

Instituto Tecnológico de Culiacán

Carrera: Ing. Tecnologías de la información y comunicaciones

TOPICOS DE INTELIGENCIA ARTIFICIAL

DR. ZURIEL DATHAN MORA FELIX

Grupo: 12:00 – 01:00 PM

Unidad 2

Tarea 1 Investigación métodos de heurística

Alumno: Rodrigo Alonso Páez Gastélum

Número de control: 20170080

# Problema de programación de trabajos (JSSP)

## Descripción detallada del problema

El Job-Shop Scheduling Problem (JSSP) y la asignación de operarios son dos problemas NP-Hard de optimización ampliamente estudiados. En este proyecto se propone una metodología para resolver la integración de estos dos problemas, dentro del contexto de las cocinas ocultas, priorizando el JSSP mediante la solución individual de este problema implementado un algoritmo genético y posteriormente la solución a la asignación de operarios con un modelo de optimización lineal tomando como insumo de conjuntos y parámetros la solución encontrada en el algoritmo genético. La metodología incluye la creación de una herramienta integral en donde inicialmente se genera una recepción y transformación de información de entrada, posteriormente la ejecución de los modelos de solución para ambos problemas y finalmente una visualización de la programación y la asignación. Se implementó el proyecto en la operación llevada a cabo en el centro de producción de Foodology SAS teniendo en cuenta instancias de tamaño real que se realizan diariamente, y los parámetros reales como maquinas, productos y tiempos de procesamiento. (Morales Gómez, 2022)

## Representación

El Algoritmo de Prim consiste en incrementar el tamaño de un árbol, empezando por un vértice inicial al cual se le van agregando sucesivamente vértices cuya distancia a los anteriores es mínima. Esto quiere decir que en cada paso, las aristas a considerar son aquellas que inciden en vértices que ya pertenecen al árbol. Por tanto, el árbol recubridor mínimo estará completamente construido cuando no queden más vértices por agregar. Los requisitos para proceder con el Algoritmo de Prim son:

* Grafo conexo
* Tener todos los arcos etiquetados

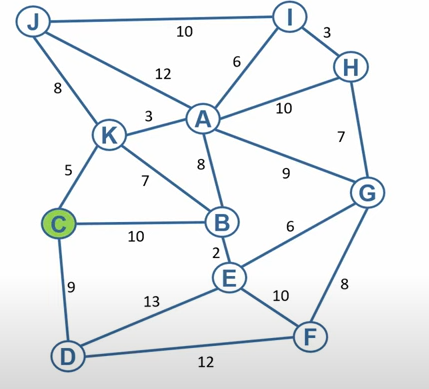
La idea fundamental consiste en añadir en cada paso una arista de peso mínimo a un árbol previamente construido. Para verlo de un modo más explícito, se definen los siguientes pasos:

1. Se elige un vértice U de G y se considera el árbol S={U}.
2. Se considera la arista e de mínimo peso que une un vértice de S y un vértice que no es de S, y se hace S=S+e .
3. Si el número de aristas de T es n-1, el algoritmo termina. En caso contrario, volveríamos al paso 2.

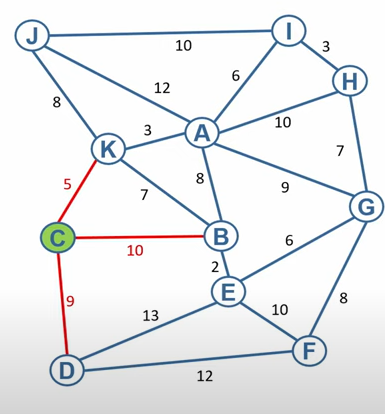
Ejemplo: Red de tuberías



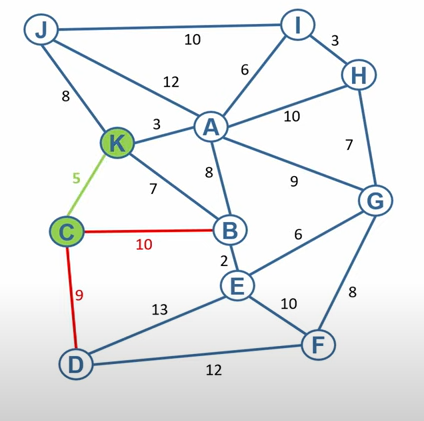
Se parte de un nodo elegido aleatoriamente



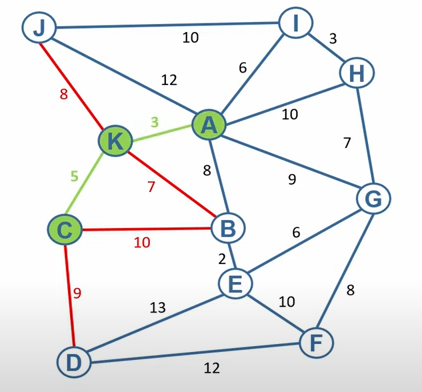
Pasar por todas las aristas posibles a otros nodos

