlace se puede ver representada como un grafo, donde los nodos representan los enrutadores y las aristas las conexiones. Este proyecto busca emular el algoritmo de ruta mínima y mostrar su función dentro de la capa de enlace.

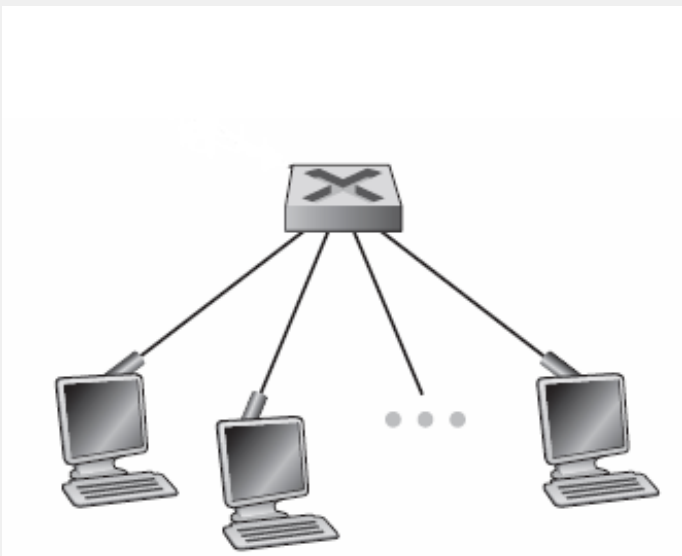


Figure 1. Red de computadores

Estado del arte

La Teoría de Grafos es un campo de estudio amplio, aplicado a muchas situaciones de la vida real, una de estas es la capa de enlace en su plano de control. Debido a lo anterior, es necesario que el lector entienda ciertos conceptos base para la comprensión de este proyecto.

- **Capa de enlace:** Interfaz encargada de los flujos de datos y la asignación de un camino entre el origen y el destino. Utiliza algoritmos de ruta mínima para determinar la ruta óptima entre dos enrutadores, teniendo en cuenta factores de congestión, velocidad y costo [1].
- **Grafo:** Terna de nodos relacionados entre sí por medio de un conjunto de aristas. Son utilizados para representar relaciones entre nodos [2]. En el caso de la capa de enlace, representa una red de conexiones de enrutadores.
- **Grafo dirigido:** Terna de nodos relacionados entre sí por medio de un conjuntos de aristas de dirección única. A estas aristas se le conocen como aristas de entrada o de salida, dependiendo de su función en cada nodo [2].
- **Grafo conexo:** Se dice que un grafo es conexo si cualquier par de nodos está conectado por un camino [3].
- **Camino en un grafo:** Conjunto de vértices y aristas, sin repetición, en el cual se representa un recorrido de un punto a otro en una red o grafo [3].
- **Algoritmo de Dijkstra:** Este algoritmo encuentra el camino de coste mínimo entre un nodo fuente y un nodo de destino [4].
- **Tabla de enrutamiento:** Es la encargada de contener la información sobre varias rutas entre dispositivos para presentar las rutas más eficientes para los paquetes de datos.

- **Árbol de enrutamiento:** Subgrafo de una red constituido por un subconjunto de aristas, las cuales representan el camino mínimo entre un nodo de origen y los demás nodos de la red. Es considerado como un resultado del algoritmo de Dijkstra. Cada nodo tiene un padre al cual enviar y retransmitir los datos recibidos. De esta manera, el nodo origen encuentra con facilidad el camino a recorrer [5].

