Tarea 4-Optimización Estocástica

1st Daniel Vallejo Aldana
Maestría en Ciencias de la Computación
Centro de Investigación en Matemáticas
daniel.vallejo@cimat.mx

***Resumen*—En el presente trabajo se implementan y se evalúan diversas metaheurísticas de trayectoria aplicadas al problema de las p-medianas. Se implementa como punto de partida el algoritmo de Recocido Simulado (SA) para el problema de las p-medianas. Así mismo se proponen dos variantes del SA usando Variable-Neighborhood-Search (VNS) en donde en el algoritmo SAVNSI el tamaño de la vecindad depende de la mejora obtenida respecto al candidato anterior. En el algoritmo SAVNSL, la vecindad decae de forma líneal partiendo de un tamaño máximo de vecindad y decayendo conforme al tiempo transcurrido en el algoritmo.**

***Index Terms*—p-medianas, grafos, búsqueda local, programación dinámica, recocido simulado, variable-neighborhood-search.**

I. Introducción

De acuerdo a [2], el problema de las p-medianas consiste en localizar p-instalaciones (medianas) dentro de una red (grafo) con $|V|$ nodos y $|E|$ aristas de tal forma que minimicemos la suma de todas las distancias de cada vértice v del grafo $G(V, E)$ a su instalación más cercana. Varios métodos se han utilizado para la resolución de este problema, en [2] se propone un método basado en relajación Lagrangiana, en [4] se analiza resolver este problema con algoritmos de métodos de partición como lo son la Variable Neighbor Search (VNS) o la búsqueda tabú. Así mismo el trabajo de Loranca et.al. [4], es usado como comparación de los resultados obtenidos en el presente trabajo.

En el presente trabajo se implementa el algoritmo de recocido simulado [3] aplicado al problema de las p-medianas [1]. Así mismo se proponen dos modificaciones al algoritmo de SA las cuales aplican el algoritmo de recocido simulado a vecindades de tamaño variable de acuerdo al porcentaje de mejora respecto a la solución anterior (SAVNSI) o respecto a un decaimiento líneal en el tamaño de la vecindad (SAVNSL). Dichas propuestas obtuvieron buenos resultados, sin embargo, no lograron superar al algoritmo de recocido simulado descrito en [1].

II. Metodología

Se implementó el algoritmo del recocido simulado aplicado al problema de las p-medianas utilizando como base lo descrito en [1]. En el Algoritmo 1 se presenta el pseudocódigo del algoritmo modificado para esta tarea.

Así mismo se propuesieron dos modificaciones al algoritmo anterior considerando vecindades $N(s_0)$ de tamaño variable, la primera modificación considera el tamaño de la vecindad de la siguiente forma

$$NSize = 1 + \lfloor \gamma * MAXNSize \rfloor$$

Donde $\gamma = \frac{mejora}{costoAnterior}$

En el caso del SAVNSL la forma en la que decae el tamaño de la vecindad es de la forma.

Algorithm 1 Algoritmo de Recocido Simulado SA

Require: T, la temperatura inicial, L, iteraciones de búsqueda, s_0 la solución inicial, G el grafo, C contenedor de información, R es la tasa de decaimiento de la temperatura.
Ensure: s^* una solución al problema de las p-medianas

```
initCost = C.GetLoss()
K = MN
while time ≤ 1200 do
    for i in L do
        s' = genRandomNeighbor(N^K(s_0))
        Δ = s' − s_0
        if Δ < 0 then
            s_0 = s'
            C.update(s_0)
            initCost = C.GetLoss()
        else
            if X ∼ Unif(0,1) < exp −Δ/T then
                s_0 = s'
                C.update(s_0)
                initCost = C.GetLoss()
            end if
        end if
    end for
    time = time + elapsedSecs
    T = R − (R*t / L)
end while
return s' = s_0
```

$$NSize = 1 + MAXNSize * (Rate - RateDecay)$$

Cabe resaltar que en la segunda implementación el algoritmo SAVNSL pasa más tiempo en las vecindades grandes, sin embargo de acuerdo a los resultados obtenidos no hay evidencia de que esto sea benéfico al momento de encontrar buenas soluciones candidatas.

Las modificaciones al algoritmo de SA agregando algunas ideas de VNS se muestran a continuación en el Algoritmo 2.

