

Metaheurística GRASP

Introducción

Para desarrollar esta metaheurística se tomó grafos de distintos tamaños representado por una matriz de adyacencias como entrada a la función grasp.

Esta función grasp además recibe como parámetros una función que realizará el random greedy, la función que realizará la búsqueda local, la cantidad máxima de iteraciones y un porcentaje de mejora para la búsqueda local.

Cada iteración de grasp se quedará con la mejor solución (la de menor tamaño) en la comparativa entre lo que devuelve el greedy random y su mejora con la búsqueda local.

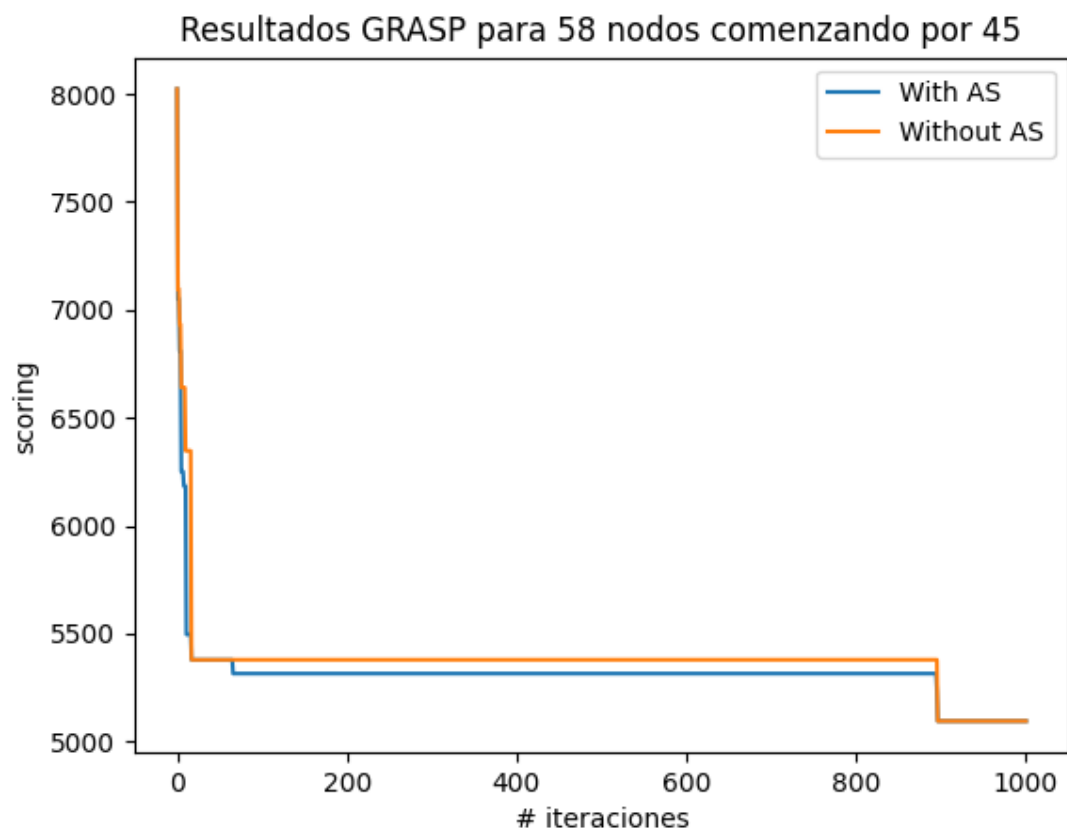
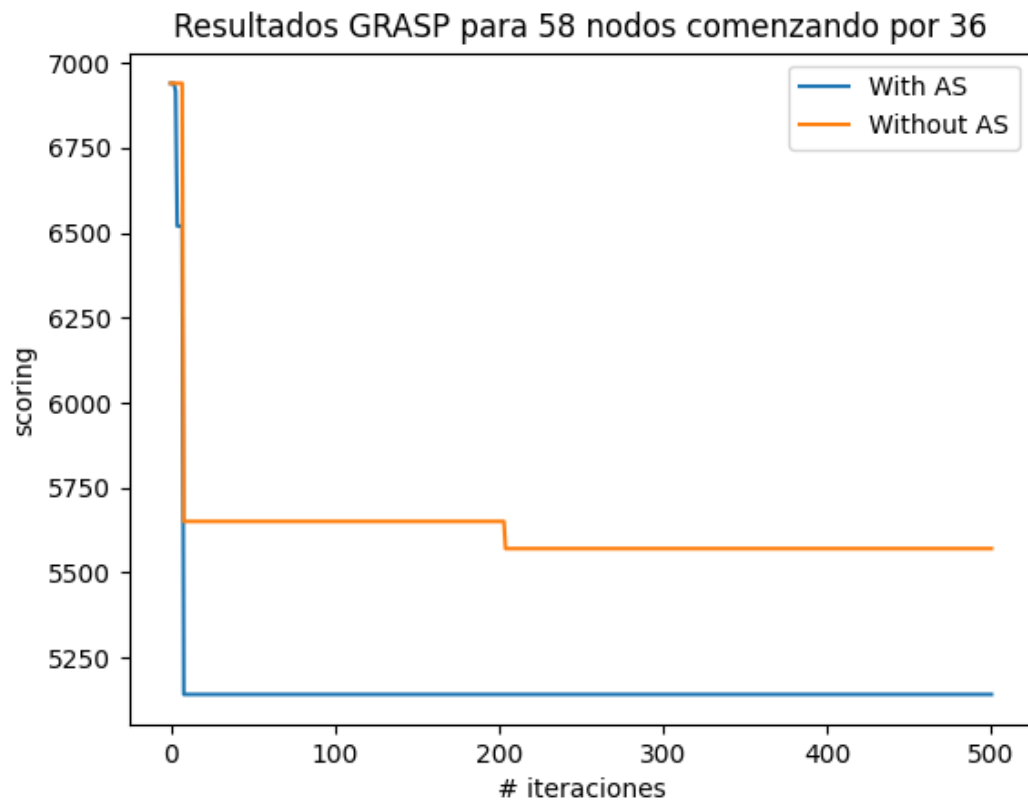
Se puede observar en los distintos gráficos que se irán presentando, que a medida que avanzan las iteraciones, encuentra un mejor resultado independientemente de la cantidad de nodos e iteraciones realizadas.

Se realizaron varios sets de pruebas para una misma instancia de N nodos donde se prueban dichas instancias para 500, 1000, 2000, 3000 y 4000 iteraciones.

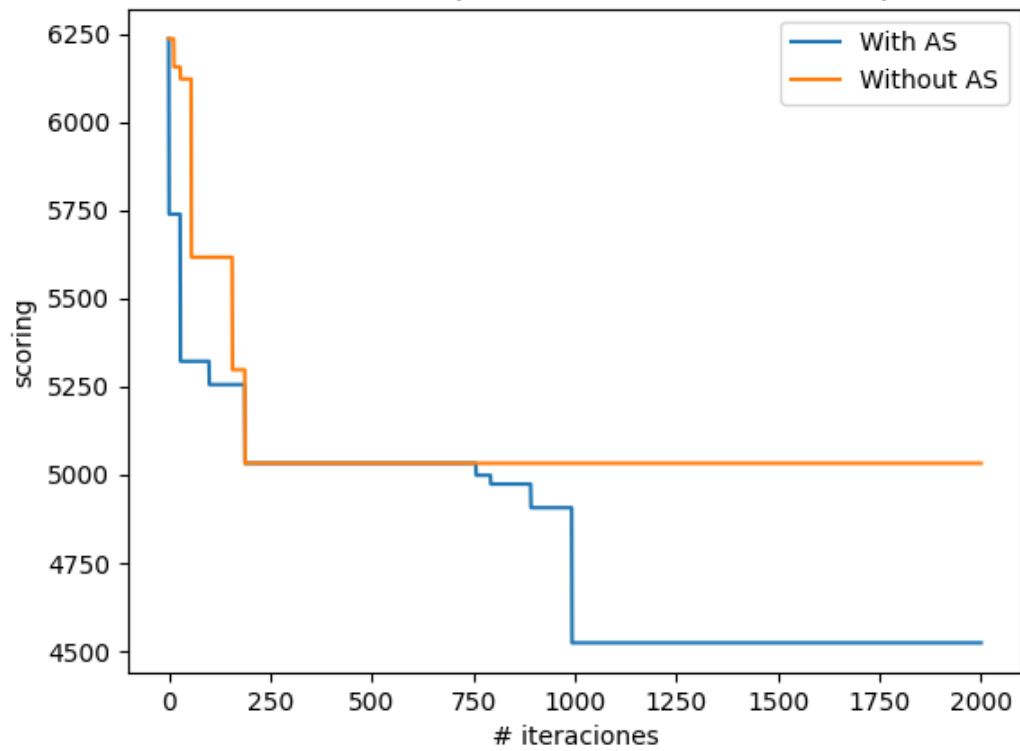
Ejercitación GRASP

Se irán presentando las iteraciones en orden creciente desde 500 hasta 4000

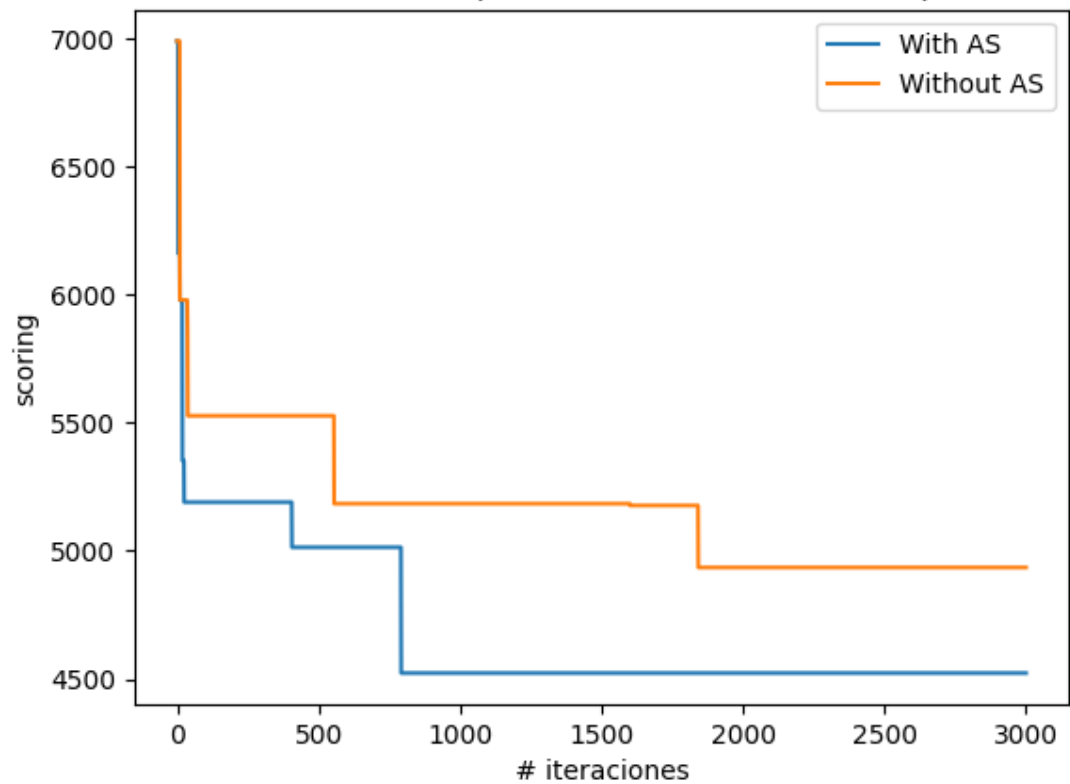
58 nodos

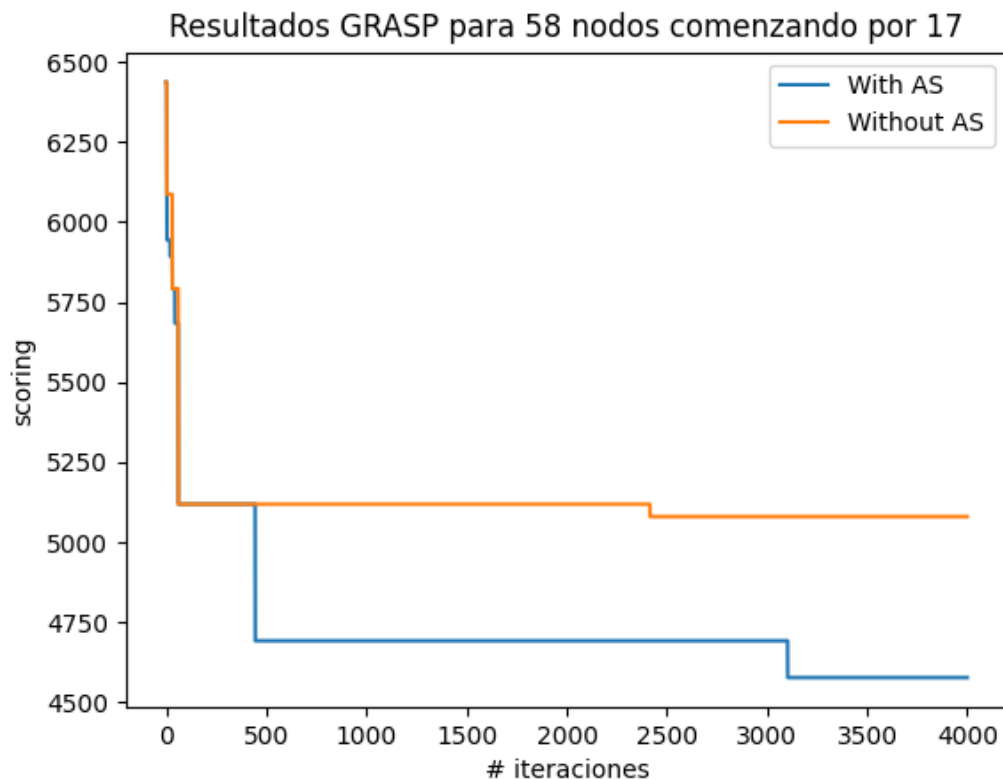


Resultados GRASP para 58 nodos comenzando por 44



Resultados GRASP para 58 nodos comenzando por 27





Se observa que a pesar que las primeras iteraciones devuelven un buen resultado para una cantidad baja de nodos, a medida que se realizan más iteraciones, encuentran un mejor resultado

58 nodos 500 iteraciones mejor resultado iteración #5

58 nodos 1000 iteraciones mejor resultado iteración #15 (a pesar de encontrar otro minimo a la iteración 1000, se descartar por mucho procesamiento ya que son casi 1000 iteraciones bajar de un minimo que ya habia encontrado)