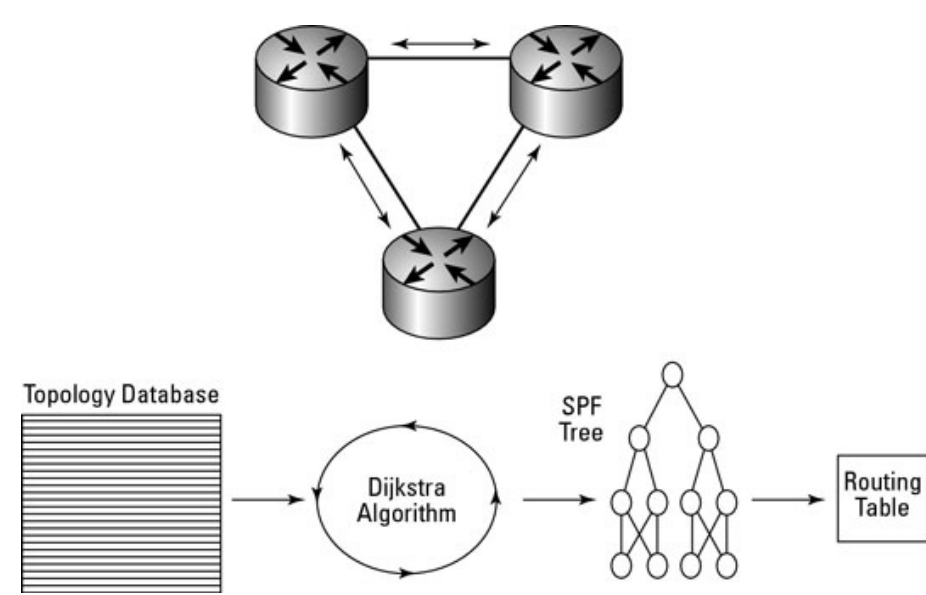


Figure 2. Red y árbol.

Problema e implementación.

El proyecto realizado busca implementar el algoritmo de Dijkstra o de ruta mínima. El usuario tiene la posibilidad de cargar un archivo .csv con la matriz dispersa de un grafo, o seleccionar la creación de un grafo aleatorio. Para aquello, se definió una función que le pregunta al usuario si desea usar su propio grafo o uno generado, recibiendo el número de nodos. Con esta información se genera la matriz dispersa y se genera una visualización del grafo.

```
[[ '1'  '2'  '10' ]  
 [ '1'  '3'  '50' ]  
 [ '2'  '3'  '20' ]  
 [ '2'  '4'  '20' ]  
 [ '2'  '5'  '10' ]  
 [ '3'  '5'  '20' ]  
 [ '4'  '5'  '30' ]  
 [ '4'  '6'  '10' ]  
 [ '5'  '6'  '20' ] ]
```

Figure 3. Matriz dispersa generada.

