

# Heurísticas para el problema de ordenamiento lineal con costos acumulados

Irma García

Oswaldo Lucio

Arturo Domínguez

**Universidad Autónoma de Coahuila**

Abril 2017

# Introducción



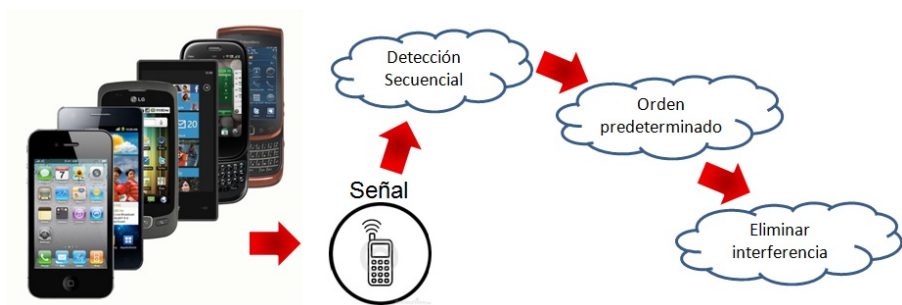
INEGI: *El crecimiento desbordado de la telefonía celular ha permitido que el 83 % de los hogares cuenten con servicio telefónico.*

Los teléfonos móviles dejaron de transmitir solamente voz y ahora se envían cada vez más datos e imágenes

# Introducción

- ▶ Los dispositivos móviles se comunican simultáneamente a centrales telefónicas, denominadas Estaciones Base.
- ▶ Cada dispositivo se identifica con un código específico, según la Universal Mobile Telecommunication Standard (UMTS): utiliza una técnica de acceso múltiple por división de código (CDMA).
- ▶ El medio de comunicación entre la central telefónica y los dispositivos móviles es el aire, donde se tiene interferencia entre dispositivos y se requiere mantener la interferencia multiusuario debajo de un nivel aceptable.

La cancelación sucesiva de interferencia es una técnica para reducir la interferencia



# Introducción

Problema 1: Orden de detección de los dispositivos.

# Introducción

Problema 2: Elección del nivel de potencia  $\alpha_i$  con el que el usuario  $i$  debe transmitir los datos.

- ▶ Un nivel de potencia alto permite una señal de potencia mejorada.
- ▶ Una minimización de la potencia de transmisión produce una mayor duración de las baterías de los dispositivos móviles.
- ▶ "La duración de las baterías es una limitante muy importante en los sistemas de comunicación móvil"
- ▶ Las limitaciones físicas y de reglamentación imponen un límite superior en la potencia de transmisión de los dispositivos móviles.

# Introducción

- ▶ El orden para el SIC y los niveles de potencia deben asegurar la confiable detección de las señales de los dispositivos móviles.
- ▶ Esto se logra cuando el porcentaje de la señal a ruido mas la interferencia en la potencia (SNIR) es igual a un nivel deseable  $\Gamma$ .
- ▶ Para SIC, la SNIR se relaciona con la potencia de interferencia generada desde el usuario  $i$  en el usuario  $j$ , denotado por  $\rho_{ij}$ .

# Introducción

Tras la detección del usuario  $k_p$  la SNIR está dada por

$$SNIR^{(p)} = \frac{\alpha_{k_p} \rho_{k_p k_p}}{N_0 \sqrt{\rho_{k_p k_p}} + \sum_{i \in \mathcal{U}_p} \alpha_i N_S \rho_{i k_p}}$$

$N_0$  potencia de ruido

$N_S$  factor de interferencia

$\mathcal{U}_p = \{k_{p+1}, k_{p+2}, \dots, k_n\}$  conjunto de usuarios no detectados en la etapa  $p$ .