58 nodos 2000 iteraciones mejor resultado iteración #40

58 nodos 3000 iteraciones mejor resultado iteración #20

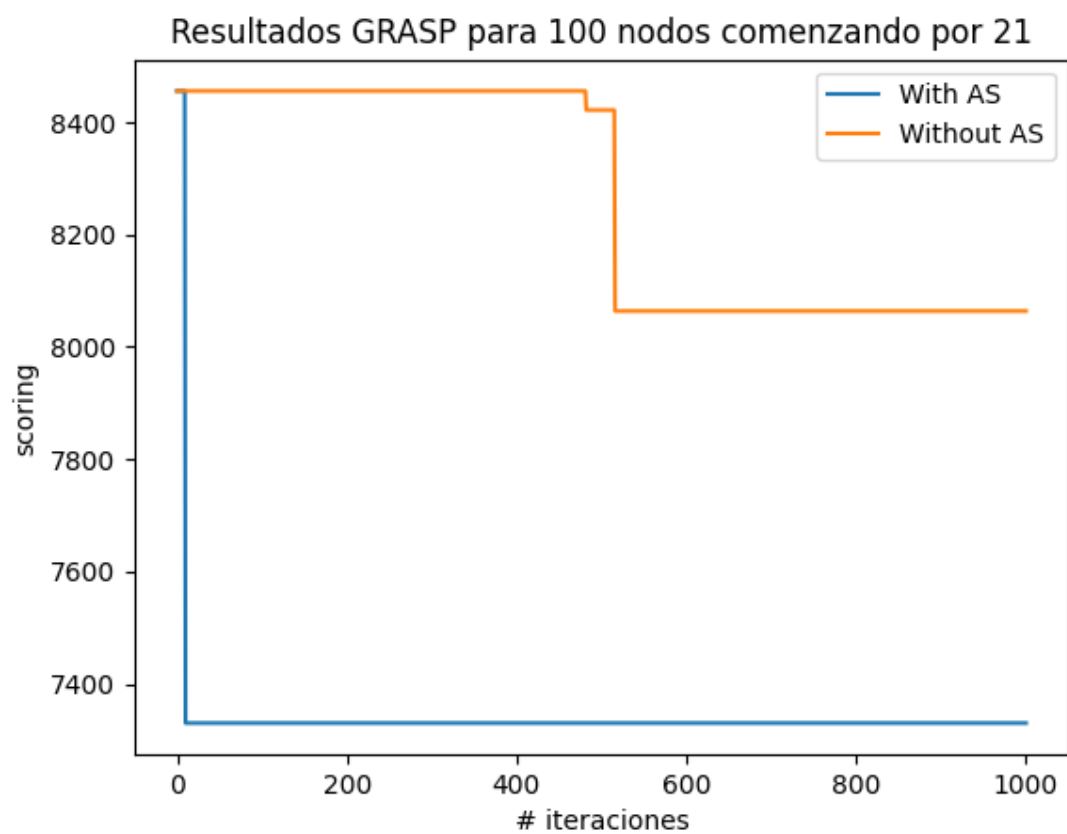
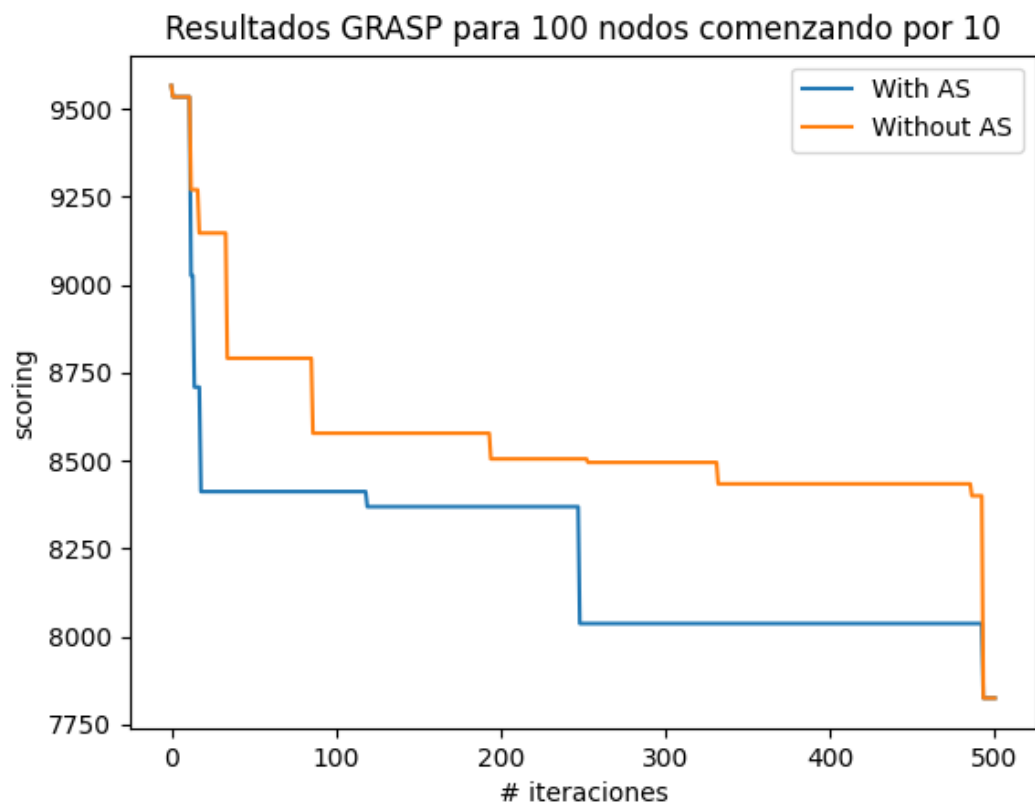
58 nodos 4000 iteraciones mejor resultado iteración #30 (a pesar de encontrar otro minimo a la iteración 3200, se descartar por mucho procesamiento ya que son casi 3000 iteraciones bajar de un minimo que ya habia encontrado)

Dado que son datos aleatorios y aproximados, se que se calcula un promedio entre la cantidad de iteraciones y la relación con los nodos

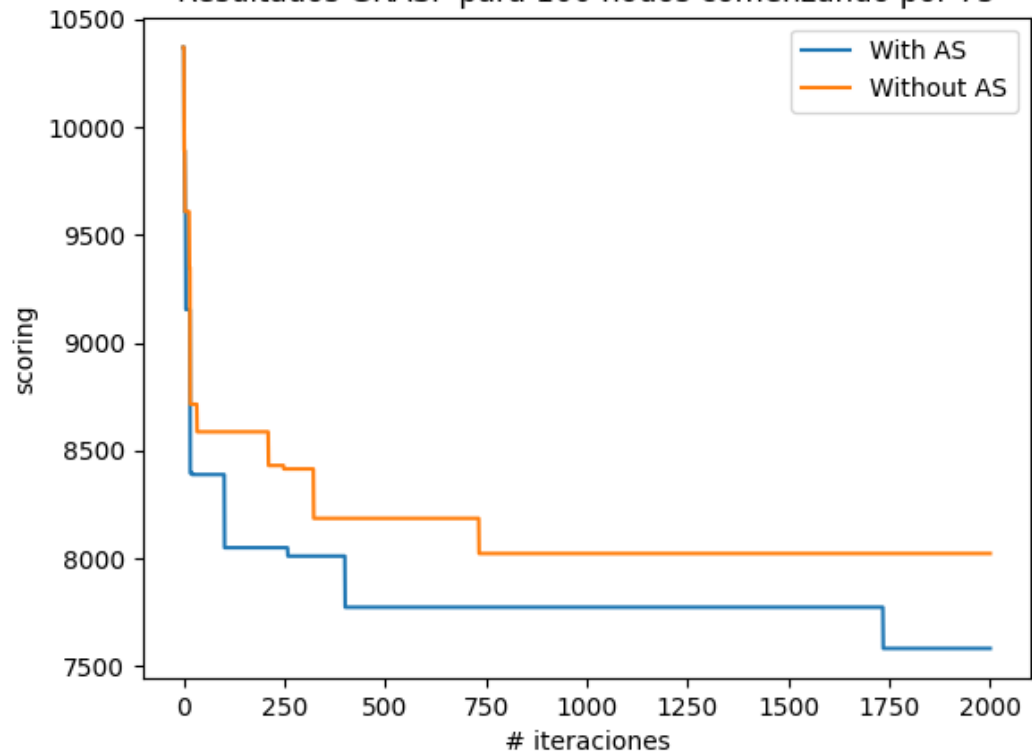
$$(5 + 15 + 40 + 20 + 30)/5$$

$$22/58 = 0.37 \text{ aprox}$$

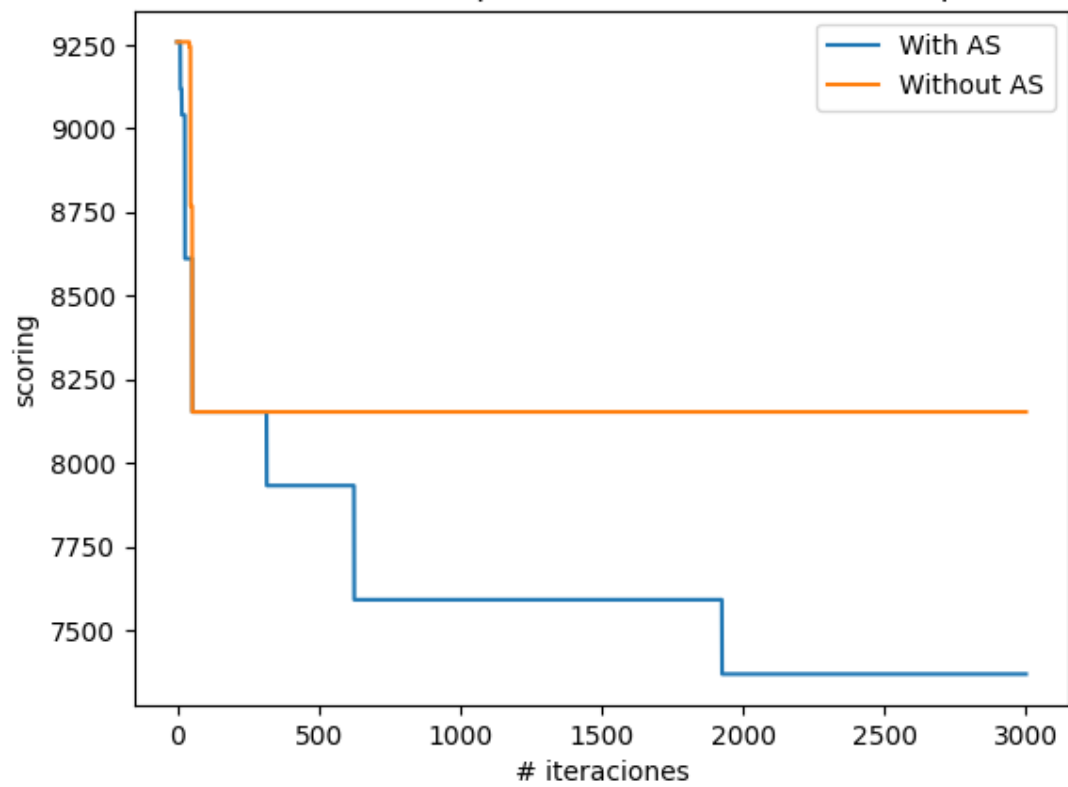
100 nodos

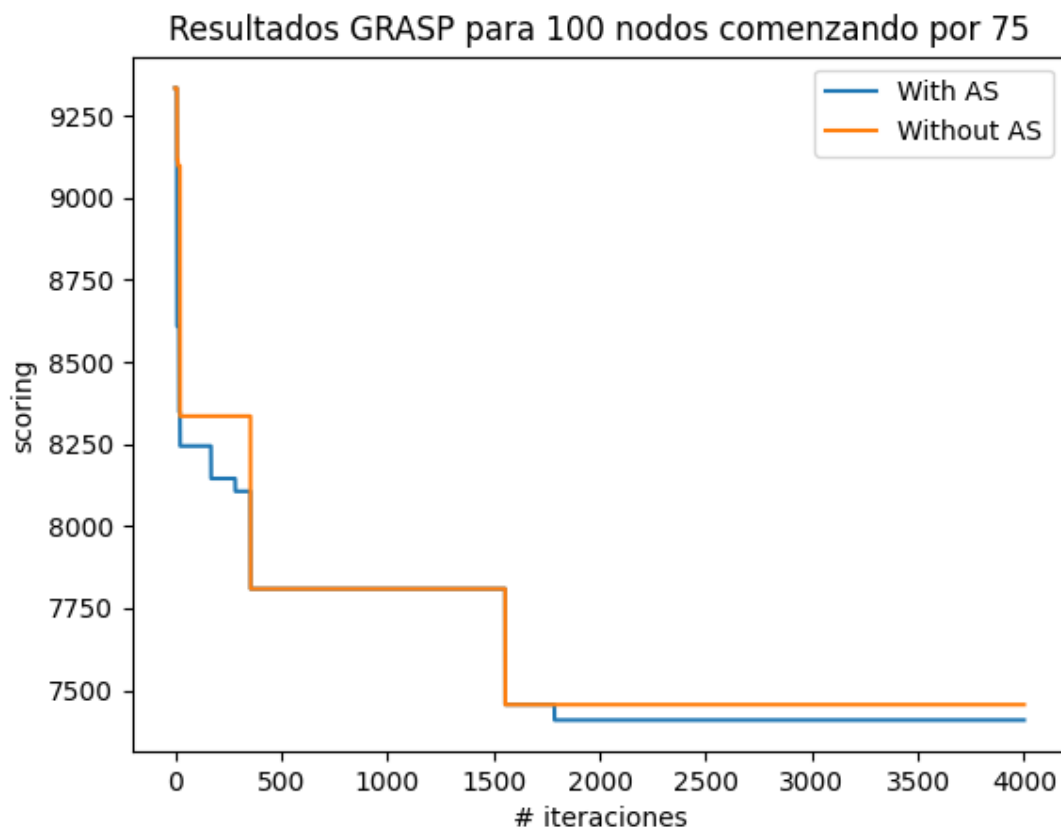


Resultados GRASP para 100 nodos comenzando por 75



Resultados GRASP para 100 nodos comenzando por 8





100 nodos 500 iteraciones mejor resultado iteración #8

100 nodos 1000 iteraciones mejor resultado iteración #5

100 nodos 2000 iteraciones mejor resultado iteración #5

100 nodos 3000 iteraciones mejor resultado iteración #270

100 nodos 4000 iteraciones mejor resultado iteración #200

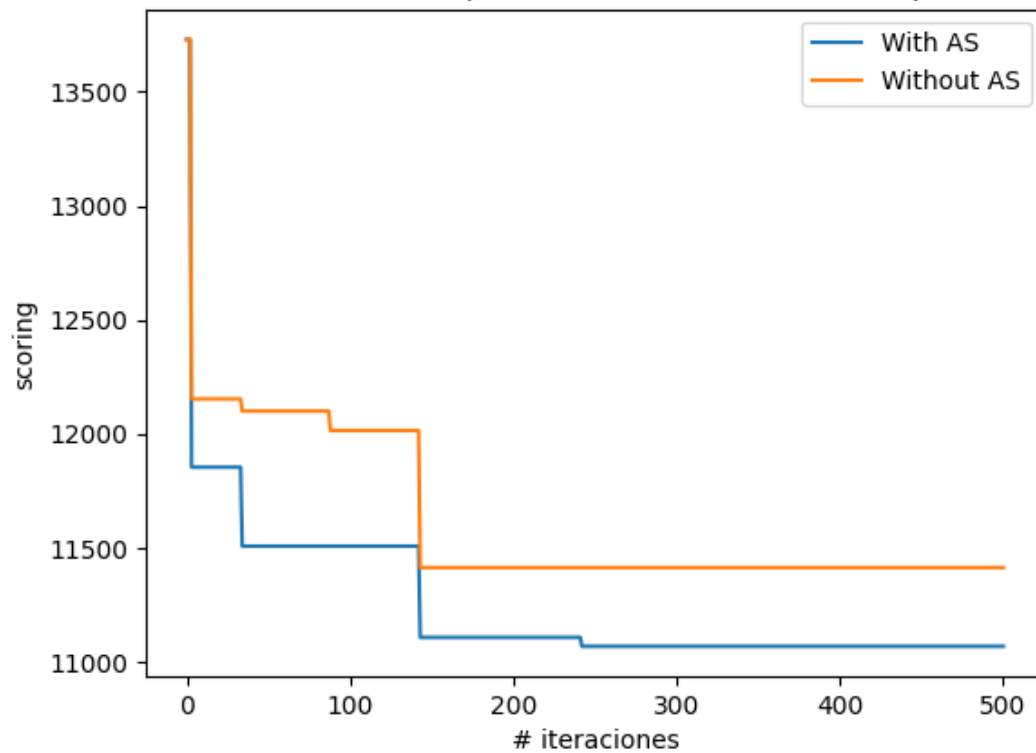
Dado que son datos aleatorios y aproximados, se que se calcula un promedio entre la cantidad de iteraciones y la relación con los nodos