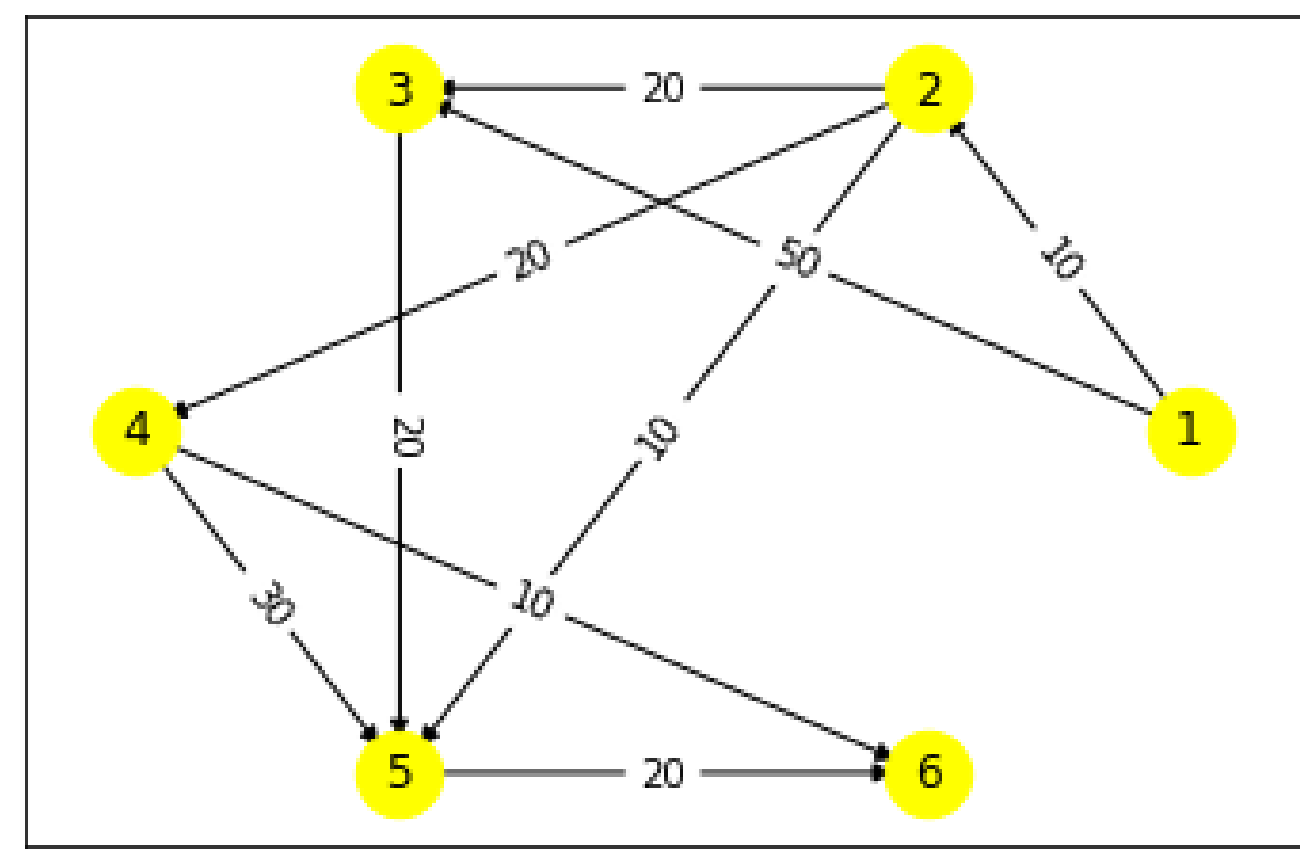


Figure 4. Grafo generado.

Posteriormente, se ejecuta el algoritmo de Dijkstra, calculando la ruta mínima entre un nodo de origen y el resto de nodos del grafo. La variante del algoritmo de Dijkstra que el grupo de trabajo implementó se caracteriza por ir vaciando la lista de nodos, en vez de ir llenándolo, como se vio en clase, además, para generar el camino de un nodo a otro, se retrocede desde el nodo de destino hasta el nodo de origen, ya que en una lista se almacenan los predecesores de cada uno de los nodos del grafo. Posteriormente, se genera el árbol resultante del algoritmo de Dijkstra, tal como se aprecia en la figura número cinco.