# JOPCO: Joint Optimization of Power Control and Ordering

- ▶ Se requiere optimizar conjuntamente el orden de detección SIC y los niveles de potencia de transmisión, minimizando la potencia de transmisión global para asegurar una recepción adecuada de todos los usuarios de dispositivos móviles.
- ▶ Optimización conjunta de control de potencia y ordenamiento (JOPCO)
- ▶ JOPCO se formula como LOP-CC (Problema de ordenamiento lineal con costos acumulados)

# LOP: Linear Ordering Problem

Problema de optimización NP-hard, que se modela con un grafo

$$G = (V, A)$$

$V = \{1, 2, \dots, n\}$  nodos o vértices

$A$  conjunto de arcos o aristas que unen los vértices.

Matriz de costos entre nodo  $i$  y nodo  $j$ .

Se busca una permutación de los nodos que optimice una función objetivo.

# LOP: Linear Ordering Problem

Dada una matriz de pesos  $C = (c_{ij})_{n \times n}$  LOP consiste en maximizar

$$f(p) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n c_{p_i p_j}$$

$p$  es una permutación de los nodos

Ejemplo

$$C = \begin{pmatrix} 0 & 2 & 1 & 3 \\ 4 & 0 & 1 & 5 \\ 2 & 3 & 0 & 5 \\ 1 & 2 & 1 & 0 \end{pmatrix}$$

Se busca una permutación de renglones y columnas que maximice la suma de los elementos de la triangular arriba de la diagonal principal

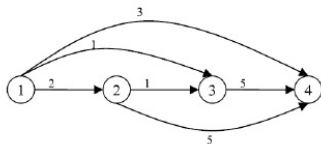
# LOP: Linear Ordering Problem

Si  $p = (1, 2, 3, 4)$  entonces

$$f(p) = c_{12} + c_{13} + c_{14} + c_{23} + c_{24} + c_{34} = 17$$

Si  $p = (2, 3, 1, 4)$  entonces

$$f(p) = c_{23} + c_{21} + c_{24} + c_{31} + c_{34} + c_{14} = 20$$



LOP tiene aplicaciones en economía con matrices input-output.

# LOP-CC: Linear Ordering Problem with cumulative costs

Se tiene un grafo completo con pesos en los nodos  $d_i$  y matriz de costos  $C$ , se busca una permutación de los nodos  $p = (p_1, p_2, \dots, p_n)$  y los valores de  $\alpha_j$  que minimizan

$$C_{LOPCC}(p) = \sum_{i=1}^n \alpha_i$$

donde

$$\alpha_{p_i} = d_{p_i} + \sum_{j=i+1}^n c_{p_i p_j} \alpha_{p_j}, \quad i = n, n-1, \dots, 1$$

El cálculo acumulado de las  $\alpha$  se hace en reversa.

# LOP-CC: Linear Ordering Problem with cumulative costs

Ejemplo:

$$C = \begin{pmatrix} 0 & 2 & 1 & 3 \\ 4 & 0 & 1 & 5 \\ 2 & 3 & 0 & 5 \\ 1 & 2 & 1 & 0 \end{pmatrix}$$

Con pesos en los nodos  $d = (3, 2, 4, 1)$

Si consideramos la permutación  $p = (1, 2, 3, 4)$ , tendremos

$$C_{LOPCC}(p) = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4$$

# LOP-CC: Linear Ordering Problem with cumulative costs

$$C_{LOPCC}(p) = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4$$

$$\alpha_4 = d_4 = 1$$

$$\alpha_3 = d_3 + c_{34}\alpha_4 = 9$$

$$\alpha_2 = d_2 + c_{23}\alpha_3 + c_{24}\alpha_4 = 16$$

$$\alpha_1 = d_1 + c_{12}\alpha_2 + c_{13}\alpha_3 + c_{14}\alpha_4 = 47$$

$$C_{LOPCC}(p) = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 73$$

## JOPCO y LOP-CC. (Benvenuto and Tomasin, 2005)

Para una permutación dada, se pueden calcular los niveles de potencia  $\alpha_i$  como

$$\alpha_{k_i} = \frac{\Gamma N_0 \sqrt{\rho_{k_i k_i}} + \Gamma N_S \sum_{l \in \mathcal{U}_i} \alpha_l \rho_{lk_i}}{\rho_{k_i k_i}}$$

Los valores de  $\alpha_{k_i}$  pueden calcularse en orden inverso para  $i = n, n-1, \dots, 1$ .