III. Resultados

III-A. Instancia 1

La primera instancia evaluada consta de un conjunto de 100 nodos con 200 aristas y $p = 5$. Se corrieron los algoritmos descritos en la presente tarea además del algoritmo que utiliza programación dinámica para encontrar soluciones descrito en la tarea 3. Los cuatro algoritmos fueron ejecutados durante 20 minutos (1200 segundos) y comparamos las mejores soluciones obtenidas por los algoritmos.

Algorithm 2 Algoritmo de Recocido Simulado SA con VNS

Require: T, la temperatura inicial, L, iteraciones de búsqueda, s_0 la solución inicial, G el grafo, C contenedor de información, R es la tasa de decaimiento de la temperatura, MN máximo tamaño del vecindario.

Ensure: s^* una solución al problema de las p-medianas

$initCost = C.GetLoss()$
while $time \leq 1200$ **do**
  for i in L **do**
    $s^{'} = genRandomNeighbor(N(s_0))$
    $\Delta = s^{'} - s_0$
    if $\Delta < 0$ **then**
      $s_0 = s^{'}$
      $K_{SAVNSI} = 1 + MN * \frac{\Delta}{C.GetLoss()}$
      $C.update(s_0)$
      $initCost = C.GetLoss()$
    else
      if $X \sim Unif(0,1) < \exp -\Delta/T$ **then**
        $s_0 = s^{'}$
        $C.update(s_0)$
        $initCost = C.GetLoss()$
      end if
    end if
  end for
  $time = time + elapsedSecs$
  $K_{SAVNSL} = 1 + \lfloor (1 - \sum_{i=0}^{itertions} DecayRate) \rfloor$
  $T = R - \left(\frac{R*t}{L}\right)$
end while
return $s^{'} = s_0$

Para todos los experimentos se definieron los parámetros $T = 1{,}25$ y $L = 3000$ tomando como sugerencia lo descrito en [1].

En la figura 1 observamos los boxplot de las soluciones arrojadas por los algoritmos.

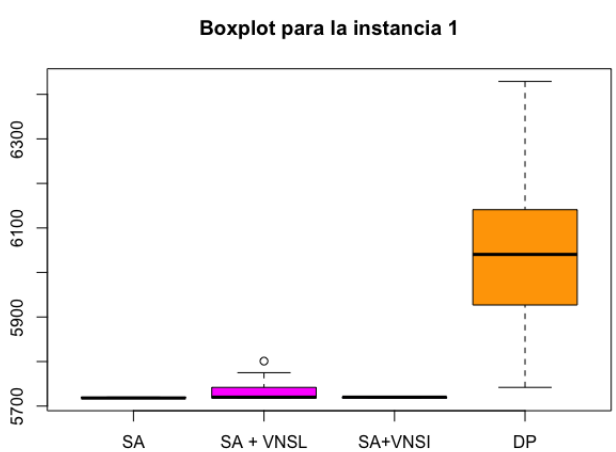

Figura 1. Boxplot de las soluciones obtenidas por los 4 algoritmos ejecutados, podemos ver que los algoritmos de recocido simulado y con decaimiento por mejora obtuvieron los mejores resultados.

Podemos observar que los algoritmos de recocido simulado logran superar por mucho al algoritmo que utilizaba programación dinámica. Cabe resaltar que dicho método obtuvo las mejores soluciones en las tareas anteriores por margen considerable de diferencia. Se resalta además que la inicialización de las soluciones es completamente aleatoria ya que esto dió mejores resultados en las tareas anteriores.

En el cuadro I podemos observar los resultados cuantitativos de las ejecuciones realizadas.

Método	**Media**	**Min**	**Max**
SA	**5719**	**5718**	**5720**
SA+VNSL	5729	**5718**	5801
SA + VNSI	**5719**	**5718**	**5720**
DP	6041	5742	6388
Mejor conocido		5891 [4]	

Cuadro I
TABLA COMPARATIVA DE RESULTADOS OBTENIDOS

III-B. Instancia 2

La segunda instancia evaluada consta de un grafo G con 200 nodos, 800 aristas y $p = 40$. Los boxplot de las soluciones obtenidas se muestran en la figura 2

