



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico III

Algoritmos y Estructura de Datos III
Primer Cuatrimestre de 2015

Integrante	LU	Correo electrónico
Iván Arcuschin	678/13	iarcuschin@gmail.com
Martín Jedwabny	885/13	martiniedva@gmail.com
José Massigoge	954/12	jmmassigoge@gmail.com
Lucas Puterman	830/13	lucasputerman@gmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

Índice

1. Ejercicio 1 - Demostración	3
1.1. Definiciones	3
1.2. Ejercicio A	3
1.3. Ejercicio B	5
1.4. Ejercicio C	5
2. Ejercicio 2 - Algoritmo exacto	6
3. Ejercicio 3 - Heurística constructiva golosa	7
4. Ejercicio 4 - Heurística de búsqueda local	8
5. Ejercicio 5 - MetaHeurística GRASP	9
6. Ejercicio 6 - Experimentación final	10

1. Ejercicio 1 - Demostración

1.1. Definiciones

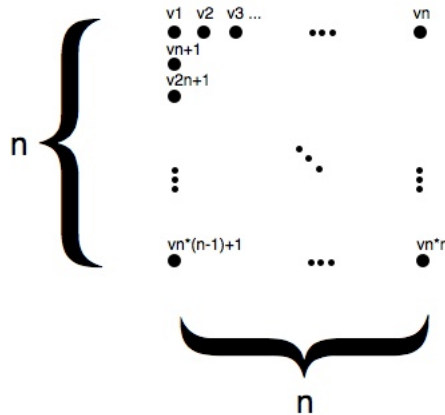
Sea $G=(V,E)$ un grafo simple:

- Definición 1: $D \subseteq V$ es un **conjunto dominante** (CD) $\iff \forall v \in V, v \in D \text{ ó } \exists w \in D \text{ tal que } (v,w) \in E$ (tiene un vecino en D).
- Definición 2: $D \subseteq V$ es un **conjunto independiente** (CI) $\iff \forall v,w \in D, (v,w) \notin E$.
- Definición 3: $D \subseteq V$ es un **conjunto independiente dominante** (CID) $\iff D$ es dominante e independiente.
- Definición 4: $D \subseteq V$ es un **conjunto independiente dominante mínimo** (CIDM) $\iff D$ es el conjunto independiente dominante de V con menos nodos. Es decir que $\forall D' \subseteq V$ tal que D' es independiente dominante, $\#(D) \leq \#(D')$.
- Definición 5: $D \subseteq V$ es un **conjunto independiente maximal** (CIMax) $\iff D$ es independiente y $\nexists D' \subseteq V$ independiente tal que $D \subset D'$ (\subset estricto).

1.2. Ejercicio A

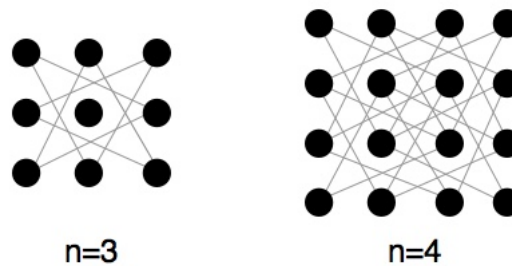
Relacionar el problema de CIDM con el problema 3 del TP 1 (i.e., similitudes y diferencias):

Si definimos a $G=(V,E)$ grafo simple para que represente al tablero de ajedrez que se forma en el ejercicio 3 del tp1, los nodos serían una cuadrícula de la pinta (sin considerar los ejes, eso lo hacemos después):



Donde 'n' es el parámetro del ejercicio que indica el tamaño de lado del tablero.

Ahora, como el ejercicio consiste en poner la mínima cantidad de caballos para que todas las casillas tengan un caballo o bien estén amenazadas, podemos representar las 'amenazas' como los ejes del grafo, por ejemplo:



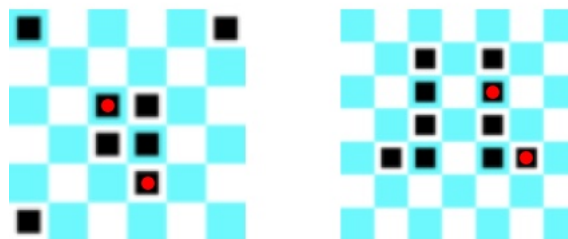
Observación: como cualquier posición x que amenaza a otra posición y sería amenazada si pongo un caballo en y , podemos representar el problema con un grafo simple y no un digrafo.

Entonces los ejercicios son similares ya que lo que estamos buscando en el ejercicio del TP1 es la mínima cantidad de caballos para que todas las posiciones (nodos) tengan un caballo o estén amenazadas (dominadas).