Se definen los pesos en los nodos como

$$d_i = \Gamma N_0 / \sqrt{\rho_{ii}}$$

Los costos

$$c_{ij} = \Gamma N_S \rho_{ji} / \rho_{ii}$$

Con esto JOPCO se puede ver como un LOP-CC.



# GRASP: Feo and Resende 1994

GRASP: Greedy Randomized Adaptive Search Procedure

Metaheurística que consta de dos pasos:

- ▶ Construcción de solución
- ▶ Búsqueda local

GRASP se basa en la premisa que soluciones iniciales buenas y diversas juegan un papel importante en el éxito de los métodos de búsqueda local.

Método multiarranque en donde la construcción de la solución introduce un factor de aleatoriedad en una forma específica.

La fase de búsqueda local explora las soluciones vecinas tratando de encontrar otra solución alguna que mejora el valor de la función objetivo.

# Multiarranque

- ▶ Inicialización ( $i = 0$ )
  - ▶ Obtener una solución con el algoritmo constructivo
  - ▶ Sea  $c^* = \text{costo}$
- ▶ Mientras ( $i < \text{MAXITER}$ )
  - ▶ Obtener una solución  $x(i)$  con el algoritmo constructivo aleatorizado
  - ▶ Aplicar búsqueda local a  $x(i)$ 
    - ▶ Sea  $x^*(i)$  la solución obtenida y  $S^*(i)$  su valor
  - ▶ Si ( $S^*(i)$ ) mejora a  $c^*$ , hacer  $c^* = S^*(i)$  y almacenar  $x^*(i)$
  - ▶  $i = i + 1$

# Componentes de GRASP

- ▶ Función de evaluación miope
  - ▶ Guía la construcción de evaluaciones
  - ▶ Efectuar el mejor movimientos disponible en cada paso
- ▶ Procedimiento de elección al azar
- ▶ Proceso de actualización adaptativo
- ▶ Búsqueda local

# Lista Restringida de Candidatos

- ▶ En cada etapa construir una lista de candidatos con movimientos admmissibles
  - ▶ Ordenados de manera decreciente con respecto a su beneficio medido por la funci3n miope
- ▶ Restringir la lista de forma que contenga buenos movimientos aunque no necesariamente 3ptimo
- ▶ Se puede restringir por cardinalidad
- ▶ Se puede restringir por valor

# Lista Restringida de Candidatos por valor

- ▶ Parámetro  $\beta$  que define la calidad de los elementos en la RCL
- ▶  $c^{min} \leq c(e) \leq c^{min} + \beta(c^{max} - c^{min})$
- ▶  $\beta = 0$ : construcción puramente miope
- ▶  $\beta = 1$ : construcción puramente aleatoria

# Adaptativo

- ▶ Una vez elegido el movimiento se procede a recalcular los beneficios asociados a cada elemento
- ▶ Debe reflejar los cambios producidos al incorporar la selección previa como una componente de la solución
- ▶ Actualizar la información para el paso constructivo siguiente

# Constructivo Cásico

1.  $U \leftarrow \{1, \dots, n\}$  Set of unselected vertices
2.  $sol \leftarrow \emptyset$  Solution under construction
3. Select  $i$  randomly form  $U$
4.  $k \leftarrow n$
5. Set  $i$  in the position  $k$  of  $sol$
6.  $U \leftarrow U - \{i\}$
7. while  $U \neq \emptyset$  do
  8.  $k \leftarrow (k - 1)$
  9.  $\forall j \in U$  Compute  $\Delta C(j)$  in position  $k$
  10.  $RCL \leftarrow \{j \in U / \Delta C(j) \leq C^{min} + \beta(C^{max} - C^{min})\}$
  11. Select  $i$  randomly from RCL
  12. Set  $i$  in the position  $k$  of  $sol$
  13.  $U = U - \{i\}$
14. end while

