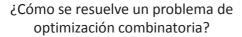
Undécima Semana

Resolución aproximada de problemas de Programación Lineal Entera (Heurísticas)

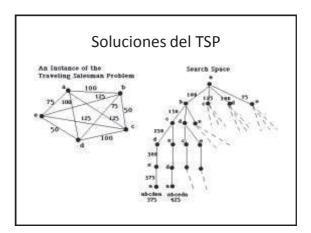
Modelos y

Algunos problemas de optimización combinatoria

- Problema de la mochila
- Problema del viajante de comercio
- Diseño de redes de comunicaciones
- VIS
- Asignación de recursos y horarios
- Secuenciamiento de ADN







Complejidad



¿Construcción o mejora?





Coloreo

Modelo matemático

Minimizar

$$\sum_{i=1}^n w_j$$

Sujeto a

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \forall i \in V$$

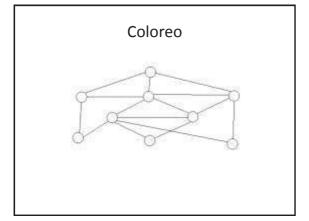
 $x_{ij} + x_{kj} \leq w_j \quad si\left[i,k\right] \in E \ \, \forall j = 1, \dots, n$

 $x_{ij}\,,w_j\in\{0,1\}\quad\forall i\in V\quad\forall j=1,\dots,n$

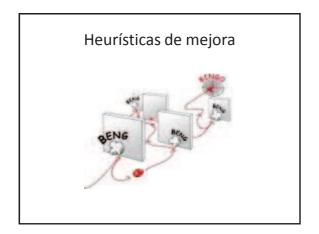
Aplicaciones prácticas

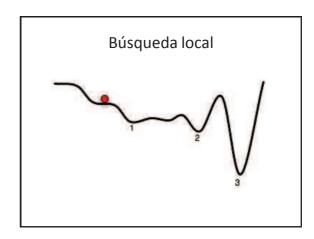
Aplicaciones prácticas de coloreo:

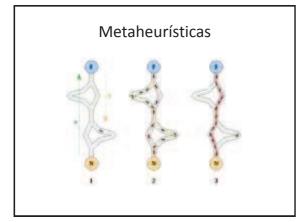
- Scheduling
- Asignación de registros en un programa compilado.
 - Teniendo los registros simbólicos y sabiendo cuales interfieren entre si se trata de mapear a k registros físicos.
- Asignación de frecuencias
- Y no menos importante: Sudoku!
 - 81 vértices, buscar un 9-coloreo











Algunas metaheurísticas

- Simulated annealing (primeros trabajos 1953, 1983)
 Tabú Search (primeras aplicaciones a optimización combinatoria en 1986, basado en algunas ideas de los 70)
 Algoritmos genéticos y evolutivos (primeras ideas en los 60, mayormente aplicaciones a problemas de IA)
 BRKGA (biased random keys genetic algorithms)
 GRASP (1989)

- Colonia de hormigas (1992)
- VNS (Variable Neighborhood Search) Iterated Local Search
- Scatter Search and Path Relinking

- Honey-bee mating optimization
 Particle Swarm Optimization
 Redes neuronales (primeras ideas en los 60, resurgieron en los 80)
- Híbridos

El problema de armar la bicicleta

- ¿Cómo mejorar el tiempo de fabricación de una bicicleta?
- · Las ventas andan bien
- Se forman equipos de dos personas para armarlas a mano, porque el equipo de ensamblado no llega a satisfacer la demanda
- Contratan a la empresa de Investigación Operativa Modelos I
- Pregunta: ¿Cuál es el menor tiempo necesario para que un equipo de 2 ensambladores arme una bicicleta

El problema de armar la bicicleta

El ensamblado de una bicicleta se divide en tareas:

- A) Preparación del marco (7 minutos)
- B) Montar la rueda delantera (7 minutos)
- C) Montar la rueda trasera (7 minutos)
- D) Montar el sistema de frenos (2 minutos) E) Montar los piñones de cambios (3 minutos)
- F) Montar la cadena al piñón (2 minutos)
- G) Montar el piñón a la bicicleta (2 minutos)
- H) Montar el pedal derecho (8 minutos)
- I) Montar el pedal izquierdo (8 minutos)
- J) Ajustes finales (18 minutos)

El problema de armar la bicicleta

¡No se puede armar la bicicleta en cualquier orden!