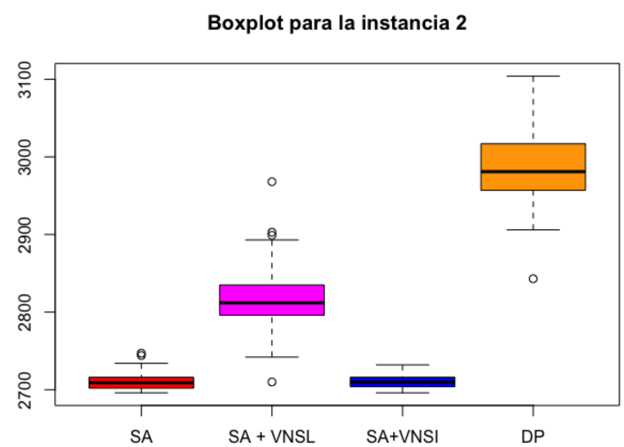

Figura 2. Boxplot de las solucione para la segunda instancia, en este caso vemos que los algoritmos de SAVNSI y SA obtienen los mejores resultados. Podemos notar que el estar más tiempo en vecindades grandes como en el caso del algoritmo SAVNSL decrece el rendimiento del algoritmo.

Para la instancia 2 se tiene la siguiente tabla comparativa presentada en el cuadro II

Método	**Media**	**Min**	**Max**
SA	2711	**2696**	2747
SA+VNSL	2818	2710	2968
SA + VNSI	**2710**	**2696**	**2732**
DP	2985	2843	3104
Mejor conocido		2837 [4]	

Cuadro II
TABLA COMPARATIVA DE RESULTADOS OBTENIDOS

Del cuadro II podemos ver que el algoritmo de SAVNSI redujo la varianza de las soluciones encontradas respecto al algoritmo SA.

III-C. Instancia 3

Debido a la complejidad computacional tan grande que implica resolver el problema de las p medianas para instancias grandes con programación dinámica, se consideró una instancia con 300 nodos, 1800 aristas y valor de $p = 100$. Los resultados de las ejecuciones se muestran a continuación en la figura 3

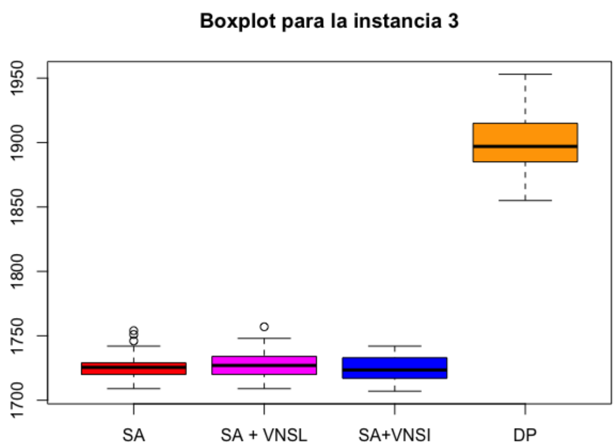

Figura 3. Boxplot de soluciones para la tercera instancia considerada, podemos ver que nuevamente los algoritmos de SA y SAVNSI obtienen los mejores resultados, e importante recalcar que el algoritmo de SAVNSI tiene menor número de outliers que el algoritmo de SA

Tenemos la siguiente tabla comparativa de resultados obtenida después de hacer las 50 corridas por 20 minutos.

Método	Media	Min	Max
SA	1726	1709	1754
SA+VNSL	1728	1709	1757
SA + VNSI	**1725**	**1707**	**1742**
DP	1901	1855	1953
Mejor conocido		1729 [4]	

Cuadro III
TABLA COMPARATIVA DE RESULTADOS OBTENIDOS

Del cuadro III podemos ver que nuevamente el algoritmo SAVNSI obtuvo los mejores resultados superando incluso al mejor valor conocido en ORLIB de acuerdo a lo reportado por [4]. Lo anterior nos indica que el tamaño de las vecindades si juega un papel importante para poder encontrar una mejor solución que solamente considerando una vecindad de tamaño fijo.