# Constructivo Modificado (Sandoya, Martí, 2014)

1.  $U \leftarrow \{1, \dots, n\}$  Set of unselected vertices
2.  $sol \leftarrow \emptyset$  Solution under construction
3. Select  $i$  randomly form  $U$
4.  $k \leftarrow n$
5. Set  $i$  in the position  $k$  of  $sol$
6.  $U \leftarrow U - \{i\}$
7. Aleatorizar el orden de los nodos restantes
8. while  $U \neq \emptyset$  do
  9.  $k \leftarrow (k - 1)$
  10.  $U' =$  porcentaje  $p$  de elementos de  $U$
  11.  $\forall j \in U'$  Compute  $\Delta C(j)$  in position  $k$
  12.  $c^{min} \leftarrow \min\{\Delta C(j)/j \in U'\}$
  13. Set  $i$  in the position  $k$  of  $sol$
  14.  $U = U - \{i\}$
15. end while



# Búsqueda Local (Duarte et.al. 2012)

---

```
begin LocalSearch(sol)
1.   Let  $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$  be the contribution of
     each vertex to the objective function
2.   imp  $\leftarrow$  true
3.   while imp do
4.       imp  $\leftarrow$  false
5.       for j = 1 to n do
6.           Sort  $\alpha$  in ascending order
7.           Let i be the element with  $\alpha[j]$  contrib
8.           Let pos_i the position of i in sol
9.           for k = pos_i - 1 to 1 do
10.              sol  $\leftarrow$  swap(sol,k)
11.              sBest  $\leftarrow$  UpdateBestSol(sol, imp)
12.           end for
13.       end for
14.   end while
15.   return sBest
end
```

---

# Experimentos computacionales

Se probaron los dos constructivos en una matriz de  $4 \times 4$  donde se conoce el valor exacto y los dos procedimientos por separado y con búsqueda local llegan al óptimo.

Se trabajó en un grupo de instancias del Depto. de Telecomunicaciones de la Universidad de Padova, Italia, que están relacionadas con el orden de detección de las redes de la UMTS (Benvenuto and Tomasin, 2004).

Se tienen 4 escenarios: transmisión síncrona y asíncrona, con y sin codificación.

# Experimentos computacionales

Para cada escenario se tienen 500 matrices  $(\rho_{ij})$  que se han generado de forma aleatoria con 16 usuarios distribuidos uniformemente en la célula de radio 580 m.

## Parámetros

Fuente de radio $\Gamma$	0.625
Factor de interferencia $N_S$	16
Potencia de ruido $N_0$	0.5047685

[http://www.math.unipd.it/~bertacco/LOPCC\\_instances.zip](http://www.math.unipd.it/~bertacco/LOPCC_instances.zip).

Se trabajó con las primeras 25 instancias de cada escenario y se ejecutaron 5000 iteraciones.

# Experimentos computacionales

Extracto primeros renglones y columnas de matriz  $\rho$

27,889190000	0,004651135	0,005851801	0,607521700
0,004651135	14,676770000	0,073464520	0,107290200
0,005851801	0,073464520	10,34819000	0,354813700
0,607521700	0,107290200	0,354813700	4,450402000