Restricciones de orden:

- A precede a B
- B, C, D, E y G preceden a J
- D precede a E y a F
- Fy G preceden a He I
- F precede a G
- D, E y A preceden a C

El problema de armar la bicicleta

Restricciones de trabajo:

- Ningún ensamblador puede estar sin hacer nada si hay alguna tarea que se puede comenzar
- Todo ensamblador termina la tarea que empieza antes de pasar a otra tarea.

El problema de armar la bicicleta

Este problema lo podríamos resolver por Programación Lineal Entera (parecido al 3.34), con variables que nos digan:

- En qué minuto comienza cada tarea
- Qué operario hace la tarea (bivalentes, una por cada operario para la misma tarea)

Y restricciones que controlen:

- Que antes de comenzar una tarea estén finalizadas las tareas precedentes
- Que no se hagan dos tareas al mismo tiempo
- Que al menos uno de los dos operarios haga la tarea



Algoritmo de resolución

Reglas heurísticas (que se agregan a las rest. de trabajo):

- El criterio para elegir un operario para realizar una tarea es el siguiente: si los dos operarios están desocupados, se le asigna tarea al operario 1 antes que al 2.
- El criterio para elegir una tarea a hacer (si tiene las precedencias cumplidas) es el siguiente: Si hay más de una tarea que se puede hacer, se hará primero la de mayor duración. Si hay más de una que tenga la mayor duración se elige entre ellas por orden alfabético.

Nota: Como criterio de selección se eligió la de mayor duración pero se podría elegir otro criterio ¿cuál podría ser?

Algoritmo de resolución

- 1) Se asigna al operario 1 una tarea que no tenga precedencias Si hay más de una se elige la de <u>mayor</u> duración
 - Si hay más de una se elige entre ellas por <u>orden alfabético</u>
- 2) Se asigna al operario 2 una tarea que no tenga precedencias Si no hay ninguna, espera a que termine el operario 1 Si hay más de una se elige la de mayor duración
 - Si hay más de una se elige entre ellas por orden alfabético
- 3) Mientras queden tareas sin asignar
 - Cuando termina de trabajar un operario se le asigna una tarea que tenga las precedencias cumplidas.
 - Si no hay ninguna, espera a que termine el otro operario Si hay más de una se elige la de mayor duración; si hay más de una se elige entre ellas por orden alfabético

Fin mientras

4) Se calcula el tiempo total que se tardó. FIN

Resolución utilizando el algoritmo planteado

- Ahora hagamos un diagrama de Gantt para resolver el problema con el algoritmo planteado.
- Vamos a plantear otro algoritmo de resolución: Reemplazamos en el que planteamos antes la palabra "mayor" por la palabra "menor". Ahora tenemos un algoritmo diferente. Resolvamos el problema con ese algoritmo para ver si tardamos menos tiempo que antes.
- ¡¡¡¡El resultado depende de los datos planteados!!!!

Algoritmo de resolución

- 1) Se asigna al operario 1 una tarea que no tenga precedencias Si hay más de una se elige la de menor duración
- Si hay más de una se elige entre ellas por orden alfabético
- 2) Se asigna al operario 2 una tarea que no tenga precedencias
 - Si no hay ninguna, espera a que termine el operario 1
 - Si hay más de una se elige la de menor duración
- Si hay más de una se elige entre ellas por orden alfabético
- 3) Mientras queden tareas sin asignar
- Cuando termina de trabajar un operario se le asigna una tarea que tenga las precedencias cumplidas.
- Si no hay ninguna, espera a que termine el otro operario Si hay más de una se elige la de <u>menor</u> duración; si hay más de una se elige entre ellas por <u>orden alfabético</u>

Fin mientras

4) Se calcula el tiempo total que se tardó. FIN

Resolución utilizando el algoritmo planteado...

- ¿Qué pasaría si el equipo que ensambla fuera de 3 personas?
- · Adaptemos el algoritmo planteado anteriormente para tres personas y resolvamos el problema (esta tarea se las dejamos a
- ¿Da mejor o peor? ¿Podría dar peor? SI, porque en las reglas de trabajo obligamos a que si alguien tiene trabajo para hacer debe hacerlo y a veces conviene esperar...