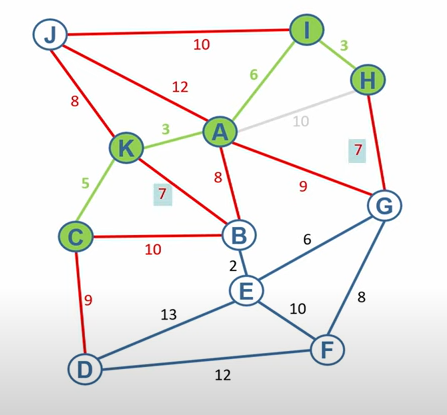


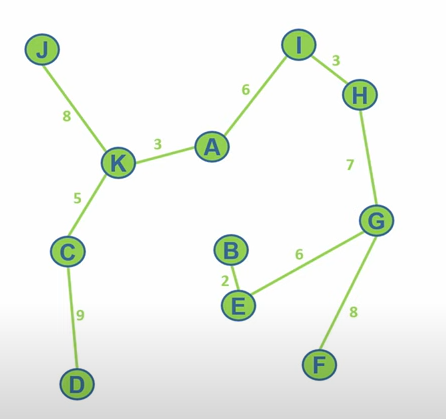
Se elige el menor



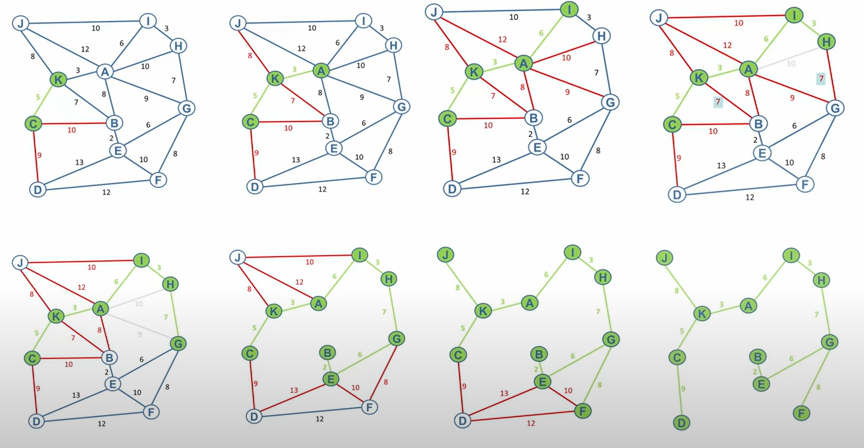
Se repite el proceso







Total de las iteraciones

 (UNIVERSIDAD POLITECNICA DE VALENCIA, Cristóbal Miralles, 2022)

# Problema de las N reinas

## Descripción detallada del problema

El problema de las ocho reinas es un pasatiempo que consiste en poner ocho reinas en el tablero de ajedrez sin que se amenacen. Fue propuesto por el ajedrecista alemán Max Bezzel en 1848. En el juego del ajedrez la reina amenaza a aquellas piezas que se encuentren en su misma fila, columna o diagonal. El juego de las 8 reinas consiste en poner sobre un tablero de ajedrez ocho reinas sin que estas se amenacen entre ellas. Para resolver este problema se puede emplear un esquema vuelta atrás (o Backtracking).

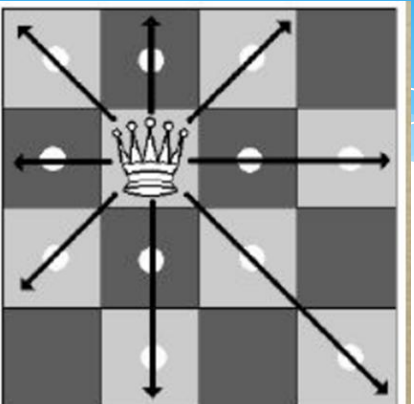
El problema fue originalmente propuesto en 1848 por el ajedrecista Max Bezzel.​ Durante años, muchos matemáticos, incluyendo a Carl Friedich Gauss, han trabajado en él y lo han generalizado a n-reinas. Las primeras soluciones fueron ofrecidas por Franz Nauck en 1850.​ Nauck también se abocó a las n-reinas (en un tablero de nxn de tamaño arbitrario). En 1874, S. Günther propuso un método para hallar las soluciones usando determinantes, y J.W.L. Glaisher redefinió su aproximación.

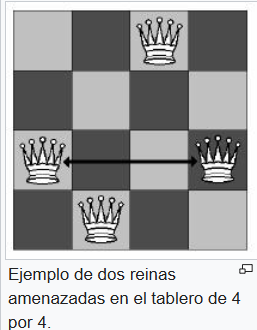
Edsger Dijkstra usó este problema en 1972 para ilustrar el poder de la llamada programación estructurada. Publicó una descripción muy detallada del desarrollo del algoritmo de backtracking, "depth-first".

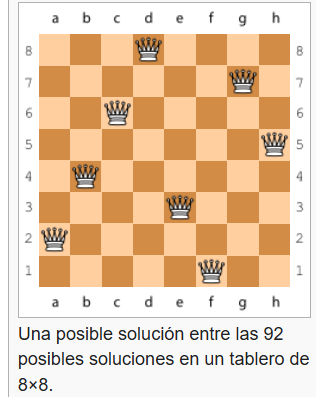
## Representación

Como cada reina puede amenazar a todas las reinas que estén en la misma fila, cada una ha de situarse en una fila diferente. Podemos representar las 8 reinas mediante un vector [1-8], teniendo en cuenta que cada índice del vector representa una fila y el valor una columna. Así cada reina estaría en la posición (i, v[i]) para i = 1-8.