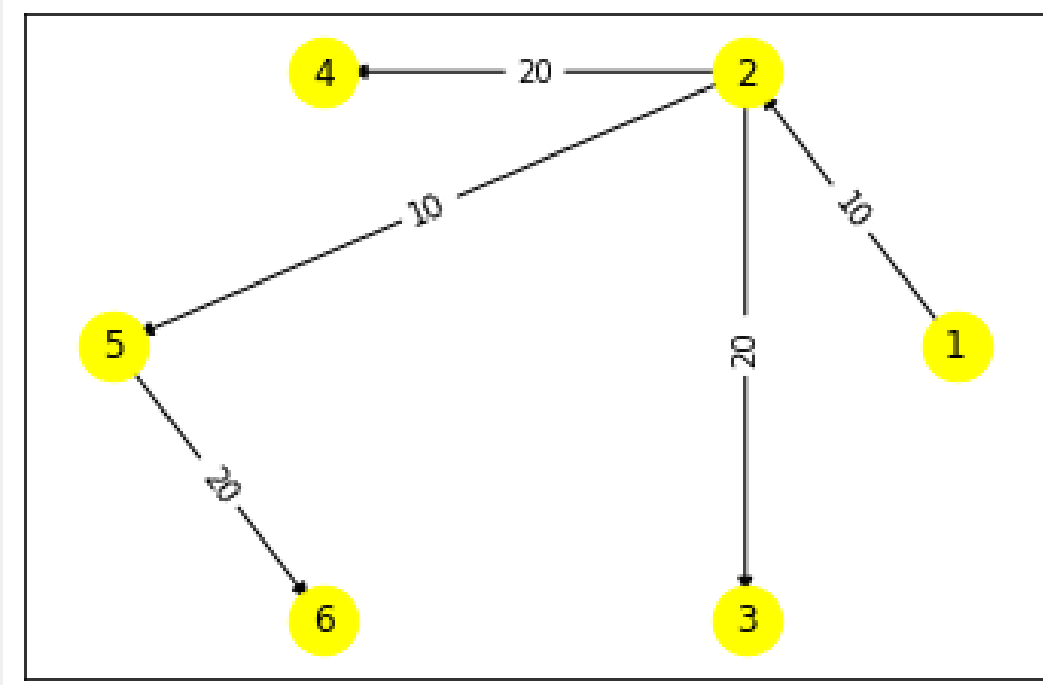


Figure 5. Árbol resultante

Finalmente, se genera la tabla de enrutamiento final para el grafo, a partir de su primer nodo. Junto con esto, se le informa al usuario acerca de los tiempos de ejecución y el número de iteraciones que realiza la implementación del algoritmo de Dijkstra. Cabe señalar que, todos los procesos anteriormente descritos funcionan tanto en caso de generar un grafo aleatorio, como si el usuario aportara la información de este.

Nodo	Distancia	Antecesor
1	0	1
4	30	2
6	40	5
3	30	2
2	10	1
5	20	2

Figure 6. Tabla de enrutamiento resultante.

Resultados y conclusiones

Como se puede evidenciar, el proyecto cumple con el objetivo planteado de emular el funcionamiento de la capa de enlace mediante el algoritmo Dijkstra para encontrar la ruta mínima de la red y así poder optimizarla, ahorrando costes en infraestructura de conexiones físicas y tiempos de envío/respuesta. El grupo de trabajo logró evidenciar y visualizar lo fundamental que son estos tipos de algoritmos para el correcto funcionamiento de la capa de enlace, y la gran aplicación que tiene la Teoría de Grafos sobre esta. A su vez, se demuestra la gran utilidad de los árboles y tablas de enrutamiento para la comprensión gráfica de las rutas mínimas a partir de un nodo base de la red.

Referencias y Código

- 1 Rountree, D., 2011. Data link layer. [online] ScienceDirect. Available at: <https://www.sciencedirect.com/topics/computer-science/data-link-layer>.
- 2 Graph Everywhere, n.d. Qué son los grafos. [online] GraphEverywhere. Available at: <https://www.grapheverywhere.com/que-son-los-grafos/>.
- 3 Universidad Politécnica de Madrid, n.d. Nociones Básicas. [online] Escuela Técnica Superior de Ingenieros Informáticos, Universidad Politécnica de Madrid. Available at: http://www.dma.fi.upm.es/personal/gregorio/grafos/web/caminos_minimos/teoria/teoria.htm.
- 4 Kurose J., Ross. K. Redes de Computadoras. Un enfoque descendente. 7ma edición. https://www.academia.edu/40738627/Redes_de_computadoras_Un_enfoque_descendente_7a_Edici%C3%B3n
- 5 Romero, E., Martínez, G. and Rico, D., 2021. Árbol de caminos mínimos: Enrutamiento, algoritmos aproximados y complejidad. [ebook] Bucaramanga, Santander: Universidad Manuela Beltrán, p.19. Available at: https://revistas.unipamplona.edu.co/ojs_viceinves/index.php/RCTA/article/view/2780/1505.