IV. CONCLUSIÓN

Con base en los resultados obtenidos en el presente trabajo se puede concluir que el uso del algoritmo de recocido simulado ofrece una mejora significativa en la obtención de buenas soluciones para el problema de las p-medianas. No obstante, podemos observar que el trabajar con el tamaño de las vecindades como se describe en el algoritmo de VNS ayuda a mejorar los resultados obtenidos por SA. como es el caso del algoritmo propuesto SAVNSI. no obstante queda como trabajo futuro el encontrar formas más eficientes de encontrar tamaños de vecindades que 1) Reduzcan aún más la variación en las soluciones y 2) Se adapten de forma automática a la mejora obtenida de las soluciones.

V. APÉNDICE A: COMPARATIVA DE MEJORA A LO LARGO DE LAS TAREAS

En los siguientes boxplots mostramos como se ha ido mejorando la estimación de las soluciones a lo largo de los algoritmos implementados desde la tarea 1 hasta la 4.

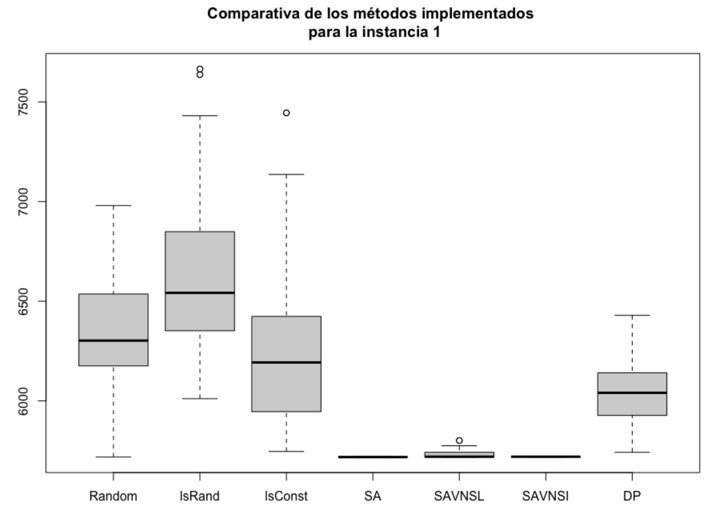

Figura 4. Comparativa de mejora para la instancia 1

V-A. *Instancia 1*

En la figura 4 mostramos las mejoras que se han realizado a lo largo de las tareas 1 a 4 sobre la instancia 1.

V-B. *Instancia 2*

Para el caso de la instancia 2 se tienen los siguientes boxplots mostrados en la figura 5.

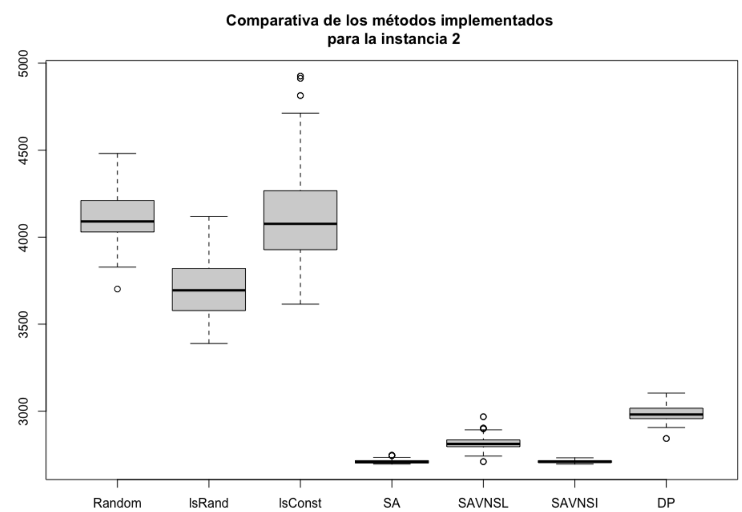

Figura 5. Comparativa de mejora para la instancia 2

REFERENCIAS

[1] Abdulrahman Al-khedhairi. Simulated annealing metaheuristic for solving p-median problem. *Int. J. Contemp. Math. Sciences*, 3(28):1357–1365, 2008.

[2] John E Beasley. A note on solving large p-median problems. *European Journal of Operational Research*, 21(2):270–273, 1985.

[3] Scott Kirkpatrick, C Daniel Gelatt Jr, and Mario P Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.

[4] María Beatríz Bernábe Loranca, Jorge A Ruiz-Vanoye, Rogelio González Velázquez, Marco Antonio Rodríguez Flores, and Martín Estrada Analco. P-median: A performance analysis. *Res. Comput. Sci.*, 88:19–30, 2014.