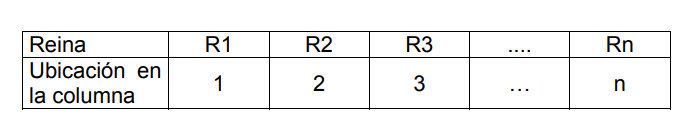
Posibles movimientos de una reina







El problema de las n-reinas consiste en colocar n reinas en un tablero de ajedrez de n x n de tal manera que no sea posible que dos reinas se capturen entre si, es decir, que no estén en la misma fila, ni en la misma columna ni en la misma diagonal. Se dice que hay una colisión si hay dos reinas que se pueden capturar entre si. Se trata pues de encontrar una configuración – elegir las n celdas donde colocar a las reinas- que minimice el número total de colisiones. Una solución tiene la forma de un arreglo n-dimensional:(r1,r2,r3,......rn)

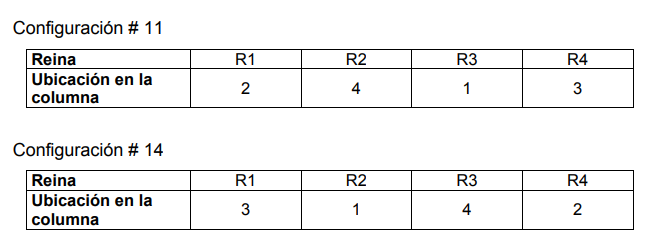


Por ejemplo una solución para n = 4 es (3,4,1,2) Matricialmente se representa:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | columna 1 | columna 2 | columna 3 | columna 4 |
| Reina 1 |  |  | q1 |  |
| Reina 2 |  |  |  | q2 |
| Reina 3 | q3 |  |  |  |
| Reina 4 |  | q4 |  |  |

Hay 4 colisiones {(1,2) (3,4) (1,3) (2,4,)}

Este problema puede resolverse de varias formas, por ejemplo enumerando todas las posibles alternativas y evaluando si se producen colisiones, en cuyo caso se tendría que evaluar factorial de n posibles soluciones. Cuando n es igual a 4 la cantidad de soluciones alternativas es de 24 y como se puede ver en el Anexo 4.1 hay 2 configuraciones que producen cero colisiones:



Observar que si se utiliza un método de búsqueda local, como el greedy, y suponiendo que como punto de partida se asigna la reina 1 a la columna 1 (ver las seis primeras alternativas en el anexo 4.1) aunque se permute exhaustivamente las otras 3 sólo se consigue un óptimo local, es decir el mínimo de colisiones posibles es una colisión y ya no se podría mejorar. (queda atrapado en un óptimo local), es decir si se fija la reina 1 en la columna 1 nunca se encontrará una configuración con cero colisiones.

Se tendría que volver a utilizar el método greedy tomando como punto de partida a la reina 2 en la columna 1 para encontrar cero colisiones en la solución (2,4,3,1). Cabe anotar que en los métodos de búsqueda heurísticos no se dispone de teoremas como el de Khun Tucker para la programación matemática, con el que es posible determinar si una solución dada es óptima global.

# Árbol de expansión mínima (MST)

## Descripción detallada del problema

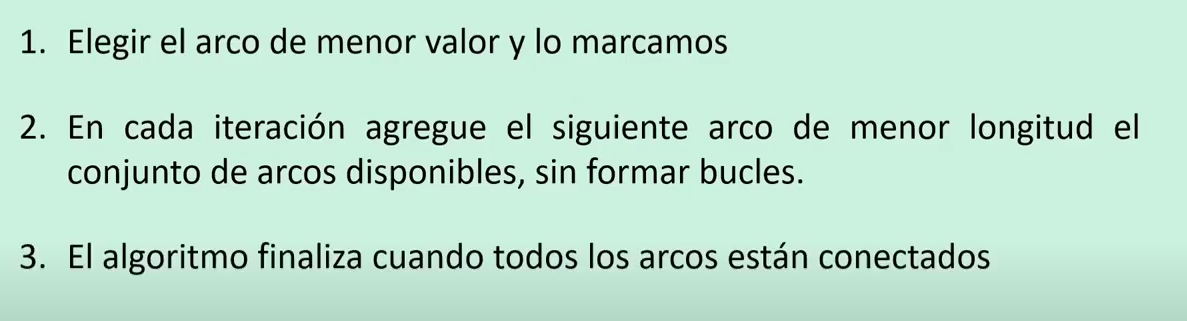
El algoritmo del árbol de expansión mínima es un modelo de optimización de redes que consiste en enlazar todos los nodos de la red de forma directa y/o indirecta con el objetivo de que la longitud total de los arcos o ramales sea mínima (entiéndase por longitud del arco una cantidad variable Según el contexto operacional de minimización, y que puede bien representar una distancia o unidad de medida). El árbol de expansión mínima es apropiado para problemas en los cuales la redundancia es expansiva, o el flujo a lo largo de los arcos se considera instantáneo. El problema surge cuando todos los nodos de una red deben conectarse entre ellos sin formar un ciclo. La aplicación de estos problemas de optimización se ubica en las redes de comunicación eléctrica, telefónica, carretera, ferroviaria, aérea, marítima, hidráulica o de gas, etc. donde los nodos representan puntos de consumo eléctrico, teléfonos, aeropuertos, computadoras y los arcos podrían ser de alta tensión, cable de fibra óptica, rutas aéreas, agua, gas etc.. También se le conoce como árbol generador mínimo, es una red conexa y ponderada que se refiere a utilizar los arcos de la red para llegar a todos los nodos de esta, de manera tal que se minimiza la longitud total. Para su solución se emplean los algoritmos de Prim y Kruskal.