$$(8 + 5 + 5 + 270 + 200)/5 = 97.6$$

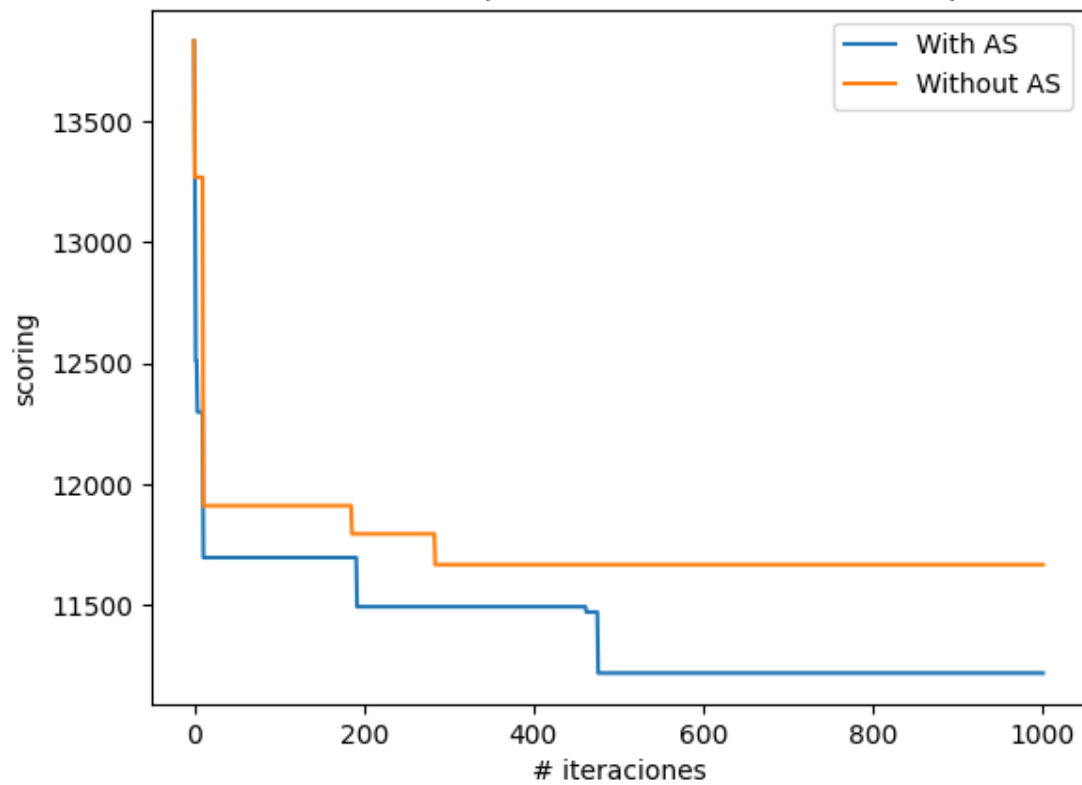
$$97.6/100 = 0.97$$

150 nodos

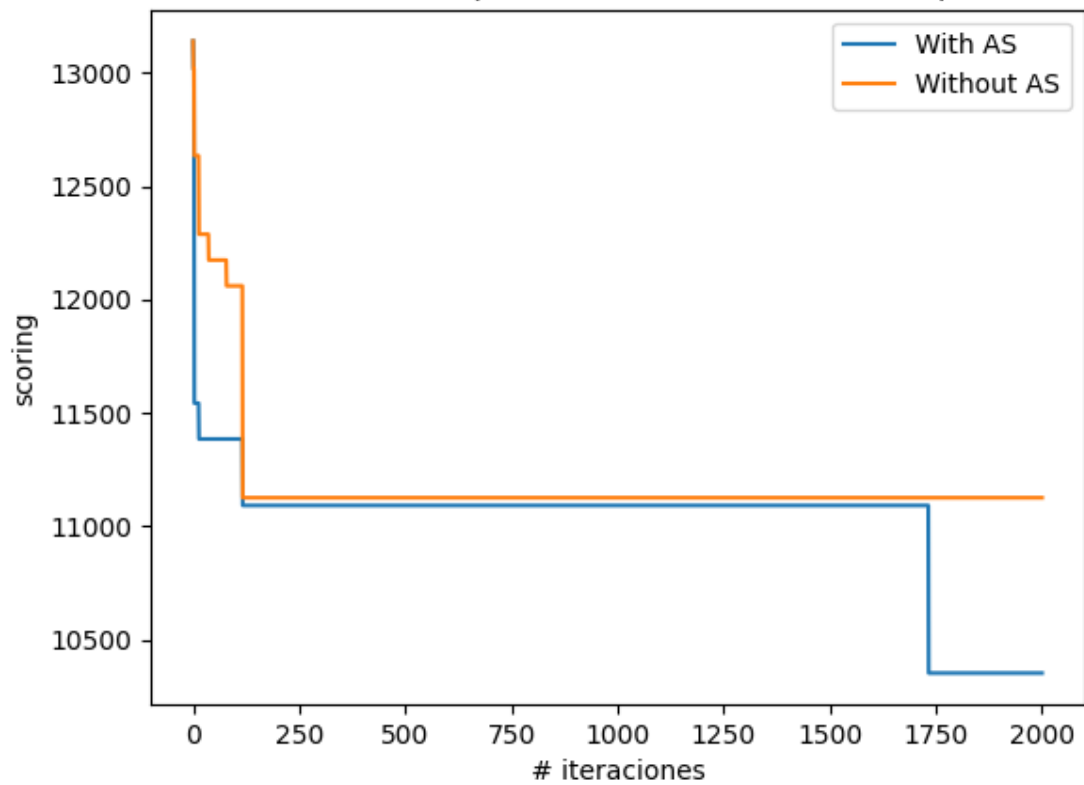
Resultados GRASP para 150 nodos comenzando por 34



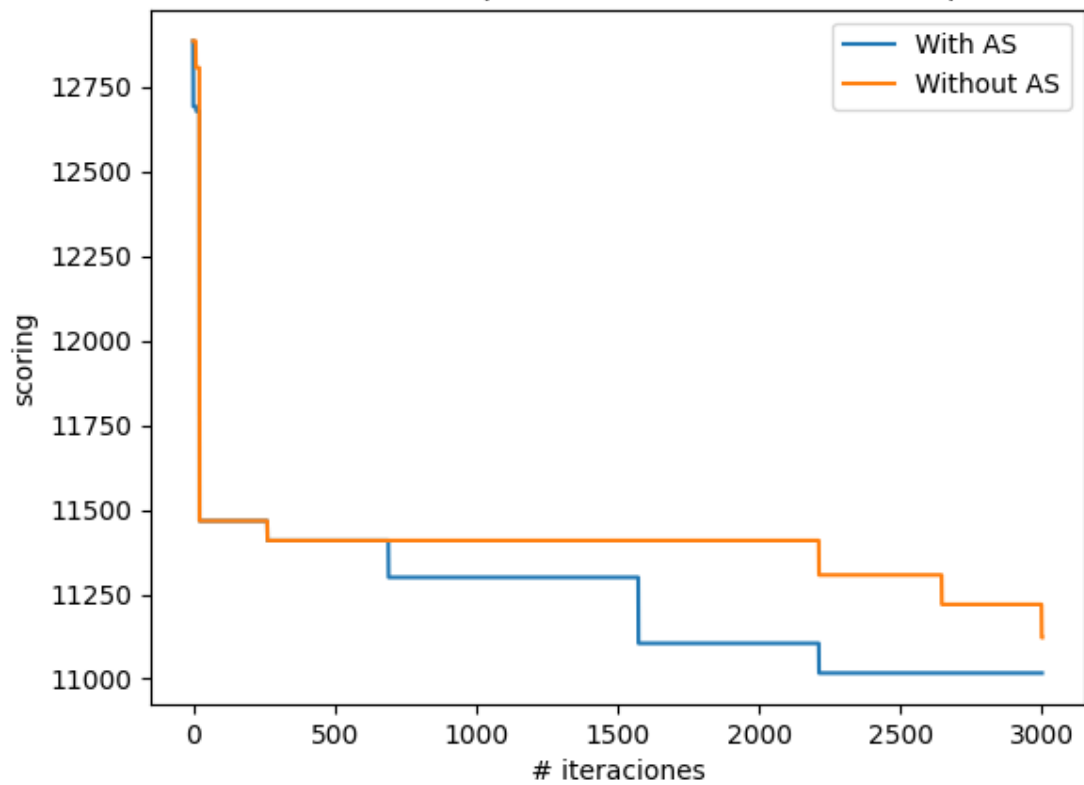
Resultados GRASP para 150 nodos comenzando por 34

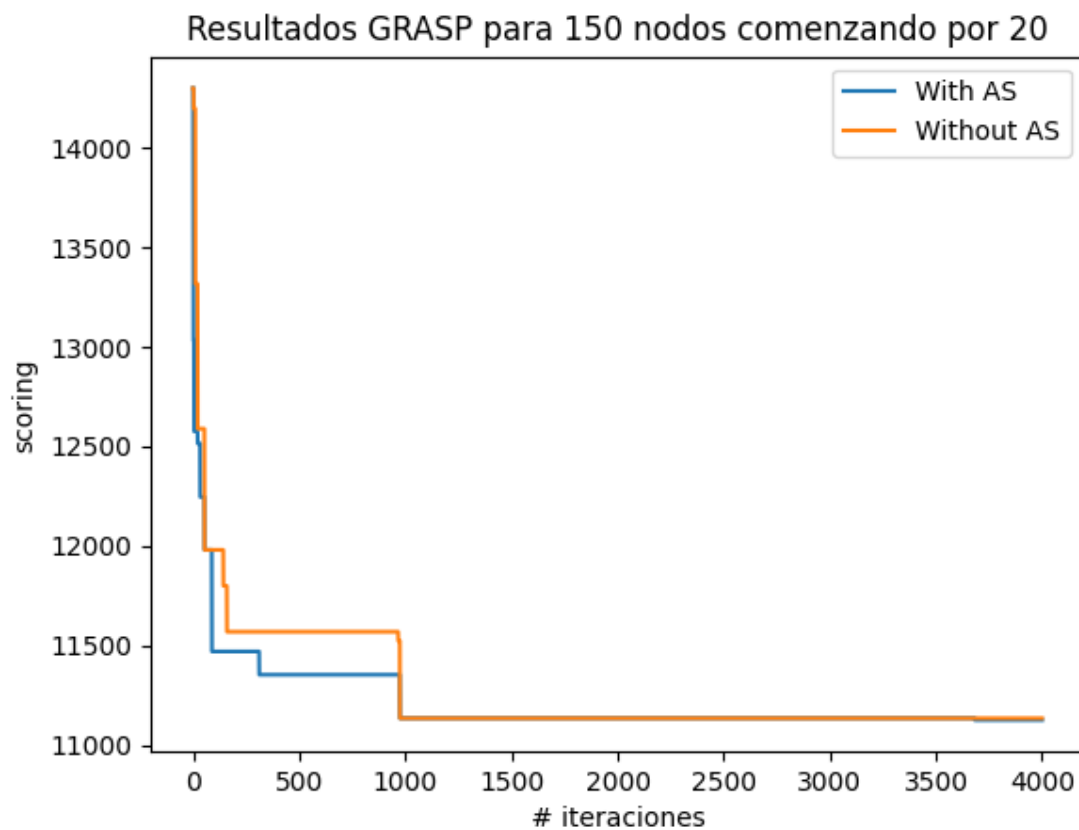


Resultados GRASP para 150 nodos comenzando por 102



Resultados GRASP para 150 nodos comenzando por 86





150 nodos 500 iteraciones mejor resultado iteración #75

150 nodos 1000 iteraciones mejor resultado iteración #5

150 nodos 2000 iteraciones mejor resultado iteración #85

150 nodos 3000 iteraciones mejor resultado iteración #200

150 nodos 4000 iteraciones mejor resultado iteración #80

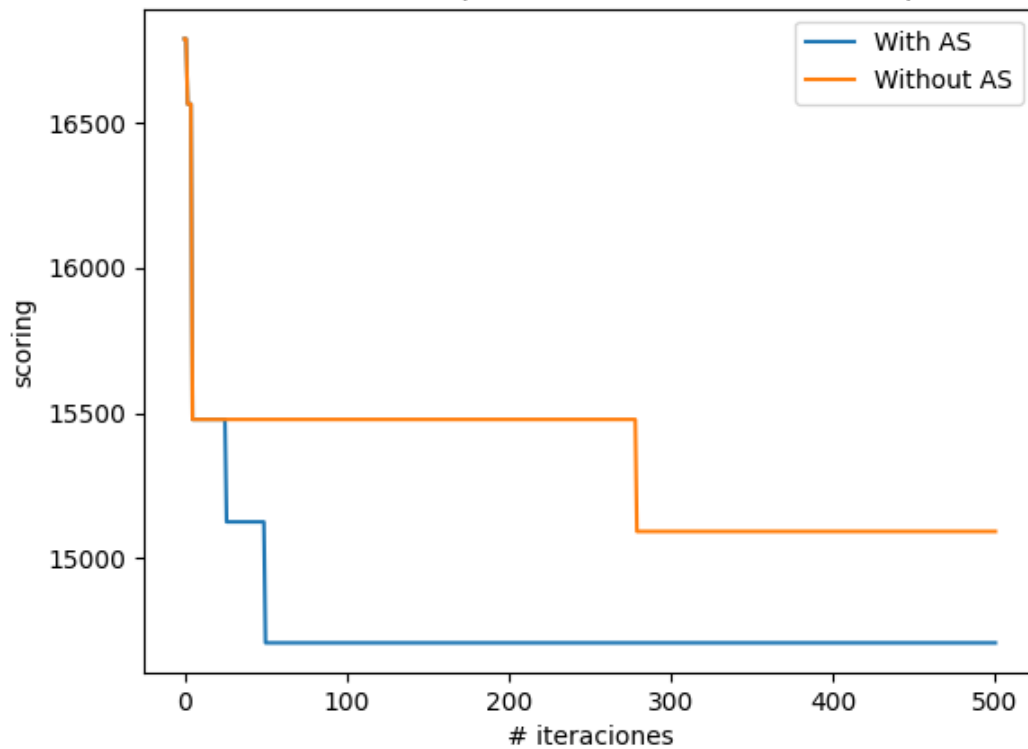
Dado que son datos aleatorios y aproximados, se que se calcula un promedio entre la cantidad de iteraciones y la relación con los nodos