*Peso1* 0,0597381891742513

*Peso2* 0,0823483231516338

*Peso3* 0,0980703685304489

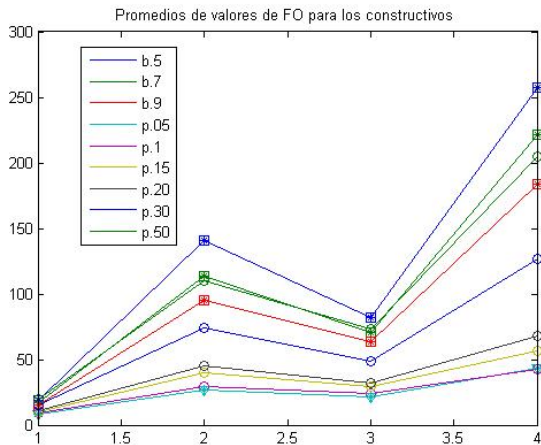
*Peso4* 0,1495444787397150

# Resultados de experimentos computacionales: valor de la fo

Promedios Función Objetivo

Parámetro \ Escenario	00	01	10	11
$\beta = 0.5$	18.75795	140.72577	81.80310	257.83975
$\beta = 0.7$	16.94055	113.27997	70.57068	221.85976
$\beta = 0.9$	15.03016	95.56282	63.56345	183.37760
<i>porcentaje = 0.05</i>	8.13060	26.68815	21.78178	43.68846
<i>porcentaje = 0.10</i>	9.15806	29.40670	23.87663	42.84730
<i>porcentaje = 0.15</i>	10.39953	39.77732	29.42065	56.55054
<i>porcentaje = 0.20</i>	11.13986	45.44507	31.79521	68.33092
<i>porcentaje = 0.30</i>	14.93148	74.07650	48.74075	127.16619
<i>porcentaje = 0.50</i>	19.69017	109.70106	72.82172	205.07911
<i>porcentaje = 0.70</i>	27.70621	191.99866	130.99842	306.45095
<i>porcentaje = 0.90</i>	38.28226	245.16550	241.42540	337.38565

# Resultados de experimentos computacionales: valor de la fo

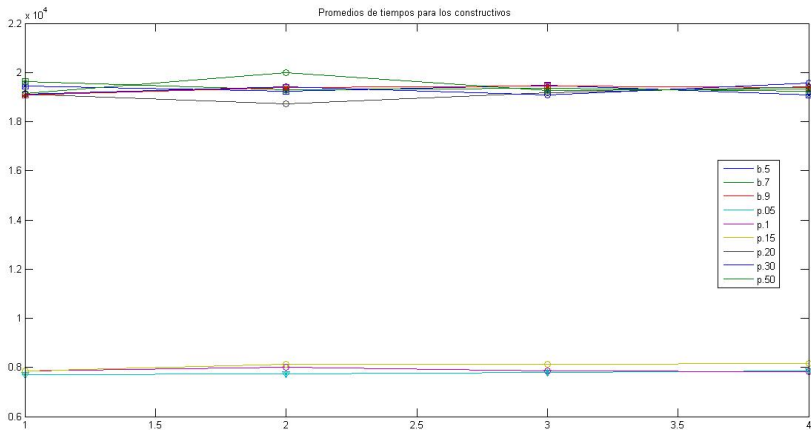


# Resultados de experimentos computacionales: tiempos de ejecución

Promedios de Tiempo en Segundos

Escenario \ Parámetro	00	01	10	11
$\beta = 0.5$	19457	19224	19494	19073
$\beta = 0.7$	19644	19286	19362	19235
$\beta = 0.9$	19098	19389	19453	19385
<i>porcentaje = 0.05</i>	7703	7736	7788	7866
<i>porcentaje = 0.10</i>	7847	8010	7869	7822
<i>porcentaje = 0.15</i>	7862	8115	8131	8153
<i>porcentaje = 0.20</i>	19112	18731	19173	19431
<i>porcentaje = 0.30</i>	19125	19414	19073	19576
<i>porcentaje = 0.50</i>	19146	20003	19278	19343
<i>porcentaje = 0.70</i>	19519	19264	19265	19439
<i>porcentaje = 0.90</i>	19359	19285	19332	19330

# Resultados de experimentos computacionales: tiempos de ejecución



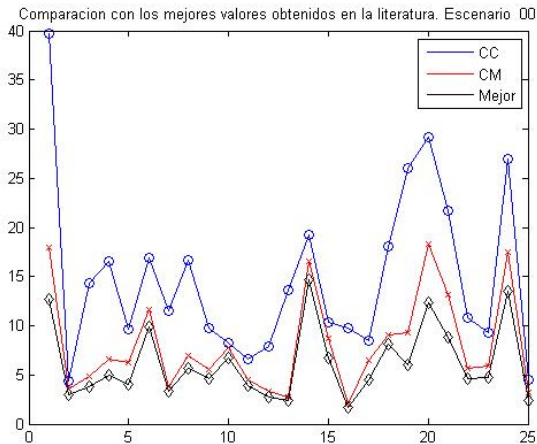


# Resultados de experimentos computacionales de los métodos constructivos comparados con los mejores valores reportados en la literatura