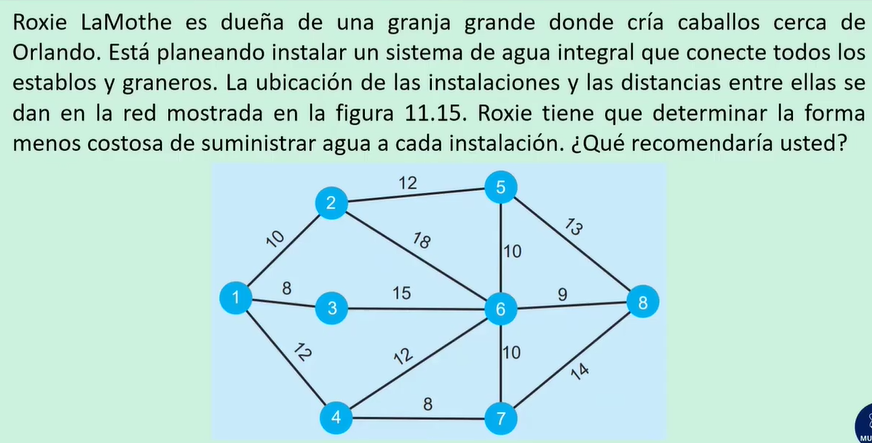
¿Qué es el árbol de expansión de peso mínimo?

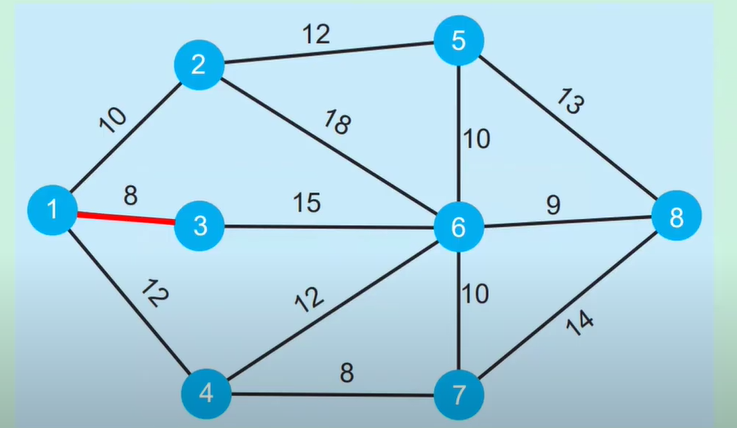
Antes de abordar a profundidad el funcionamiento de este importante algoritmo, descubramos un poco sobre su historia. Esté algoritmo diseñado para diseñar un árbol de expansión mínima fue desarrollado por el científico checo Otakar Boruvka en 1926. Este algoritmo nace en el intento de encontrar una red eléctrica que fuese eficiente para Moravia. A partir de este algoritmo nace lo que conocemos comoárbol de expansión de peso mínimoque comienza desde un vértice especificado dentro de un grafo y encuentra todos los vértices a los que tiene accesibilidad determinando el conjunto de relaciones que conectan los nodos con un valor de peso del menor tamaño posible. Un algoritmo de árbol de mínima expansión (MST, por sus siglas en inglés, "Minimum Spanning Tree") es un procedimiento utilizado en teoría de grafos y redes para encontrar un subconjunto de aristas (o conexiones) que conectan todos los nodos (o vértices) de un grafo sin ciclos y con el peso total mínimo posible.