$$(75 + 5 + 85 + 200 + 80)/5 = 89$$

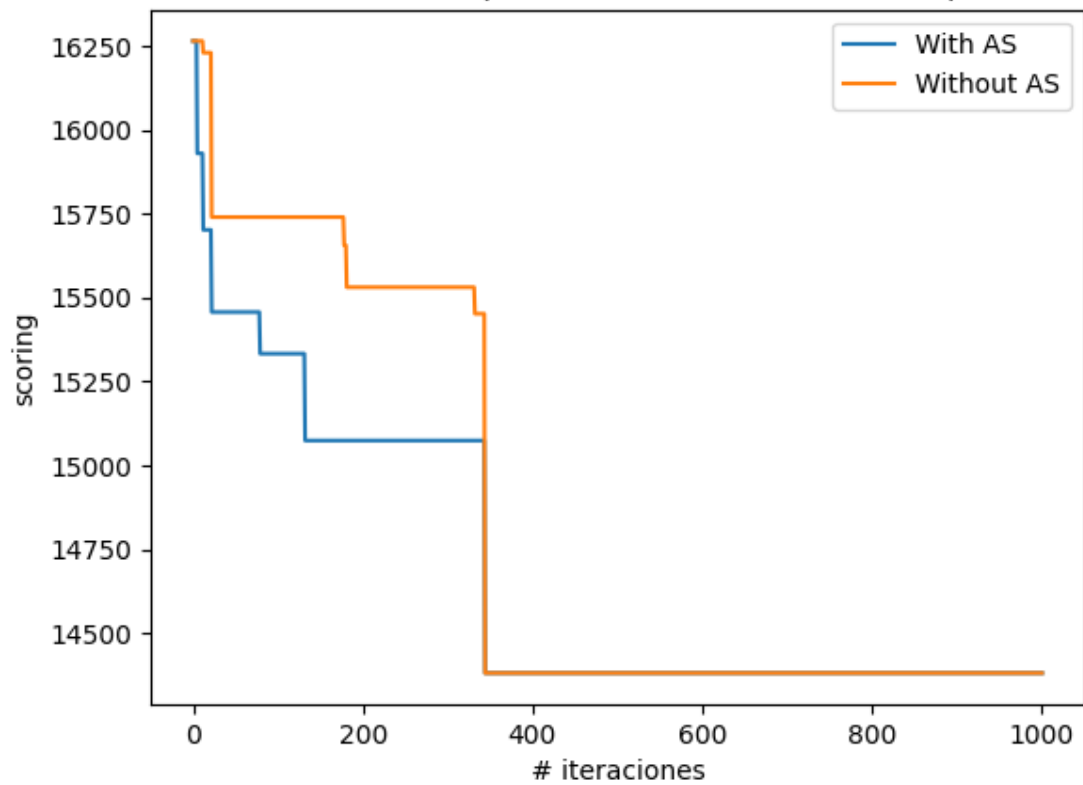
$$89/150 = 0.59$$

200 nodos

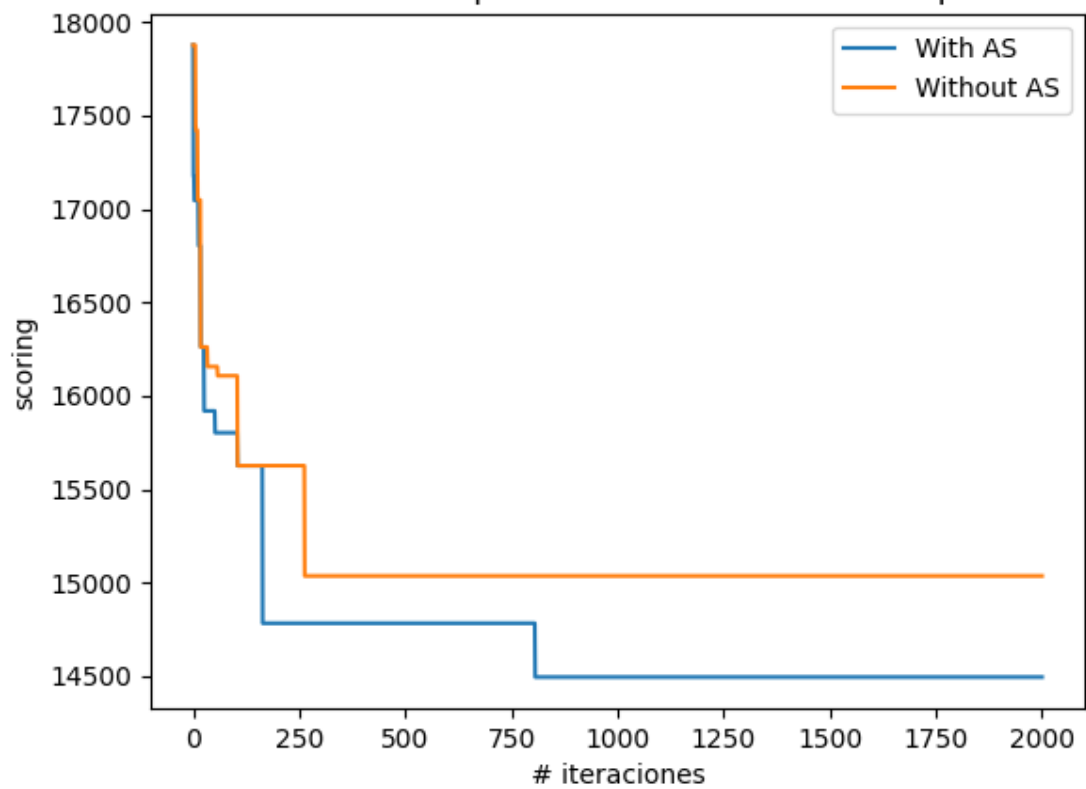
Resultados GRASP para 200 nodos comenzando por 17



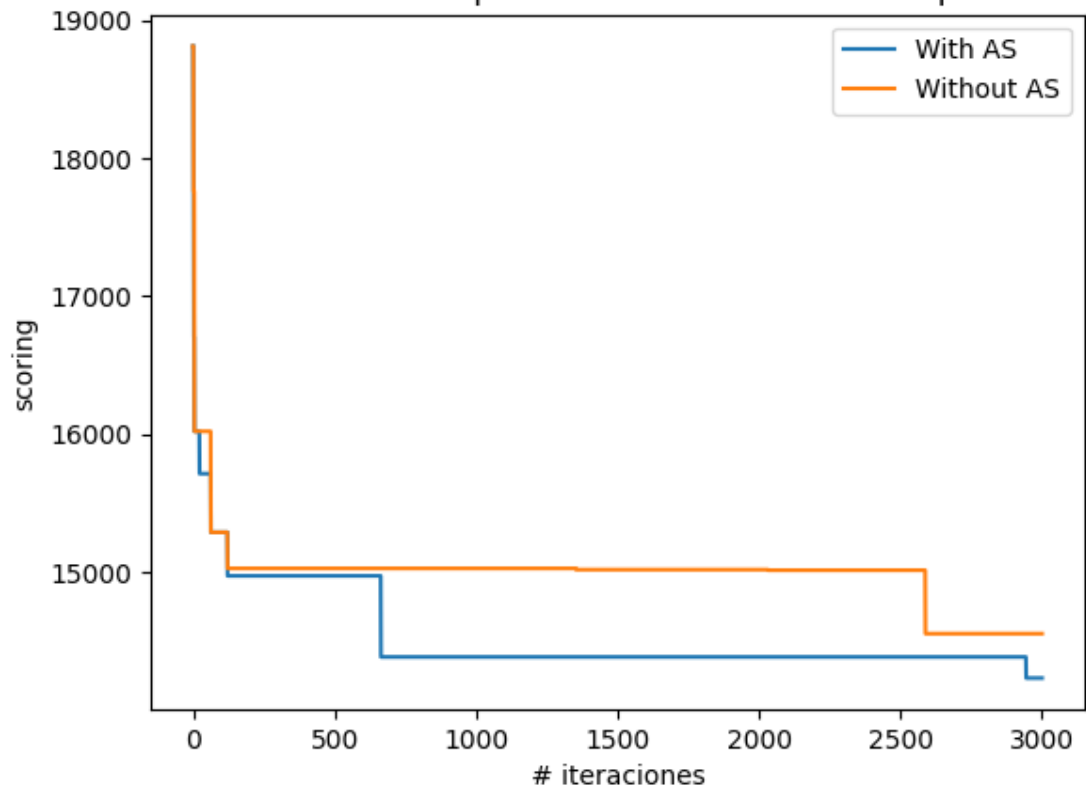
Resultados GRASP para 200 nodos comenzando por 16

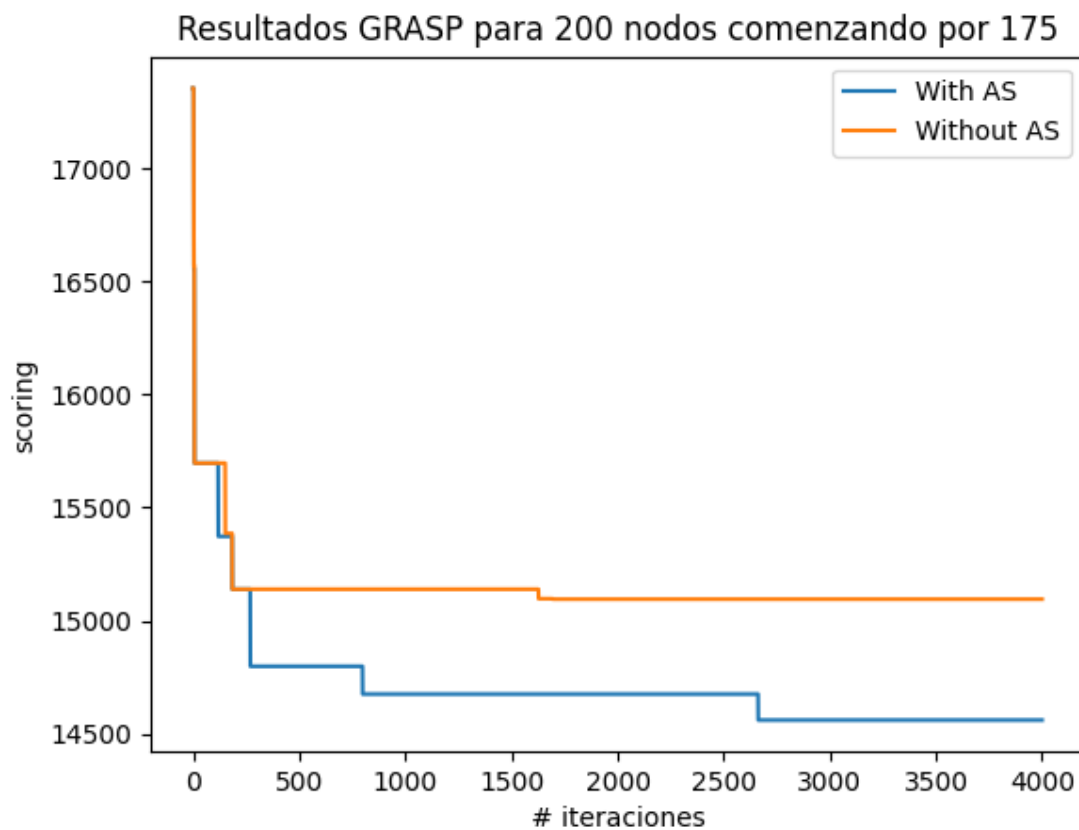


Resultados GRASP para 200 nodos comenzando por 159



Resultados GRASP para 200 nodos comenzando por 123





200 nodos 500 iteraciones mejor resultado iteración #50

200 nodos 1000 iteraciones mejor resultado iteración #190

200 nodos 2000 iteraciones mejor resultado iteración #180 (a pesar de encontrar una mejor a la iteración casi 800 pasaron varias iteraciones ocupando mucho cómputo)

200 nodos 3000 iteraciones mejor resultado iteración #180 (a pesar de encontrar una mejor a la iteración 700 pasaron varias iteraciones ocupando mucho cómputo)

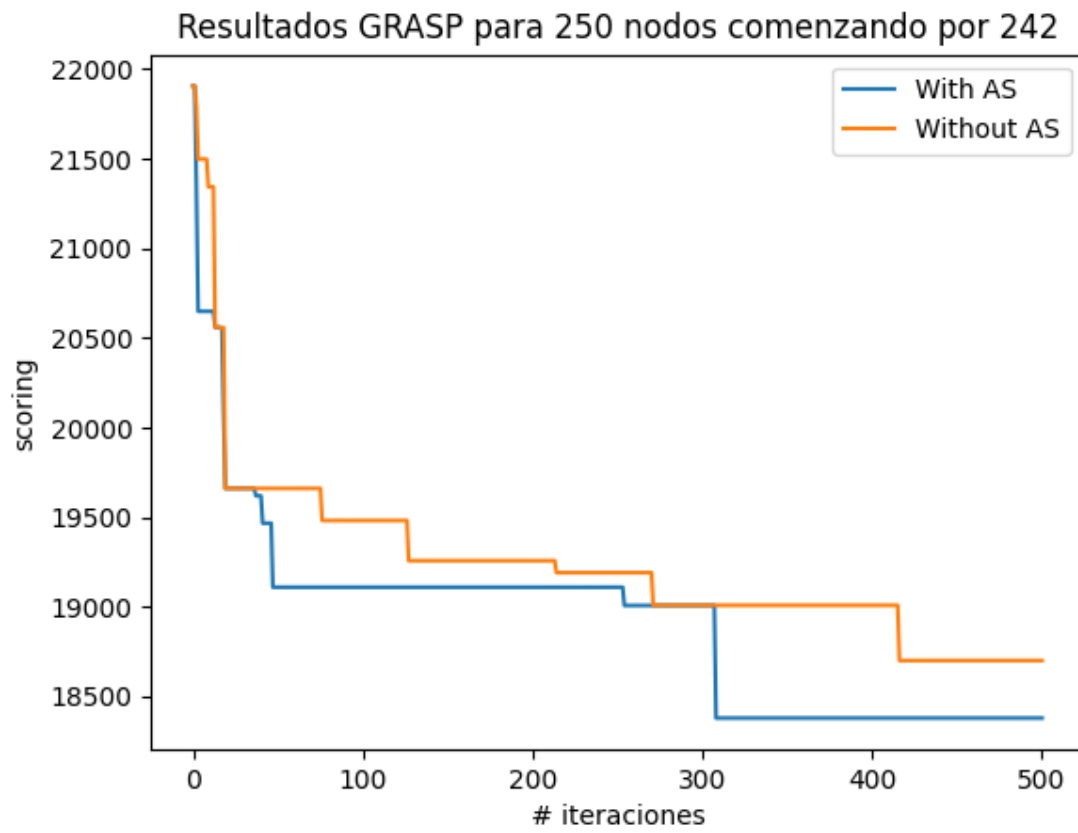
200 nodos 4000 iteraciones mejor resultado iteración #250 a pesar de encontrar mejores soluciones por ejemplo en la iteración #900 y #2700 la mejora es muy poca por lo que se descarta

Es por eso que se calcula un promedio

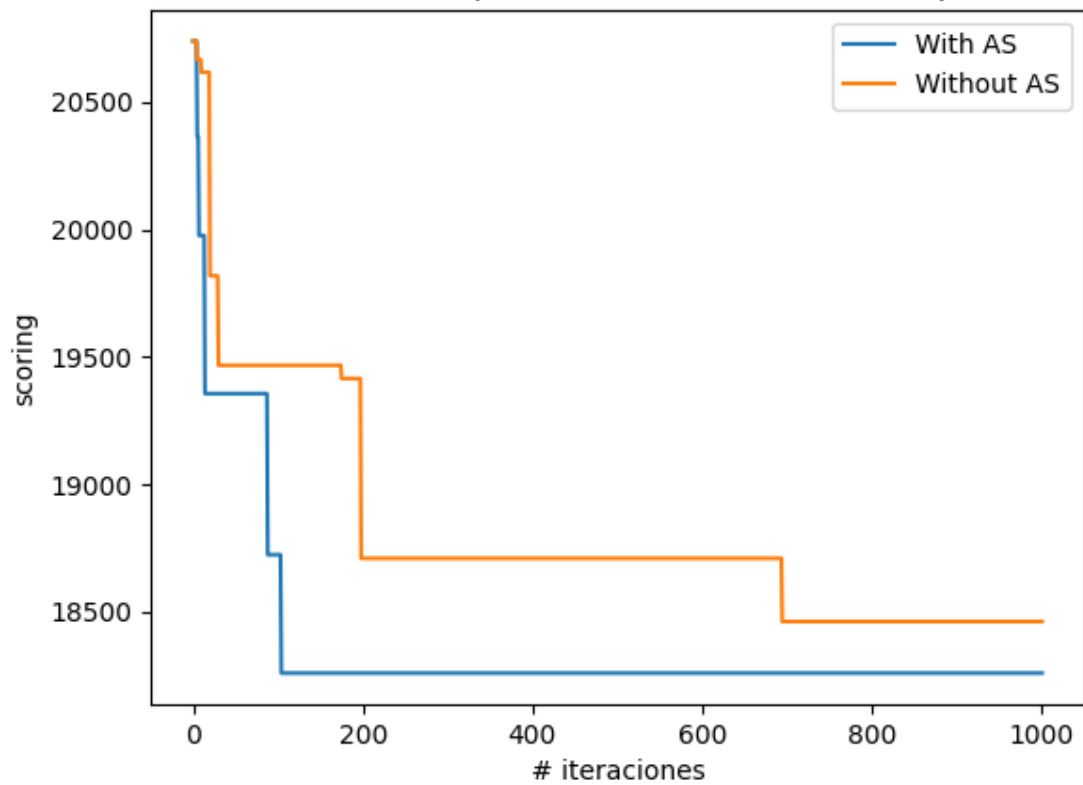
$$(50 + 190 + 180 + 180 + 250)/5 = 170$$

