④ Semi definite programming applied to Max-cut

↳ La programación semidefinida consta de minimizar o maximizar una función lineal obteniendo una combinación de matrices simétricas que son positivas semidefinidas es decir que para una matriz A tal que $Z_1^T A Z_1 \geq 0$ para todo $Z_1 \in \mathbb{R}^n$ y $A \in \mathbb{R}^{n \times n}$.

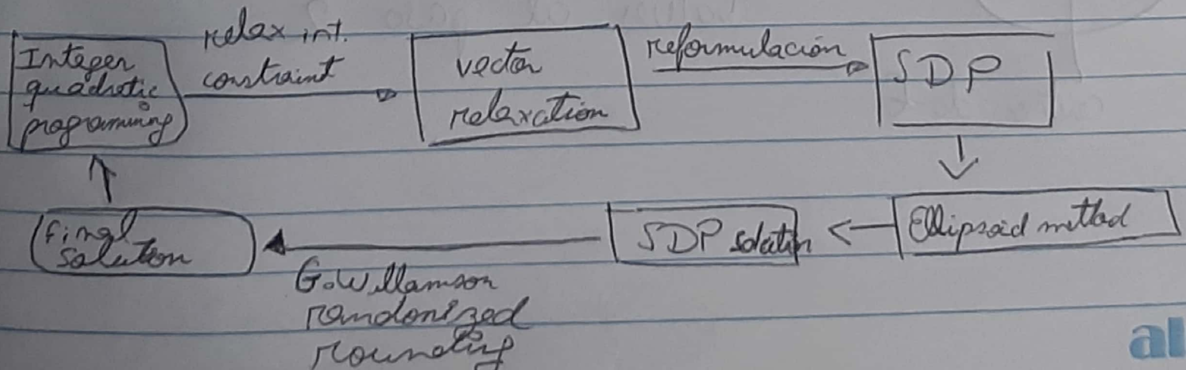
↳ El corte de un grafo consiste una bipartición de los vértices tal que $X \subseteq V(G)$ y el corte es $(X, V(G) \setminus X)$ y para $(u,v) \in E$ es corte si u y v están en diferentes partes de la bipartición. El tamaño de un graph-cut es la cantidad de edges con ese comportamiento (edge cut)

↳ Entonces el problema consiste en encontrar el corte de G que sea máximo, es decir, con mayor cantidad de edges cut. El problema es NP-Completo porque puede ser reducido a 3-SAT y por ende a SAT que ya se han probado como NP-completo

↳ Aplicaciones: Cualquier problema de optimización al que se le puede dar la forma de Semi-definite programming

El método de aproximación de elipsoide usado para resolver este problema se usa en cálculos de tamaño para transistores y cables; evaluación del potencial metabólico.

Approach general:



→ El análisis va para ellipsoid method que tiene costo de: para la solución del problema convexo $\min\{\theta \mid A_i * M \leq b_i\}$

$$O((m+n^2)\cdot n^5 \lg(n.R))$$

m = # of equations

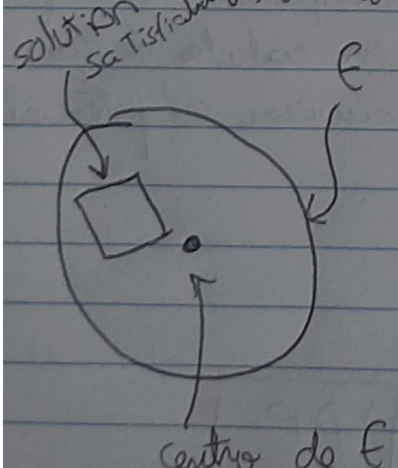
R = numerical size de coeficientes

Existen $O(n^2(m+n))$ operaciones por cada iteración y las iteraciones requeridas en ellipsoid method son $O(n^4 \lg(nR/\epsilon))$

1 • La elipsoide tiene todas las posibles soluciones, se coloca la solución al problema dentro de la elipsoide (E)

2 • Verifican si el centro de E está ~~dento~~ de la solución, en caso sí se satisface el problema y la solución fue encontrada. End.

↳ caso contrario, se usa un oracle de separación para generar un nuevo elipsoide que contenga la mitad de las soluciones de la elipsoide anterior.



↳ Si la nueva elipsoide es tan pequeña que ya no puede dar solución se ~~regresa~~ termina el algoritmo y no existen solución, de lo contrario volver al paso 2.

Con Goemans Williamson se obtiene a través de un análisis probabilístico de

$$Pr[(i,j) \in E(G) \text{ y sea edge cut}] = \frac{\theta}{\pi}$$

y SDPopt

$$E[\text{cut value}] / SDPopt \geq \min_{0 \leq \theta \leq \pi} \left\{ \frac{2}{\pi} \frac{\theta}{1-\cos\theta} \right\}$$

$$= 0.87\ldots$$

osea que SDOpt no es mayor que $1 - 0.87 = 13\%$ comparado con el problema NP