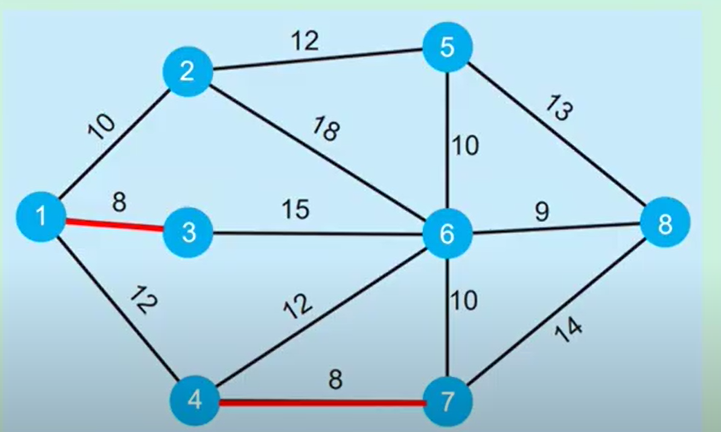
## Representación

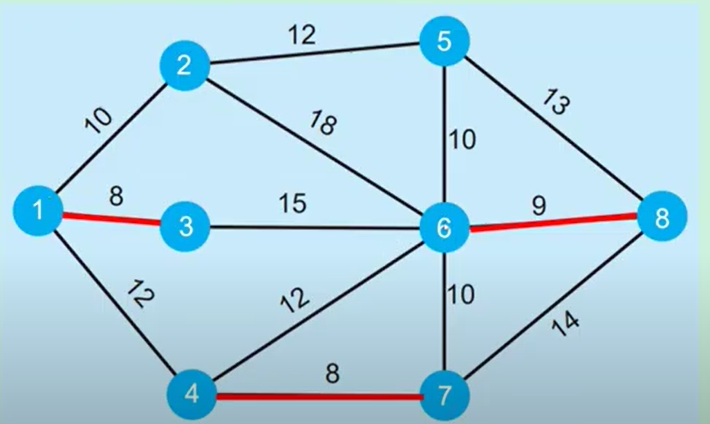
Algoritmo de Kruskal

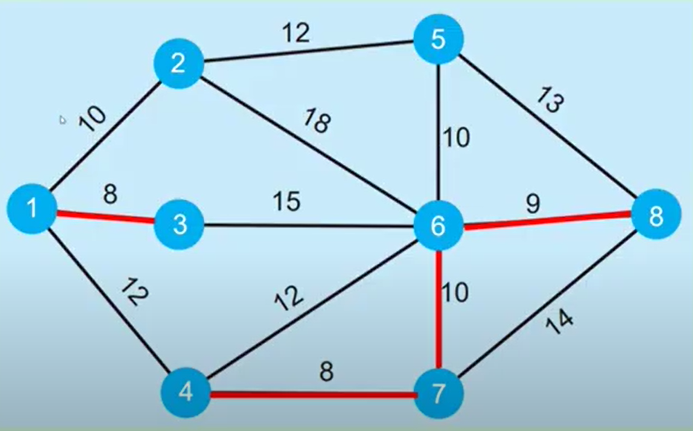


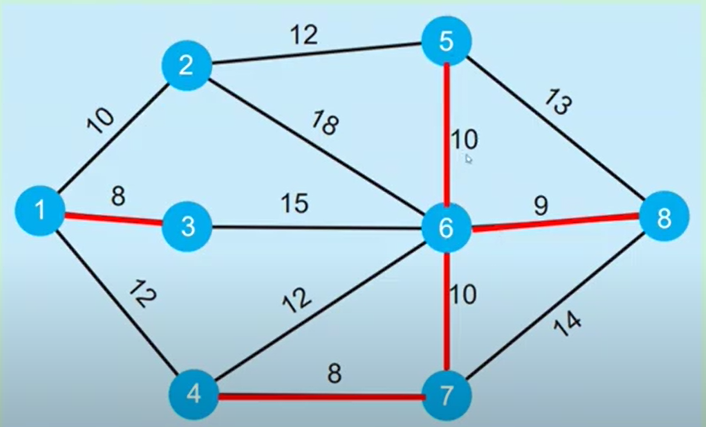


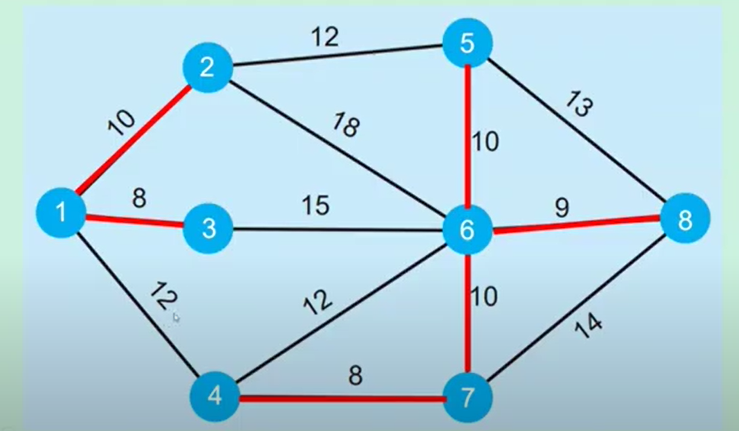












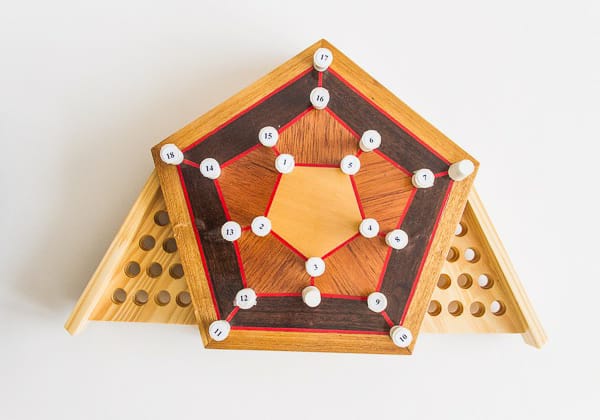
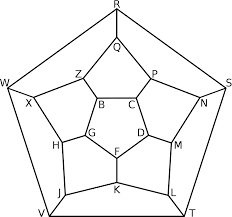
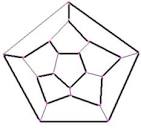
 (Mundo IO Academy, Cristian Rojas, 2022)

# Problema del agente viajero (TSP)

## Descripción detallada del problema

El problema del vendedor viajero o problema del viajante, por sus siglas en ingles “travelling salesman problem” responde a la siguiente pregunta: dada una lista de ciudades y las distancias entre cada par de ellas ¿Cuál es la ruta más corta que visita cada ciudad exactamente una vez y al finalizar regresar a la ciudad de origen?