$$170/200 = 0.85$$

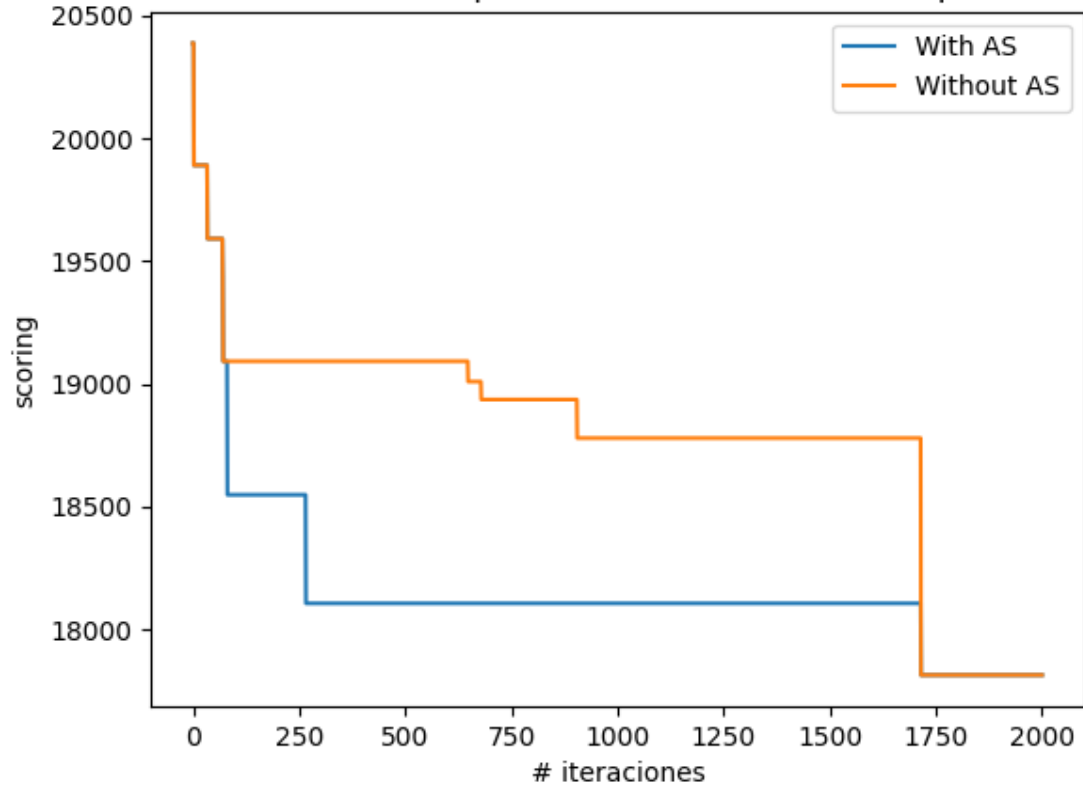
250 nodos



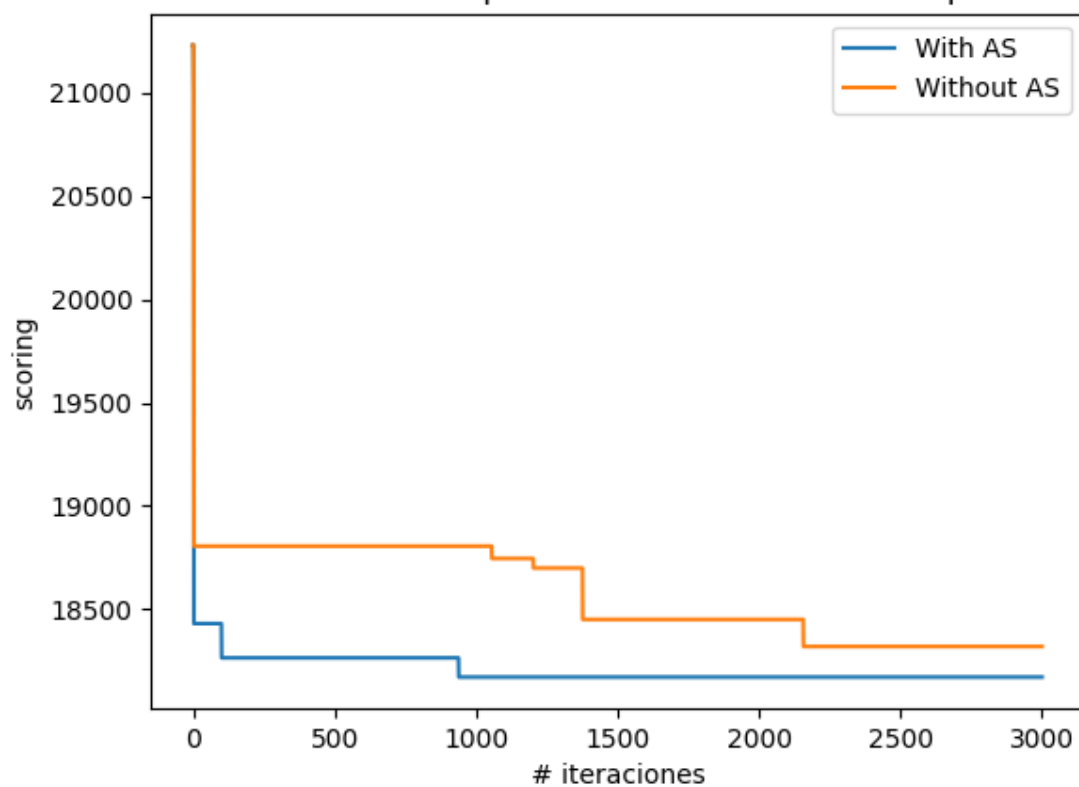
Resultados GRASP para 250 nodos comenzando por 248



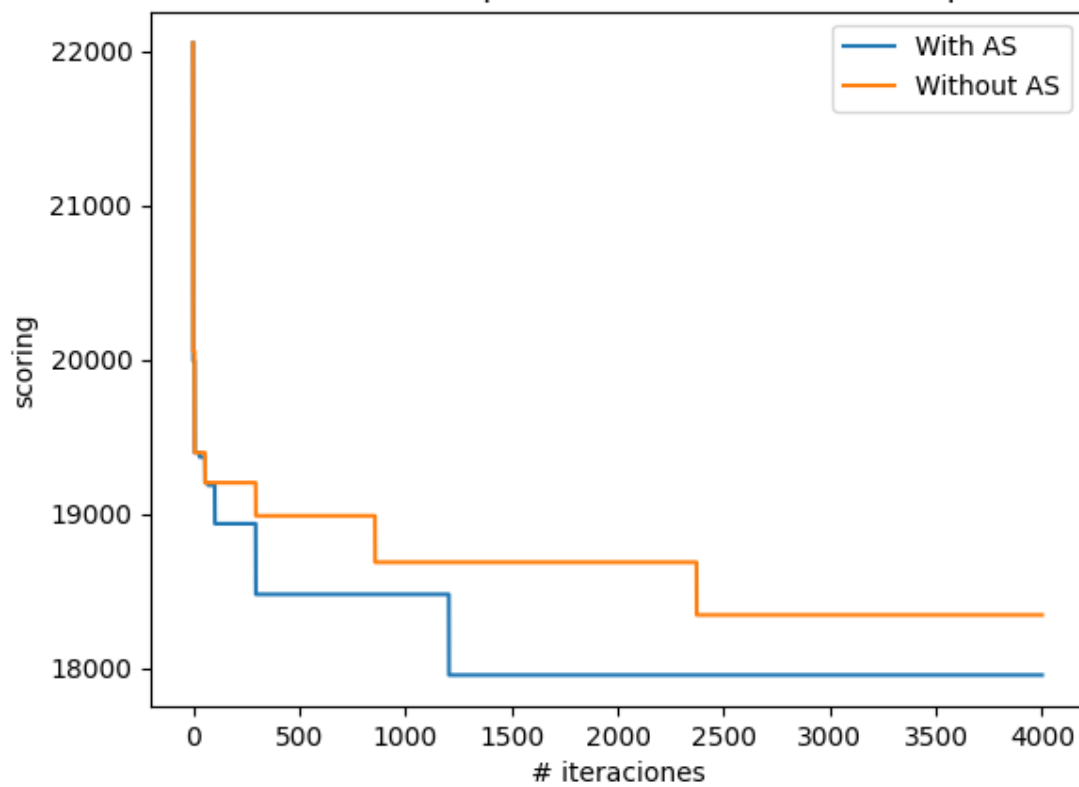
Resultados GRASP para 250 nodos comenzando por 114



Resultados GRASP para 250 nodos comenzando por 77



Resultados GRASP para 250 nodos comenzando por 55



250 nodos 500 iteraciones mejor resultado iteración #50 a pesar de encontrar un mejor resultado a la vuelta 320 no logró una gran mejora y le costo varias iteraciones.

250 nodos 1000 iteraciones mejor resultado iteración #180

250 nodos 2000 iteraciones mejor resultado iteración #250 a pesar de encontrar un mejor resultado a la vuelta 1750 no logró una gran mejora y le costo varias iteraciones.

250 nodos 3000 iteraciones mejor resultado iteración #200 a pesar de encontrar un mejor resultado a la vuelta 1300 no logró una gran mejora y le costo varias iteraciones.

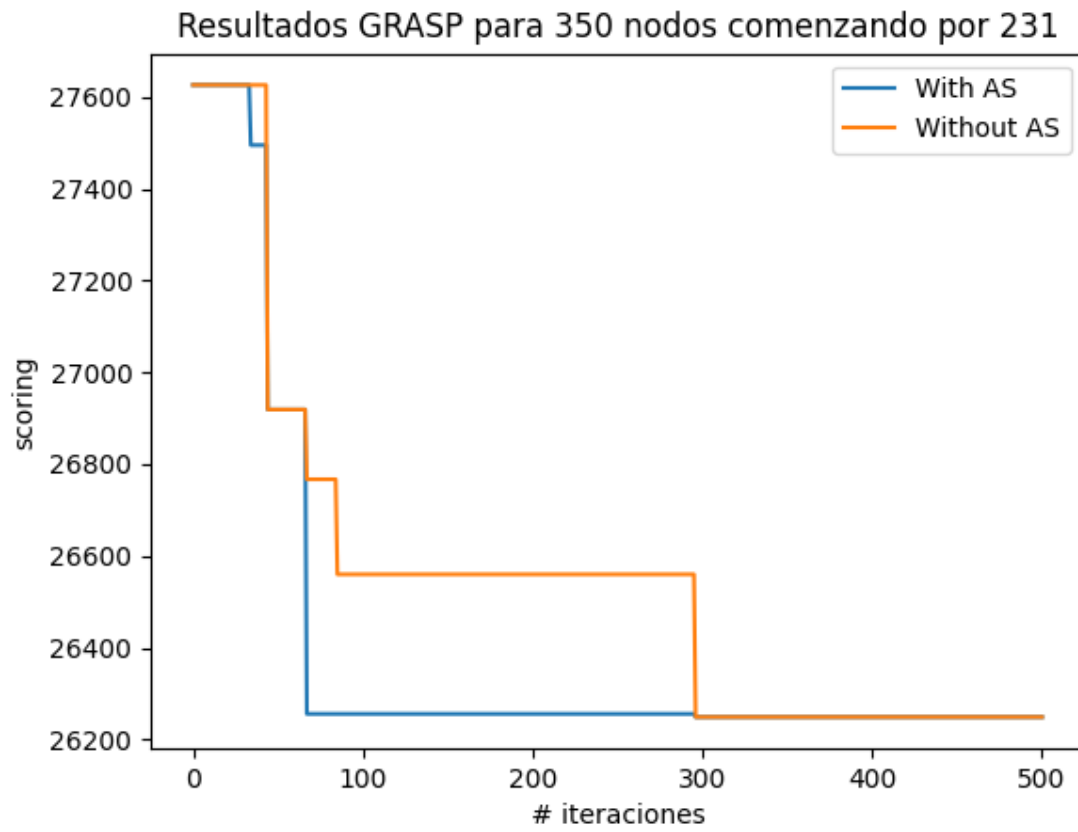
250 nodos 4000 iteraciones mejor resultado iteración #400 a pesar de encontrar un mejor resultado a la vuelta 1300 no logró una gran mejora.

Dado que son datos aleatorios y aproximados, se que se calcula un promedio entre la cantidad de iteraciones y la relación con los nodos

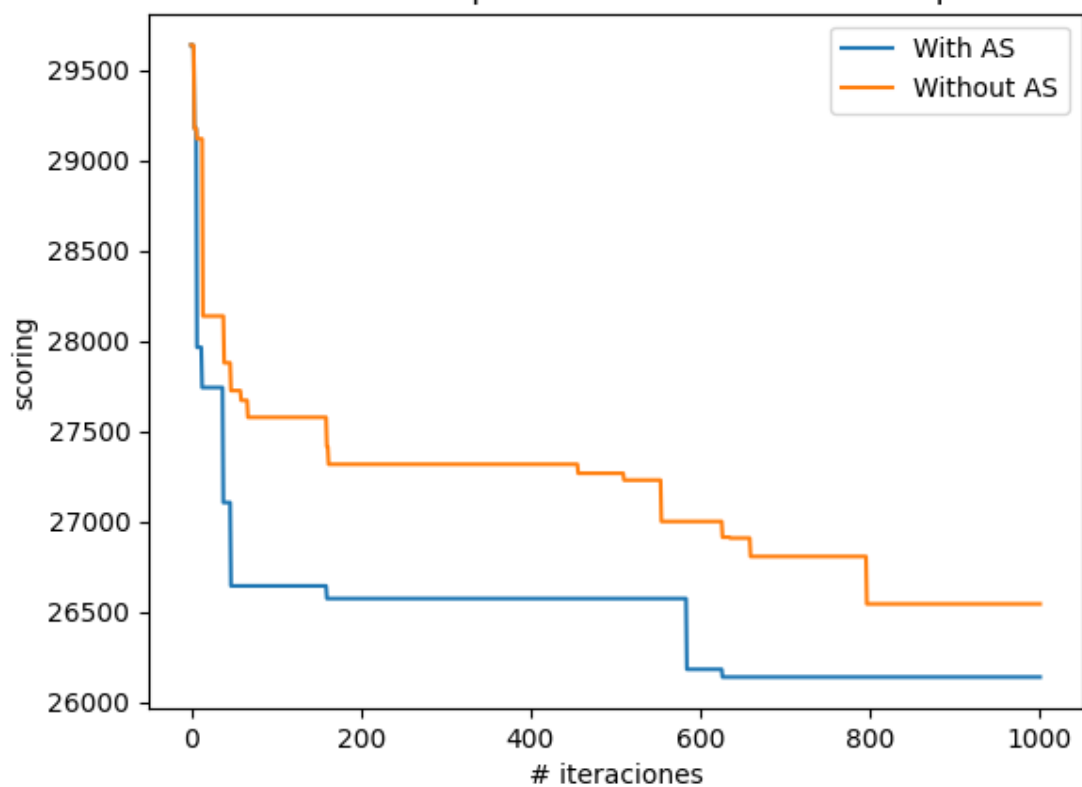
$$(50 + 180 + 250 + 200 + 400)/5 = 216$$