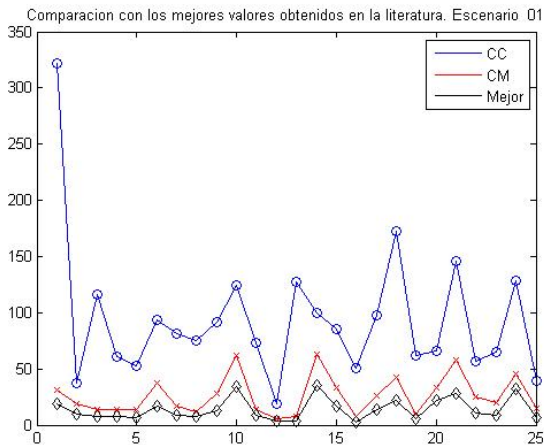
Mejores resultados obtenidos (Escenario 00 - Matriz 1-25)

Instancia	CC	CM	Mejor FO
mrho1.00	39.74200	17.94440	12.67500
mrho2.00	4.40994	3.59932	2.95300
mrho3.00	14.34160	4.81715	3.77400
mrho4.00	16.51790	6.55932	5.00200
mrho5.00	9.61255	6.29709	4.02100
mrho6.00	16.85010	11.65100	9.85600
mrho7.00	11.50970	3.77994	3.37200
mrho8.00	16.72450	6.98196	5.63300
mrho9.00	9.71630	5.60694	4.62700
mrho10.00	8.24814	7.72014	6.74100
mrho11.00	6.56136	4.48478	3.96300
mrho12.00	7.95067	3.37789	2.75100

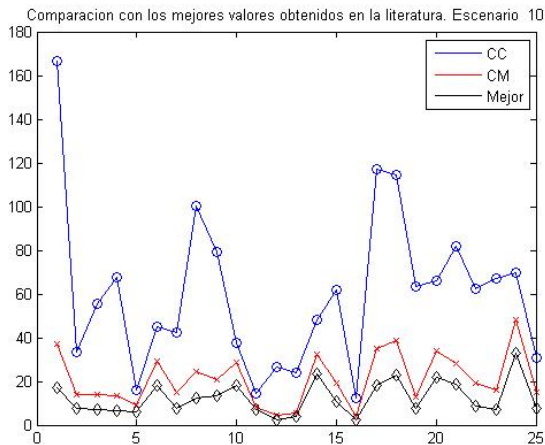
# Comparación de constructivos con los mejores valores reportados en la literatura.



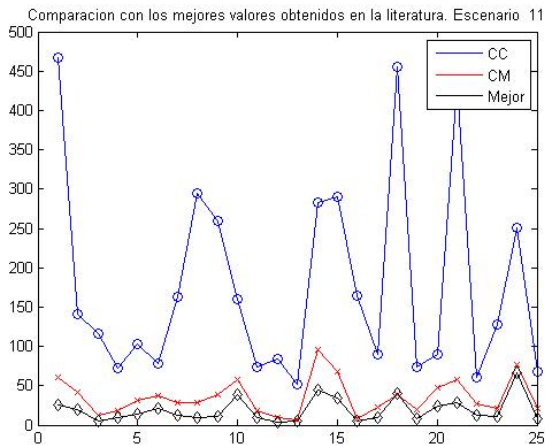
# Comparación de constructivos con los mejores valores reportados en la literatura.