Es un problema de teoría de la complejidad computacional dentro de optimización combinatoria. Fue planteado en 1930 y es uno de los problemas de optimización mas estudiados. La primera formulación matemática fue realizada en el siglo XIX por W.R. Hamilton y Thomas Kirkman. El juego icosiano de Hamilton era un rompecabezas recreativo basado en encontrar un ciclo hamiltoniano, que en realidad es una solución al problema de TSP en un grafo (nota: el TSP es el ciclo hamiltoniano de menor longitud en un grafo conexo).



El TSP se puede formular como un modelo de Programación Lineal Entera. Aunque se conocen varias formulaciones para resolver este problema, dos de ellas son las más destacadas: la propuesta por Miller, Tucker y Zemlin, y la propuesta por Dantzig, Fulkerson y Johnson.

Con el fin de obtener buenas soluciones en un tiempo más corto, se ha realizado un gran esfuerzo para intentar resolver este problema con una variedad de métodos heurísticos. De todo este grupo de heurísticas, nos gustaría destacar las inspiradas en la biología: Optimización por Colonia de Hormigas y Algoritmos Genéticos (GA).

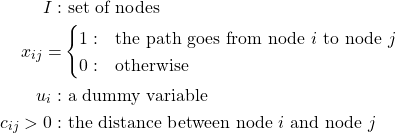
## Representación

Programación Lineal Entera

Como mencionamos, existen dos formulaciones principales para el TSP, el propuesto Miller, Tucker y Zemlin (MTZ) y el Dantzig, Fulkerson y Johnson (DFJ). Aunque la formulación DFJ es más fuerte, la formulación MTZ puede ser útil.

#### **Formulación MTZ**

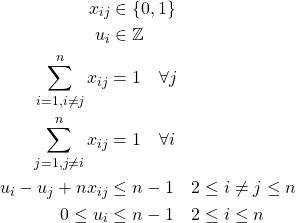
La formulación matemática propuesta por Miller, Tucker y Zemlin es la siguiente:



Usando este conjunto, las dos variables y el parámetro, podemos formular el problema como:



Sujeto a:



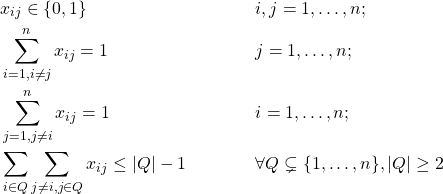
La primera y la segunda ecuación refuerzan el tipo de las diferentes variables, la tercera y la cuarta ecuaciones aseguran que cada nodo sea alcanzado y abandonado solo una vez, mientras que las dos últimas ecuaciones imponen que solo una ruta cruce todos los nodos.

Formulación DFJ

En la formulación propuesta por Dantzig, Fulkerson y Johnson partimos de la misma variable x\_ {ij} y parámetro c\_ {ij}> 0 para que la formulación se pueda completar con:

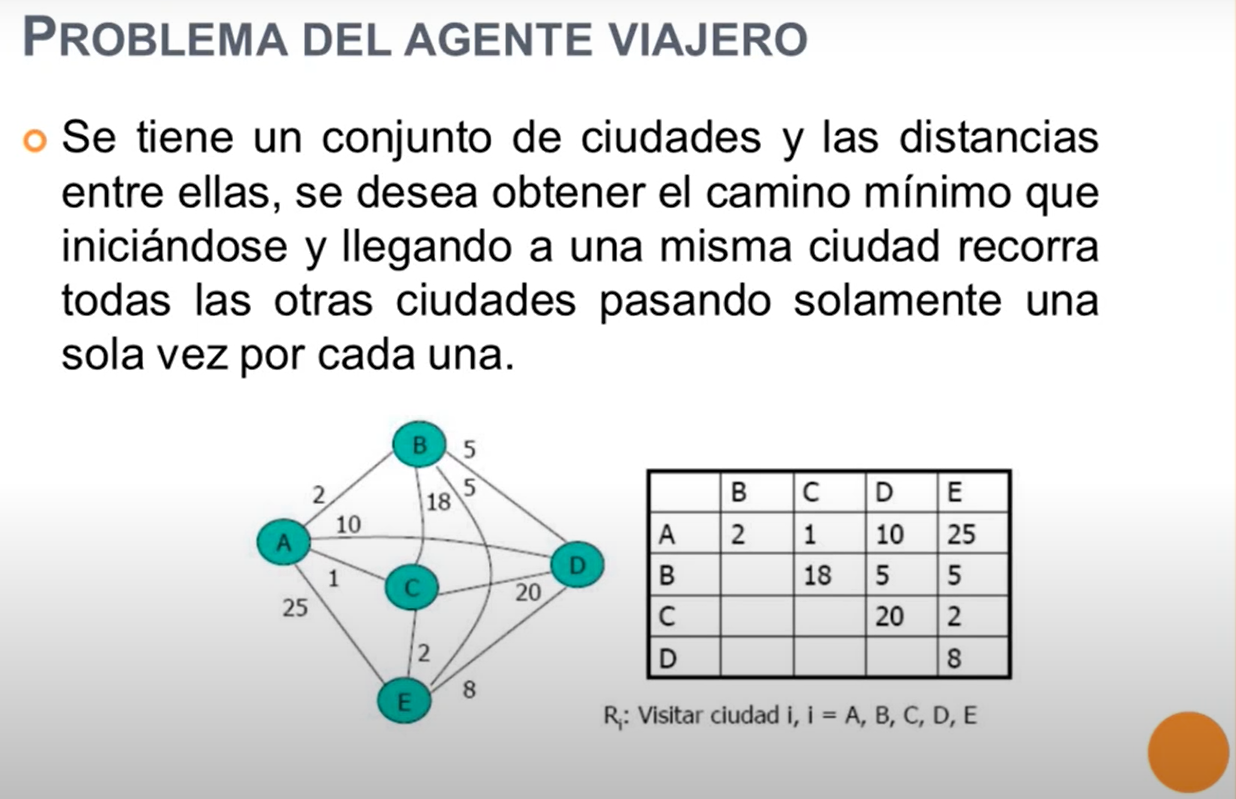


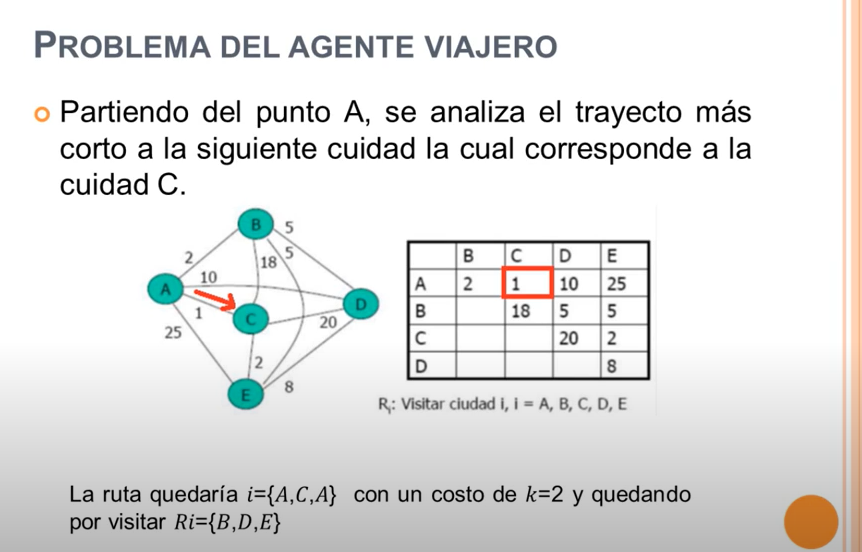
Sujeto a:

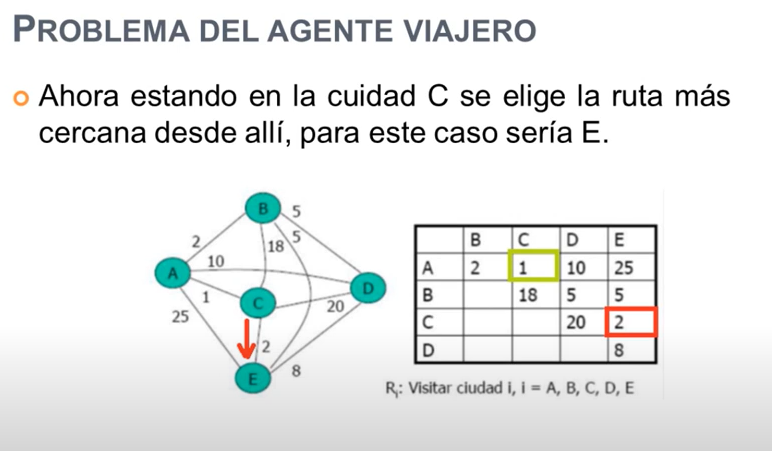


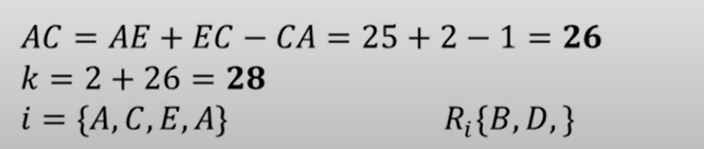
Las tres primeras ecuaciones son las mismas que en la formulación anterior, y la nueva restricción (la última) asegura que no haya sub-recorridos, por lo que la solución devuelta es un recorrido único y no la combinación de recorridos más pequeños. Debido a que esto conduce a un número exponencial de posibles restricciones, en la práctica se resuelve con una generación de columnas retrasada.

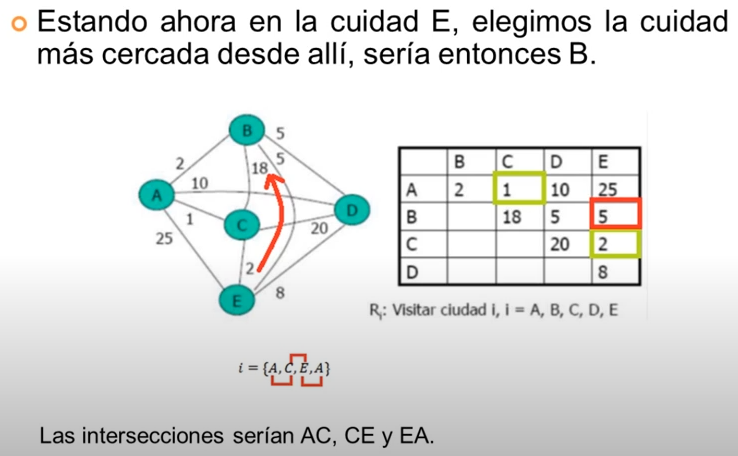
….