$$216/250 = 0.86$$

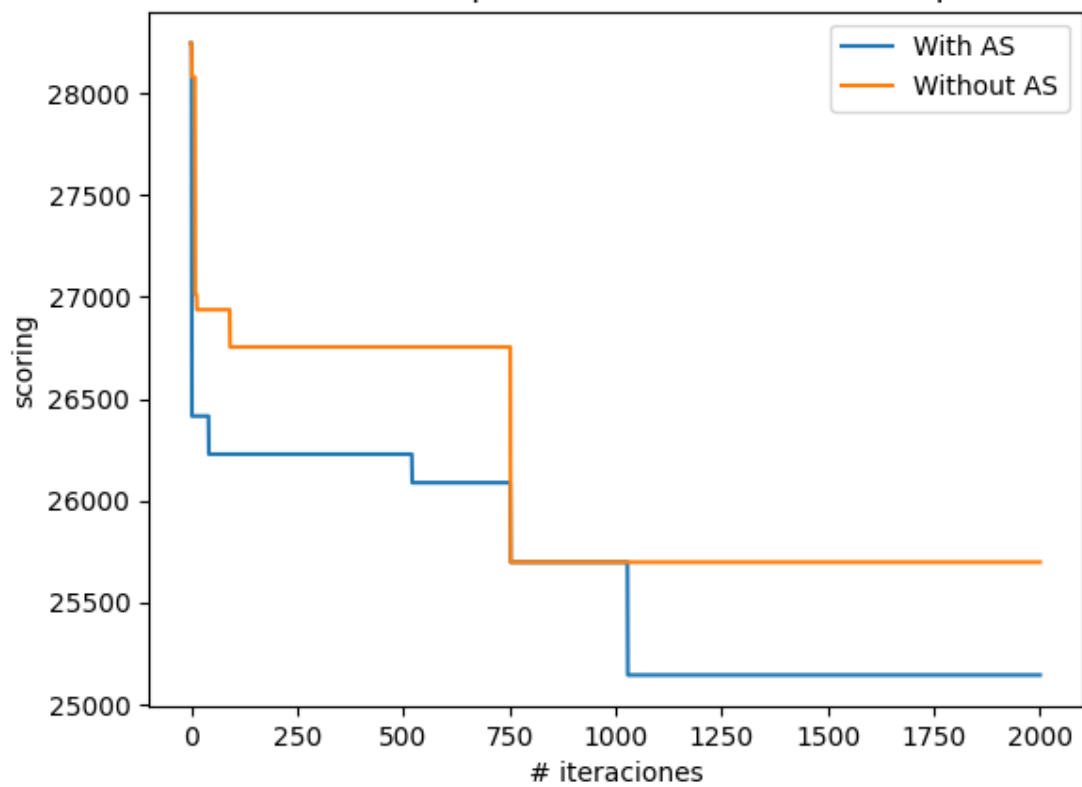
350 nodos



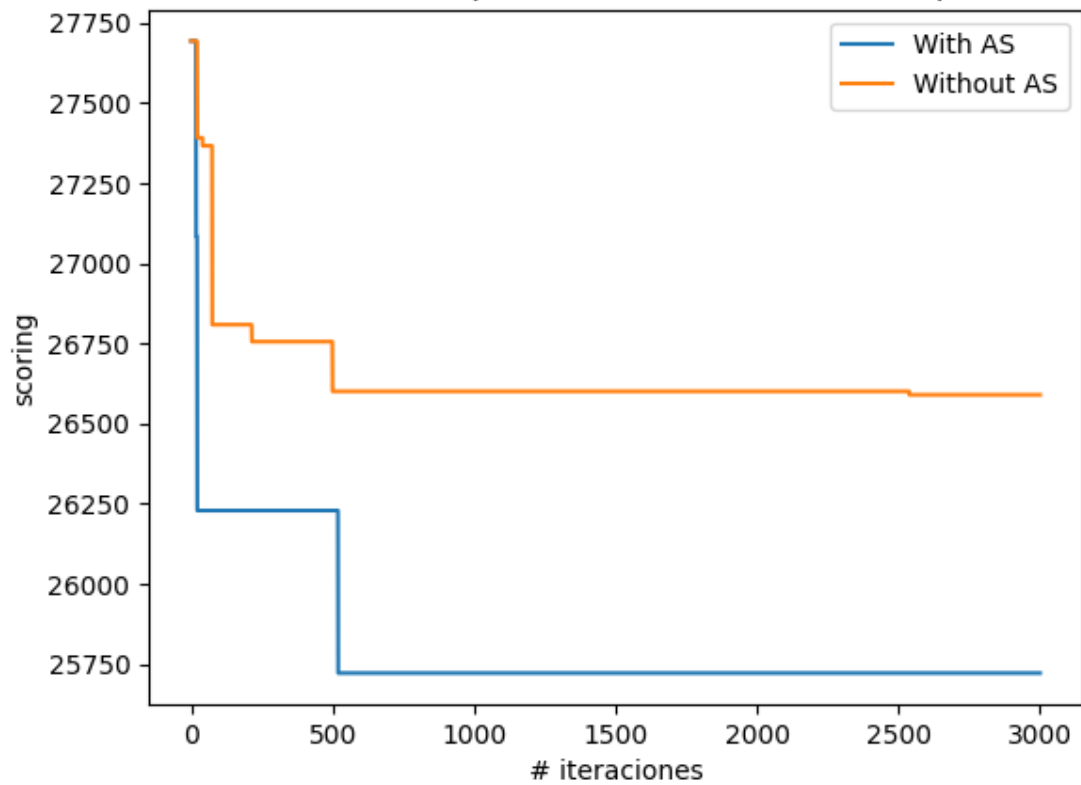
Resultados GRASP para 350 nodos comenzando por 170



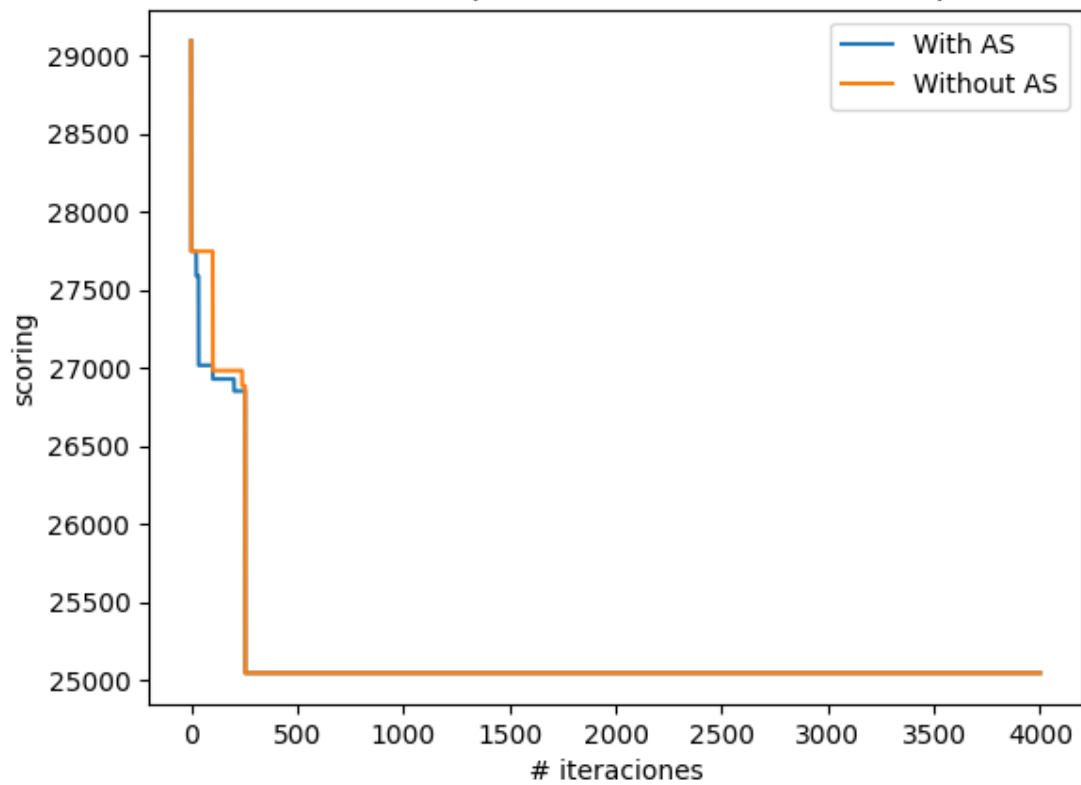
Resultados GRASP para 350 nodos comenzando por 153



Resultados GRASP para 350 nodos comenzando por 293



Resultados GRASP para 350 nodos comenzando por 284



350 nodos 500 iteraciones mejor resultado iteración #120

350 nodos 1000 iteraciones mejor resultado iteración #200

350 nodos 2000 iteraciones mejor resultado iteración #250

350 nodos 3000 iteraciones mejor resultado iteración #100

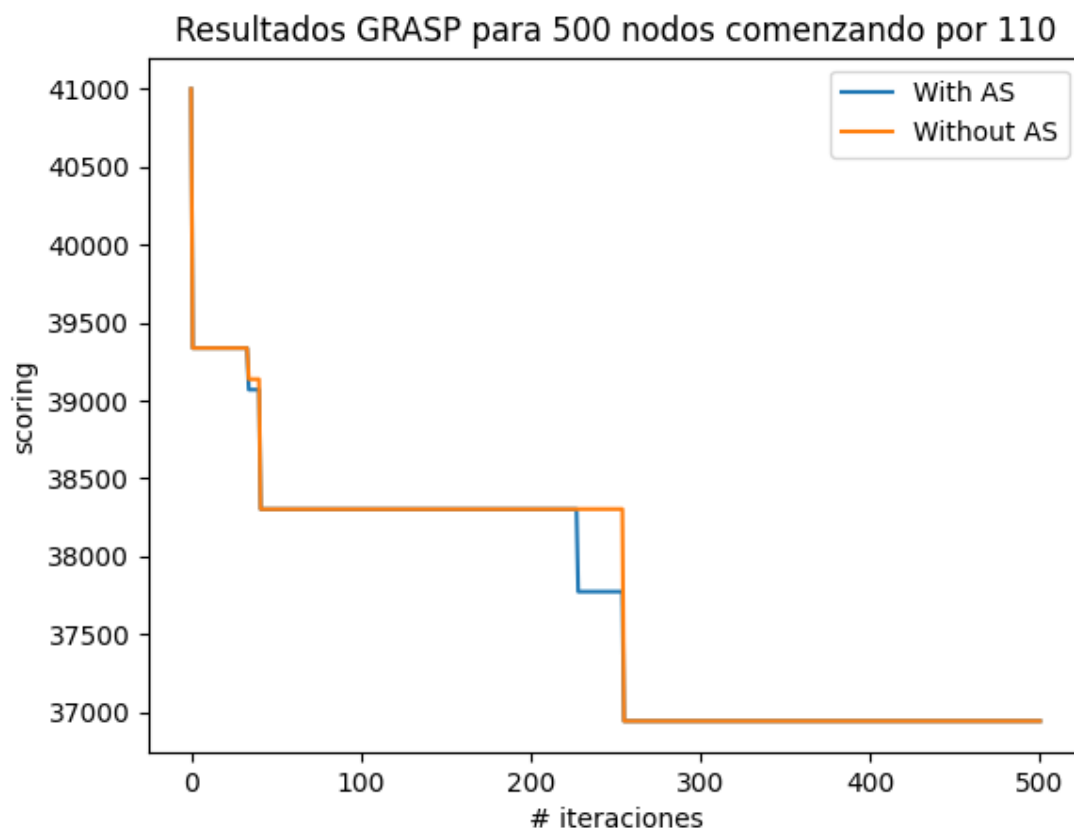
350 nodos 4000 iteraciones mejor resultado iteración #300

Dado que son datos aleatorios y aproximados, se que se calcula un promedio entre la cantidad de iteraciones y la relación con los nodos

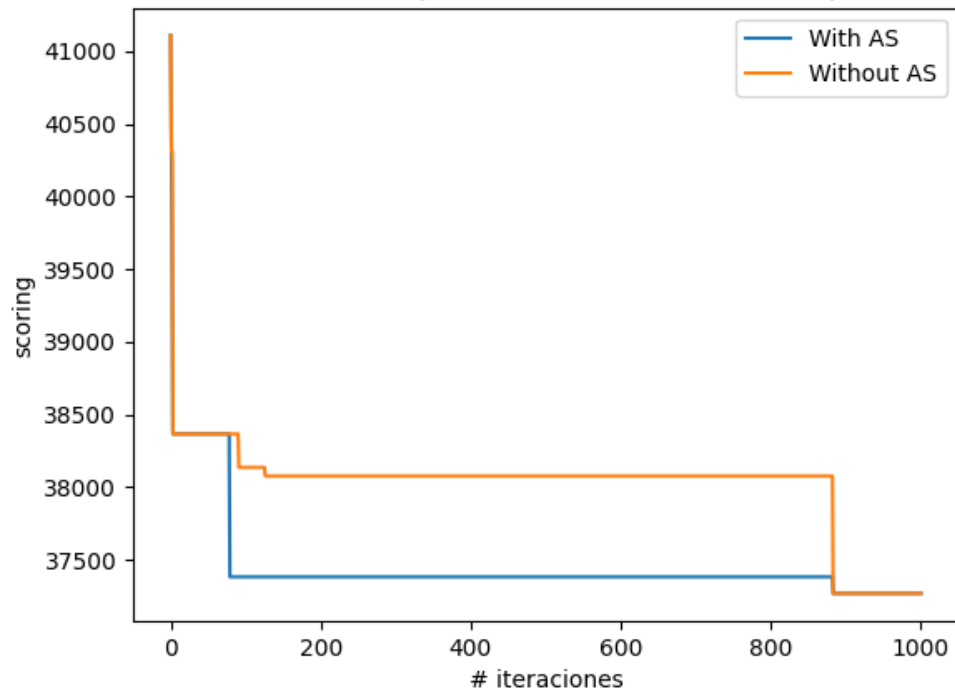
$$(120+200+250+100+300)/5 = 194$$

$$194/350 = 0.55$$

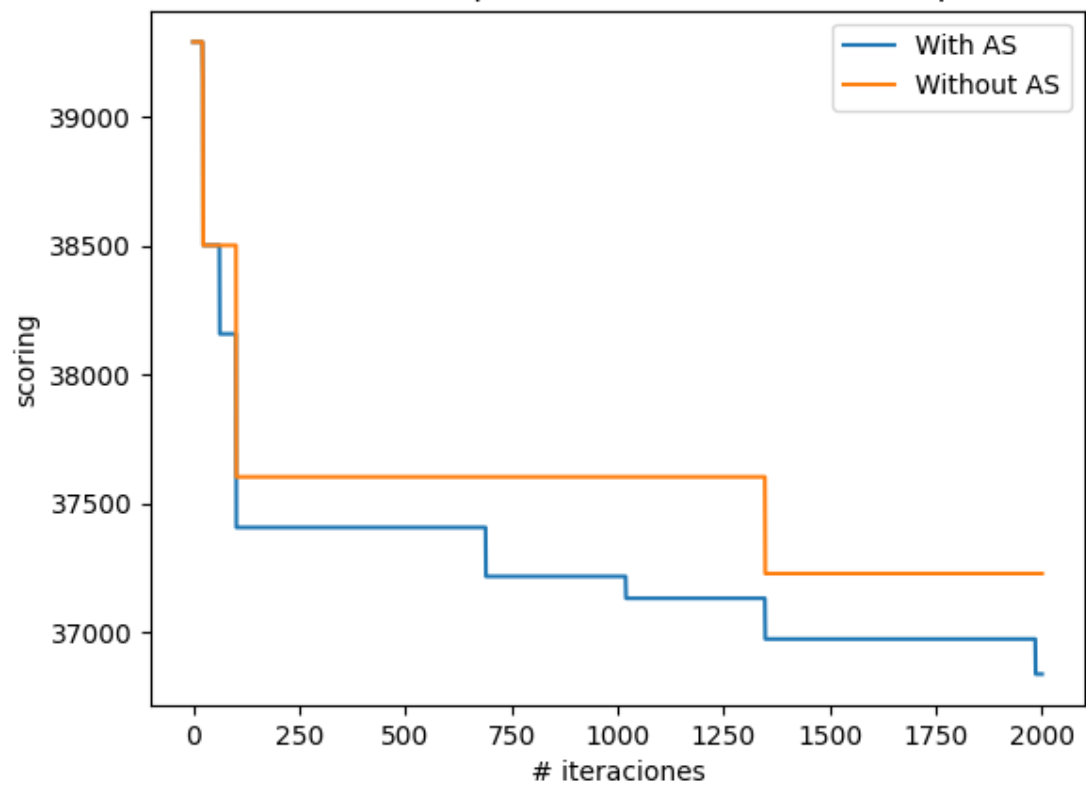
500 nodos



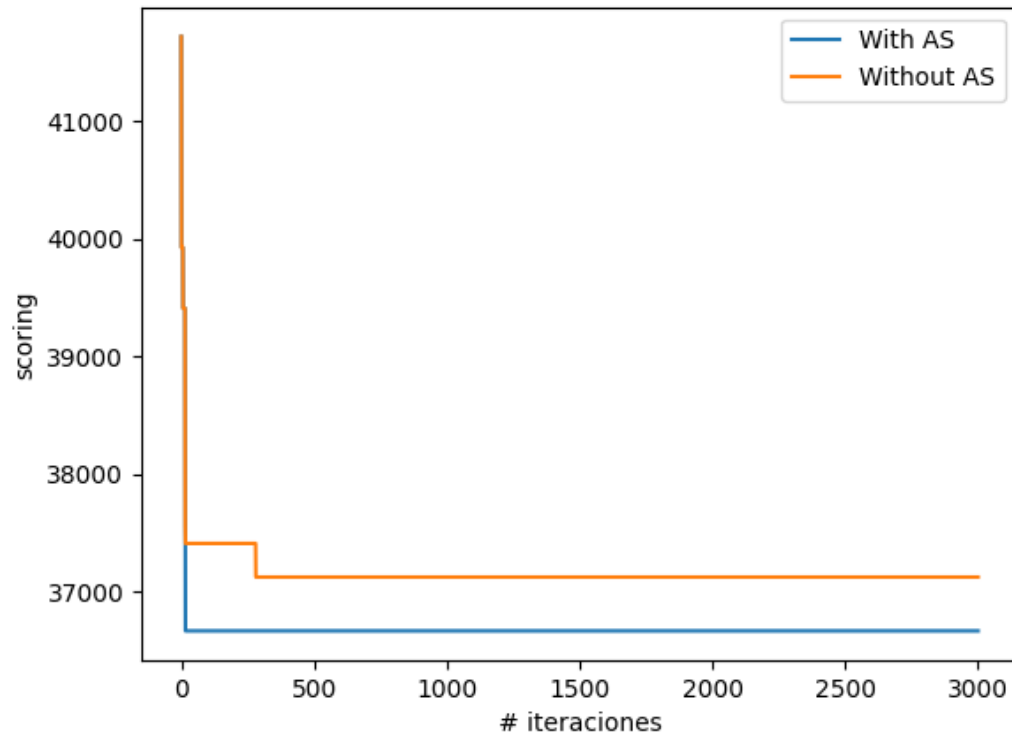
Resultados GRASP para 500 nodos comenzando por 144