# Comparación con los mejores valores reportados en la literatura.

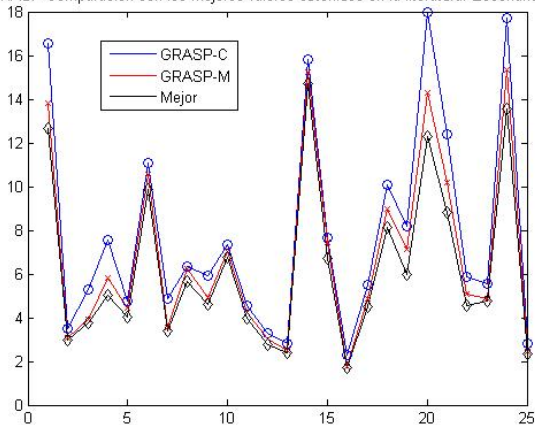


# Comparación de constructivos con los mejores valores reportados en la literatura.



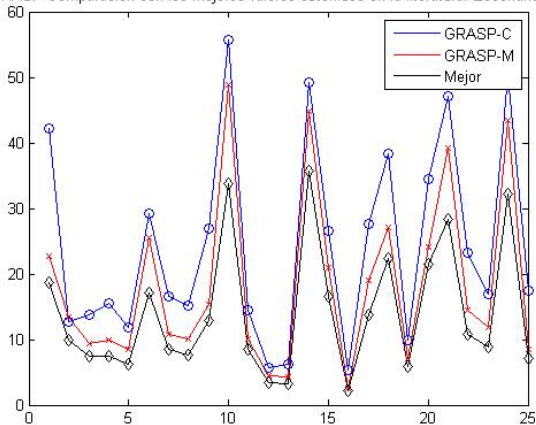
# Comparación GRASP con los mejores valores reportados en la literatura.

GRASP Comparacion con los mejores valores obtenidos en la literatura. Escenario 00



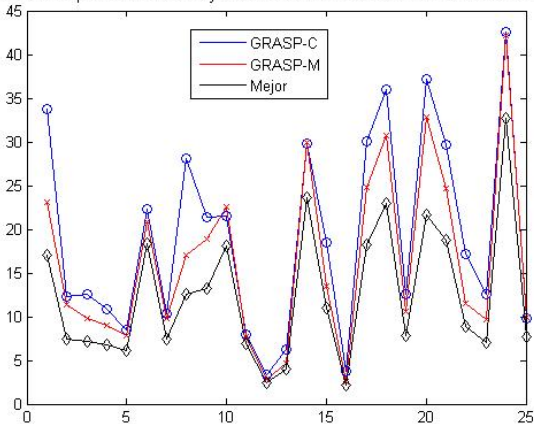
# Comparación GRASP con los mejores valores reportados en la literatura.

GRASP Comparacion con los mejores valores obtenidos en la literatura. Escenario 01



# Comparación GRASP con los mejores valores reportados en la literatura.

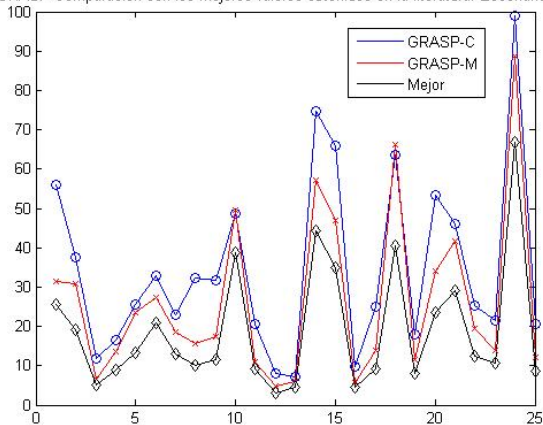
GRASP Comparacion con los mejores valores obtenidos en la literatura. Escenario 10





# Comparación GRASP con los mejores valores reportados en la literatura.

GRASP Comparacion con los mejores valores obtenidos en la literatura. Escenario 11



# Conclusiones

Con el procedimiento constructivo modificado es con el que se obtiene el mejor valor del GRASP en todos los escenarios.

Trabajo a futuro: Incorporar algún postprocesamiento para lograr llegar al mejor valor reportado en la literatura.

El GRASP modificado propuesto es una buena alternativa al cálculo de soluciones de calidad con bajo costo de tiempos de ejecución.

Alternativa de construcción de soluciones con tiempos de cómputo menores para otros problemas sobre todo con funciones objetivo caras al evaluar.

# Fin