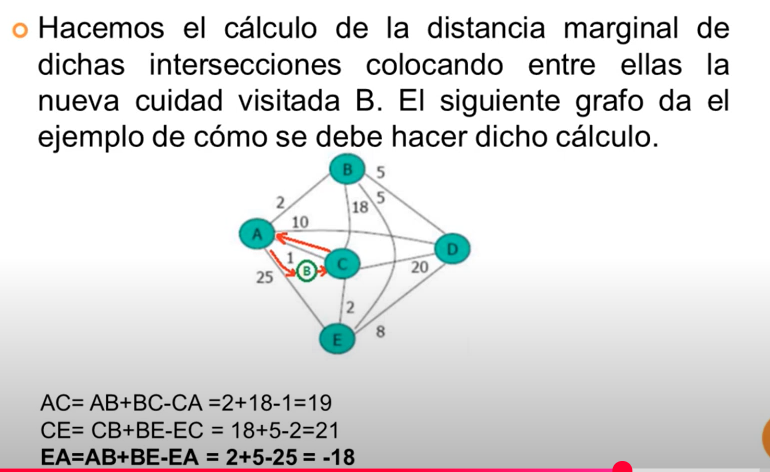


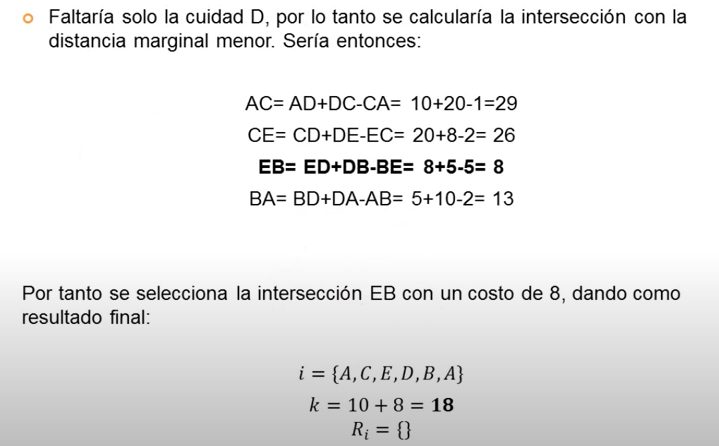


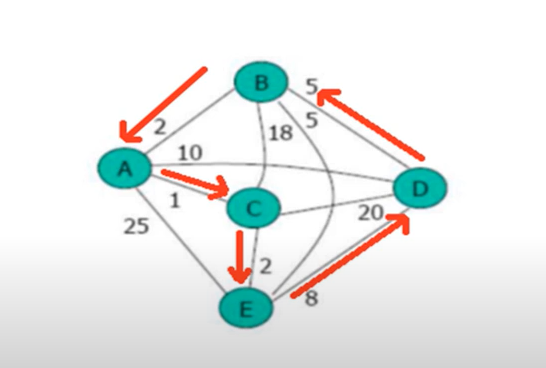












# Referencias

Baobab soluciones. (10 de OCTUBRE de 2020). *Tres métodos diferentes para resolver el problema del viajante*. Obtenido de https://baobabsoluciones.es/: https://baobabsoluciones.es/blog/2020/10/01/problema-del-viajante/#:~:text=El%20problema%20del%20viajante%20(por,Su%20origen%20no%20est%C3%A1%20claro.

Fernandez, E. A. (30 de NOVIEMBRE de 2013). *Problema del Agente Viajero*. Obtenido de Youtube: https://www.youtube.com/watch?v=ZnQtzQFTwfA

Morales Gómez, M. P. (1 de DICIEMBRE de 2022). *Problema integrado de Job-Shop Scheduling con asignación y programación de operarios*. Obtenido de Problema integrado de Job-Shop Scheduling con asignación y programación de operarios: https://repositorio.uniandes.edu.co/entities/publication/1b6d6242-5043-4369-85f2-c75ae483c130

Mundo IO Academy, Cristian Rojas. (13 de FEBRERO de 2022). *Árbol de Expansión Mínima - Algoritmo de Kruskal*. Obtenido de Youtube: https://www.youtube.com/watch?v=lTCDUJw\_4GM

UNIVERSIDAD POLITECNICA DE VALENCIA, Cristóbal Miralles. (20 de MARZO de 2022). *Grafos: árbol parcial mínimo con algoritmo de PRIM | | UPV*. Obtenido de Youtube.com: https://www.youtube.com/watch?v=6pPSPYUbTlw

Wikipedia. (15 de Agosto de 2024). *Problema del viajante*. Obtenido de Wikipedia: https://es.wikipedia.org/wiki/Problema\_de\_las\_ocho\_reinas

wikipedia. (29 de enero de 2025). *es.wikipedia*. Obtenido de wikipedia: https://es.wikipedia.org/wiki/Problema\_del\_viajante

www.cliffsnotes.com. (25 de MAYO de 2024). *cliffsnotes*. Obtenido de Algoritmo de árbol de mínima expansión-Problemas de la ruta más corta: https://www.cliffsnotes.com/study-notes/17040135#:~:text=Un%20algoritmo%20de%20%C3%A1rbol%20de,el%20peso%20total%20m%C3%ADnimo%20posible.