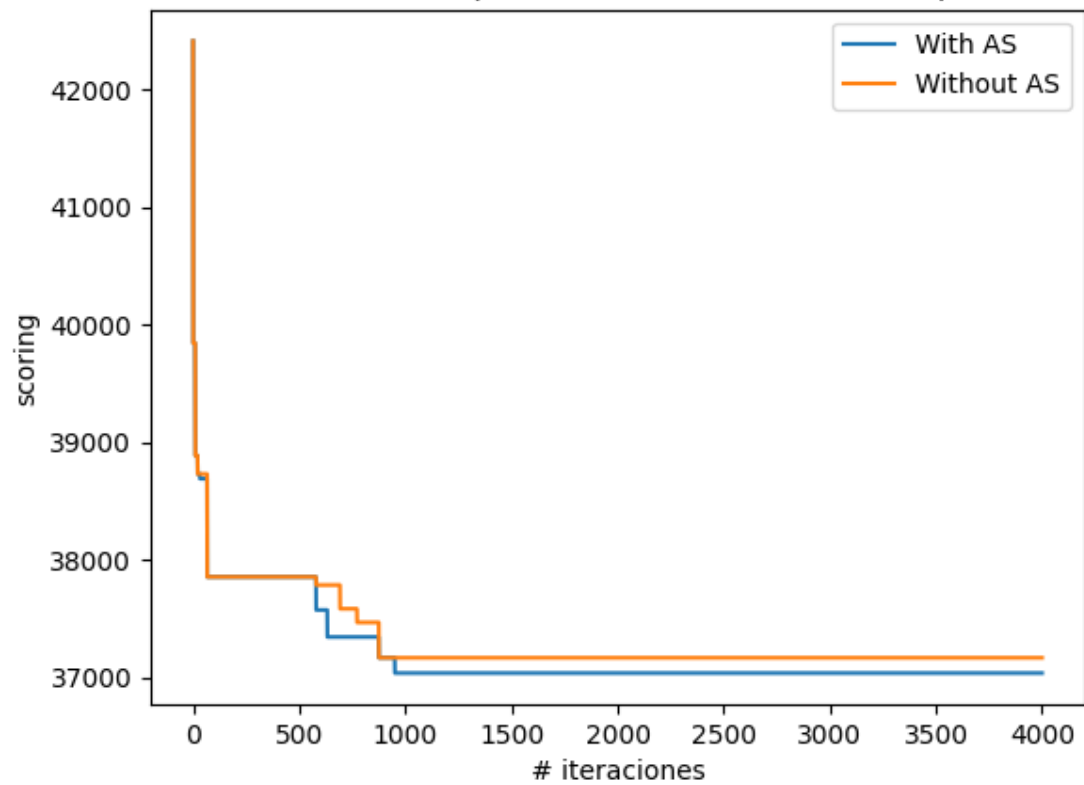
Resultados GRASP para 500 nodos comenzando por 431



Resultados GRASP para 500 nodos comenzando por 188



Resultados GRASP para 500 nodos comenzando por 194



500 nodos 500 iteraciones mejor resultado iteración #250

500 nodos 1000 iteraciones mejor resultado iteración #180 a pesar de encontrar un mejor en iteraciones posteriores no se logra una mayor mejora

500 nodos 2000 iteraciones mejor resultado iteración #220 a pesar de encontrar un mejor en iteraciones posteriores no se logra una mayor mejora

500 nodos 3000 iteraciones mejor resultado iteración #10

500 nodos 4000 iteraciones mejor resultado iteración #500 a pesar de encontrar un mejor en iteraciones posteriores no se logra una mayor mejora

Dado que son datos aleatorios y aproximados, se que se calcula un promedio entre la cantidad de iteraciones y la relación con los nodos

$$(250 + 180 + 220 + 10 + 500)/5 = 232$$

$$232/500 = 0.46$$

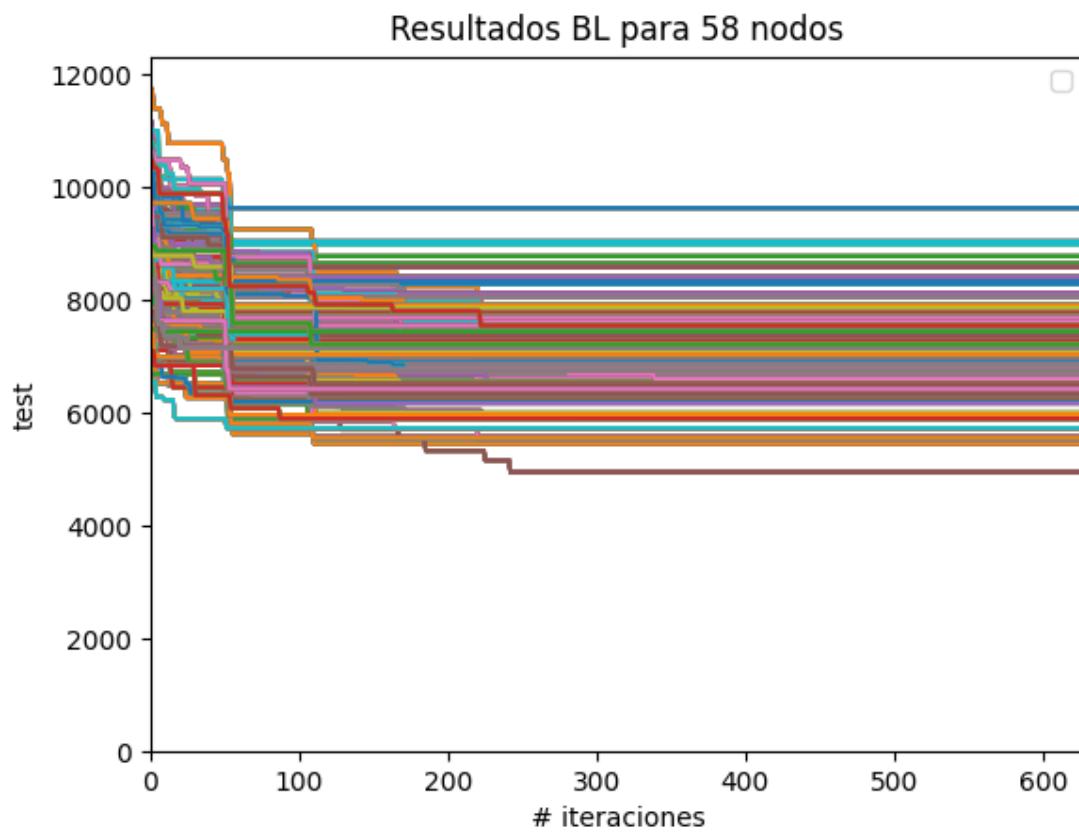
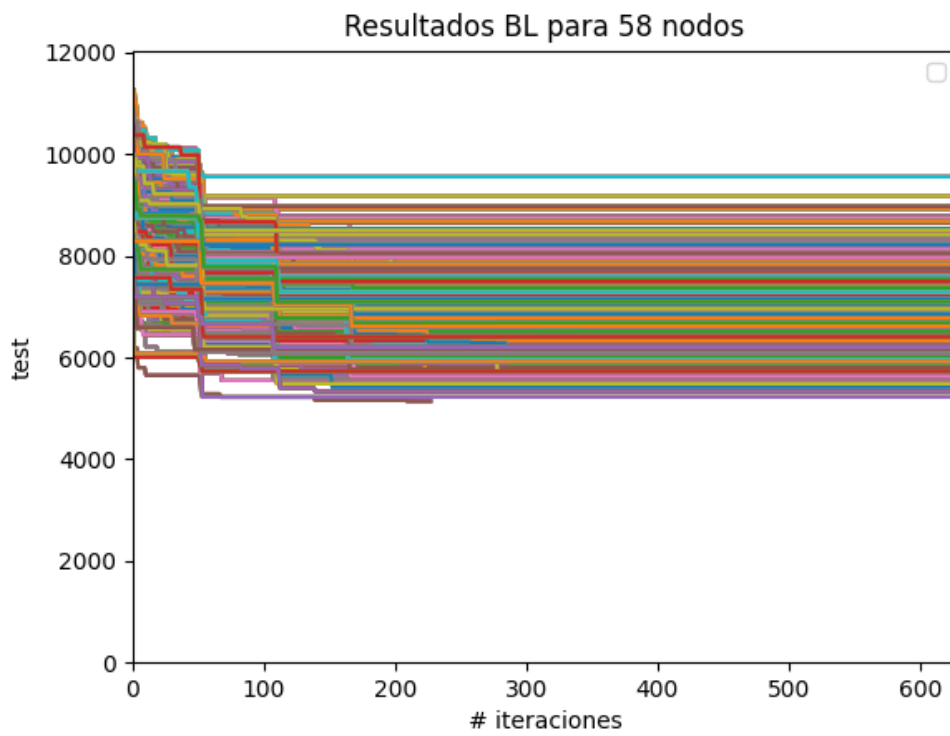
Cálculos auxiliares

Dado que se encuentra una variación de porcentaje entre las distintas instancias creo conveniente a la vez hacer un promedio entre los promedios.

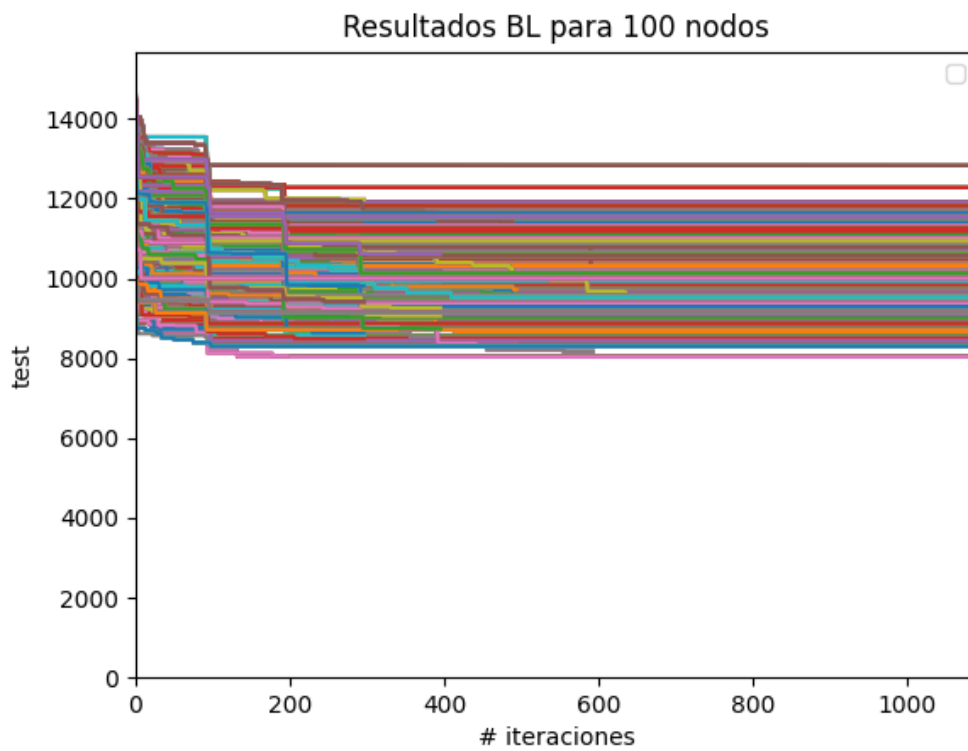
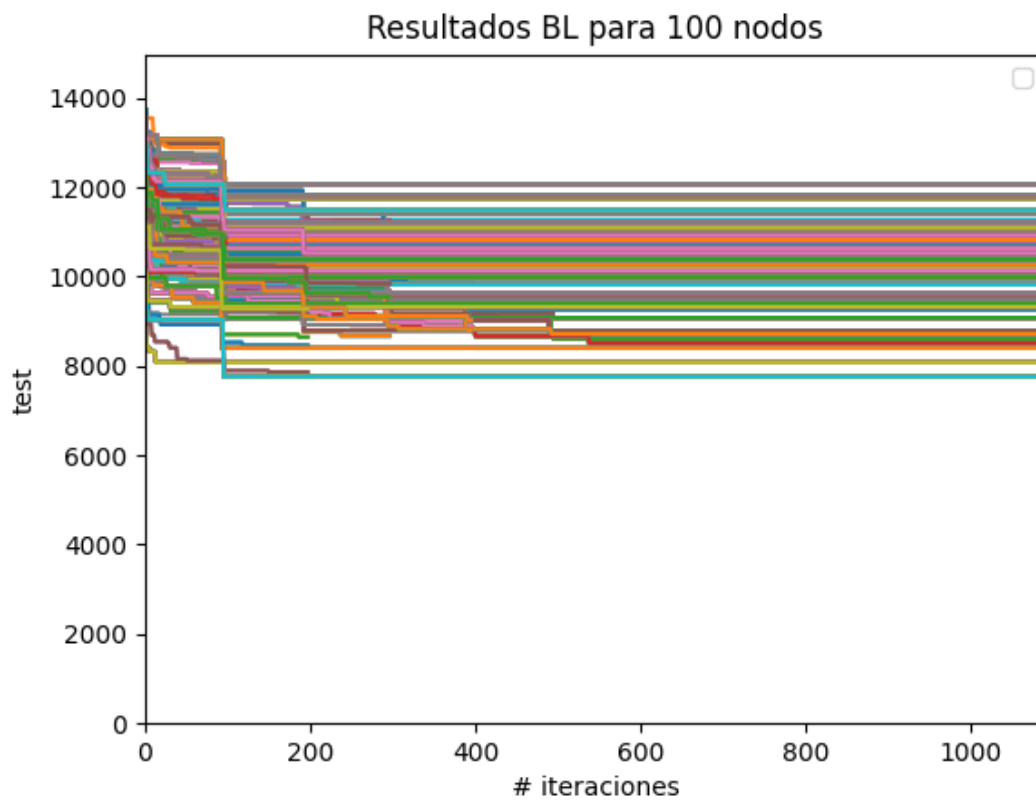
$$(0.37 + 0.97 + 0.59 + 0.85 + 0.86 + 0.55 + 0.46)/7 = 0.66 = 66\% \text{ aproximado}$$