Es decir, si los caballos son un conjunto D de vértices en V , queremos hallar $\#(D)$ tal que:

- $\forall v \in V, v \in D \text{ o } \exists w \in D \text{ tal que } (v,w) \in E \iff D \text{ es dominante.}$
- $\forall D' \subseteq V \text{ tal que } D' \text{ es dominante, } \#(D) \leq \#(D') \iff D \text{ es mínimo entre los conjuntos dominantes de } V.$

Por lo tanto el problema consiste en hallar un conjunto dominante mínimo pero, a diferencia del problema en este TP, puede o no ser independiente. Esto se debe a que existen respuestas para el problema del TP1 donde hay caballos en ciertas posiciones que se amenazan entre sí:



Observación: estos graficos fueron extraidos de la página: <http://home.earthlink.net/~morgenstern/solution/knsols1.htm>. El primero representa una solución óptima del ejercicio del TP1 con un tablero de 6x6 y la segunda imagen es para un tablero de 7x7. En ambas, se marcan con un punto negro los casilleros que tienen un caballo y, como vemos señalado en rojo, hay caballos en posiciones que se amenazan entre sí.

1.3. Ejercicio B

Demostrar que todo conjunto independiente maximal es un conjunto dominante:

Sea $G=(V,E)$ un grafo simple y $D \subseteq V$ un conjunto independiente maximal, quiero ver que D es un conjunto dominante.

Supongamos (por absurdo) que D no es un conjunto dominante:

$\Rightarrow \exists v \in V$ tal que $v \notin D \wedge \forall w \in D, (v,w) \notin E$ (no es vecino de ningún nodo en D).

$\Rightarrow D \cup \{v\}$ es independiente ya que como D es independiente sucede que $\forall w \in D \cup \{v\} \nexists x \in D$ tal que $(x,w) \in E$.

\Rightarrow Absurdo! pues D era maximal y $\exists D' = D \cup \{v\}$ tal que D' es independiente y $D \subset D'$, así contradiciendo el hecho de que D es un conjunto independiente maximal.

Este absurdo vino de suponer que D no era dominante.

Por lo tanto, si D es un conjunto independiente maximal $\Rightarrow D$ es dominante.

1.4. Ejercicio C

Describir situaciones de la vida real que puedan modelarse utilizando CIDM:

- **El turista:** tenemos un conjunto de ciudades que forman un grafo simple conexo donde los nodos son las ciudades y dos nodos están conectados si y solo si las ciudades son vecinas. Suponiendo que queremos conocer tantas culturas distintas como sea posible, decidimos que no queremos visitar ningún par de ciudades vecinas ya que sus culturas son muy similares. Como el problema que proponemos consiste en hallar un conjunto mínimo de ciudades (nodos en el mapa) tal que toda ciudad sea visitada o bien una ciudad vecina sea visitada pero no ambas (el conjunto de nodos sea independiente y dominante), podemos decir que estamos buscando un CIDM.
- **Spamear una red social:** supongamos que tenemos un virus que se ocupa de spamear Facebook de manera que un mensaje sea propagado por toda la red. Asumimos que los usuarios de Facebook son nodos en un grafo simple conexo y que dos nodos están conectados si y solo si esos dos usuarios son amigos en la red. Ahora, el virus no quiere ser descubierto, así que lo que debe hacer es infectar la menor cantidad de cuentas que no sean amigas, ya que si spamea demasiadas cuentas o un par de cuentas que son amigas, sería mucho más fácil de descubrirlo. Luego, estamos buscando el mínimo conjunto de nodos en la red de usuarios tal que todos los usuarios vean el mensaje (dominación) y dos amigos no sean infectados a la vez (independencia), el cual es un CIDM.
- **Cámaras de seguridad en un barrio:** tenemos casas en un barrio que representan los nodos de un grafo simple conexo, donde dos nodos están conectados si y solo si sus casas respectivas son vecinas. Queremos poner cámaras de seguridad en el barrio de manera que toda casa tenga una cámara de seguridad o la tenga una casa vecina, ya que el rango de cobertura de la cámara es amplio y puede filmar la casa donde está y también las vecinas (dominación). No queremos poner cámaras en dos casas vecinas (independencia) para que no sea tan notorio y arruine el paisaje. A la vez, tenemos un presupuesto limitado, por lo cual queremos poner tan pocas cámaras como sea posible (minimalidad). Por lo tanto, el conjunto de casas que estamos buscando para ponerles cámaras es un CIDM.

2. Ejercicio 2 - Algoritmo exacto

3. Ejercicio 3 - Heurística constructiva golosa

4. Ejercicio 4 - Heurística de búsqueda local

5. Ejercicio 5 - MetaHeurística GRASP

6. Ejercicio 6 - Experimentación final