Ejercitación Búsqueda local

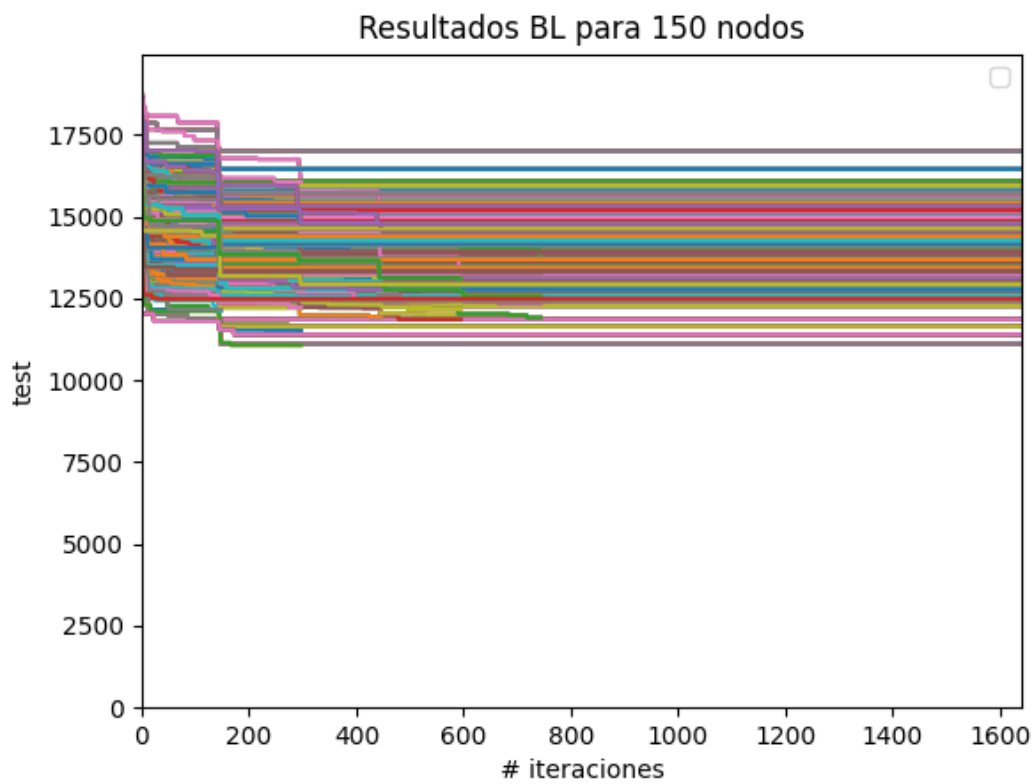
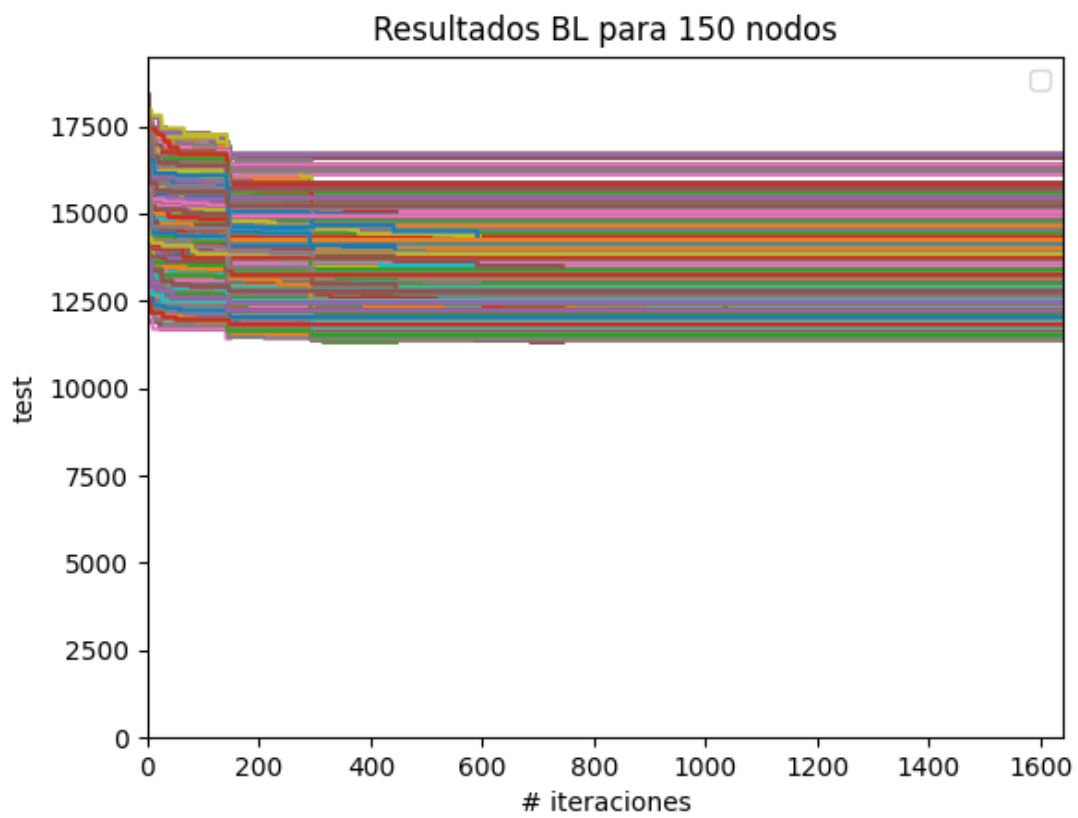
Se repite el proceso mostrando diferentes gráficos de búsqueda locales para la misma cantidad de nodos



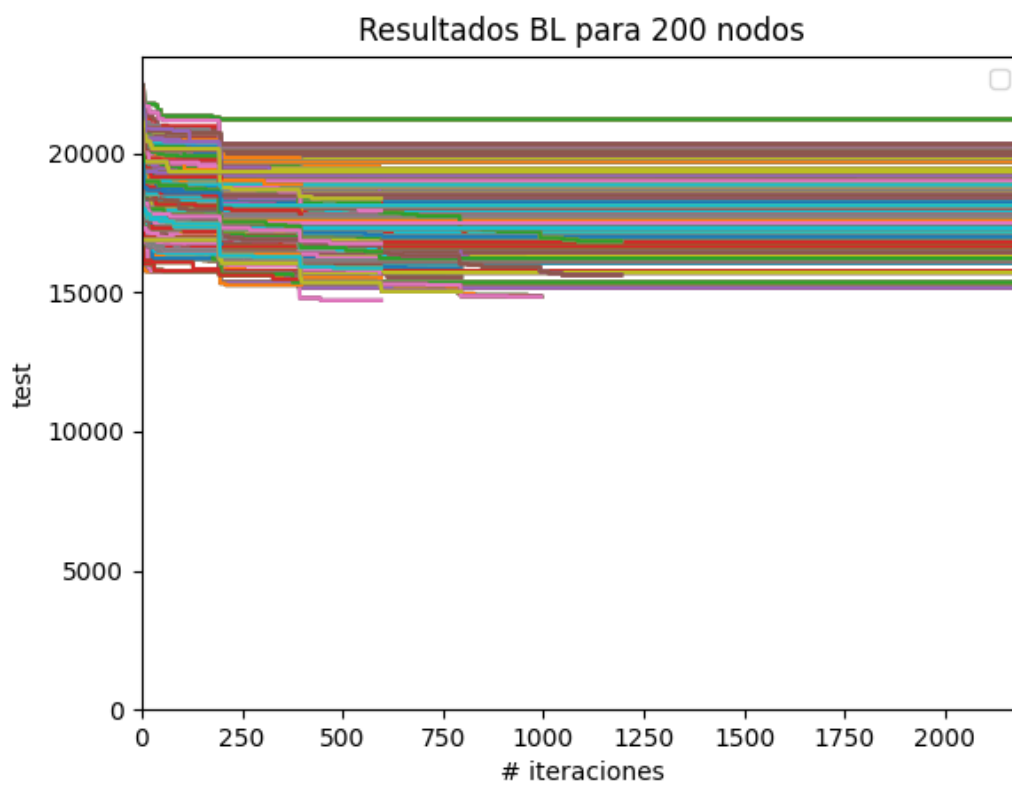
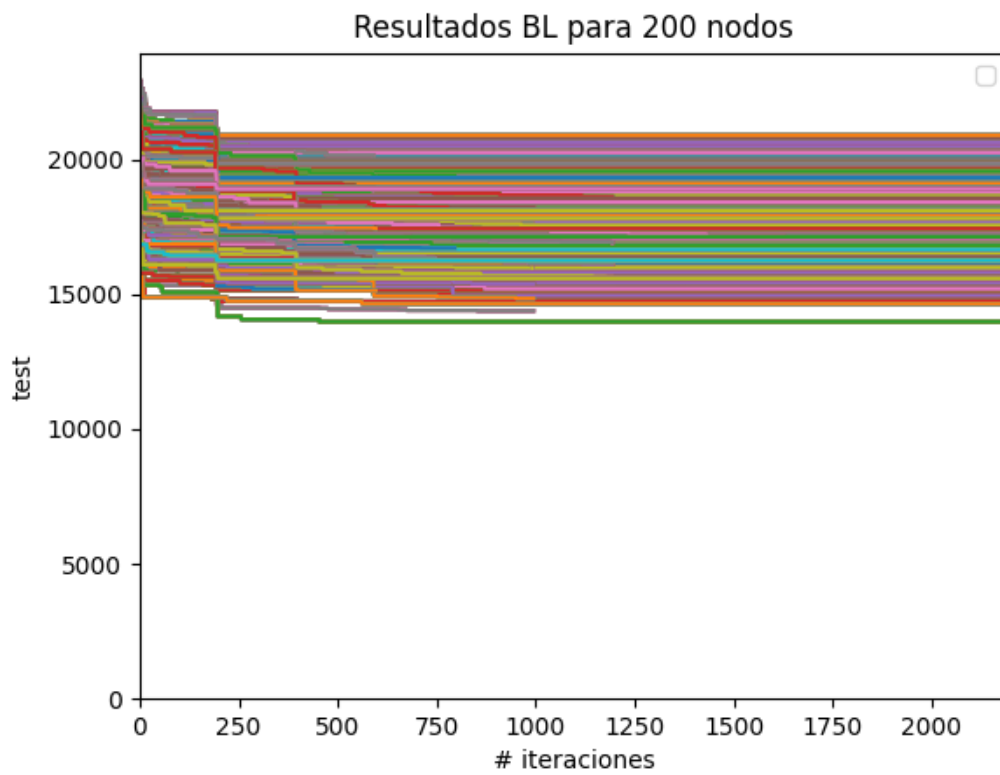
limite corte de mejora 0.9 y 1.2 respectivamente



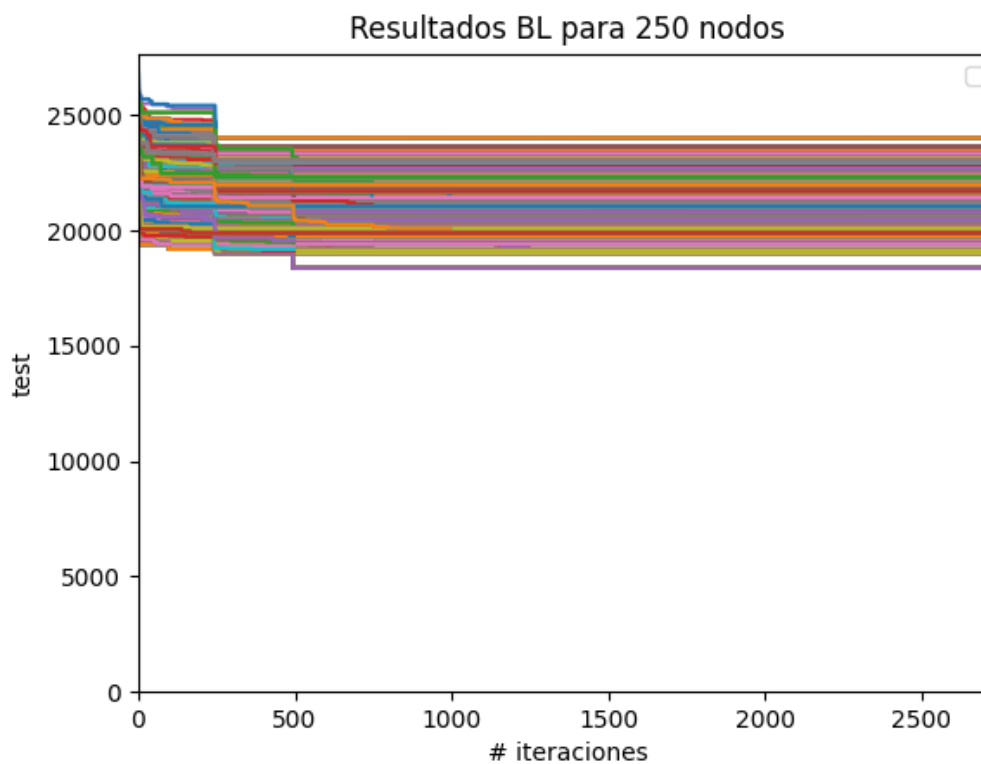
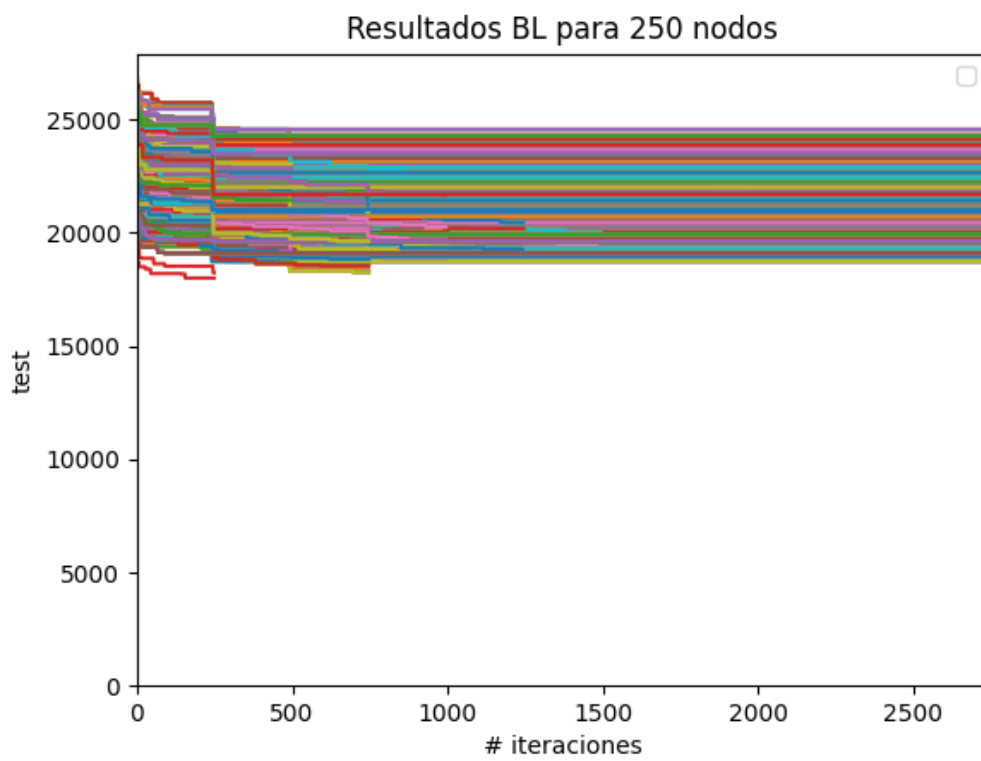
limite corte de mejora 0.9 y 1.2 respectivamente



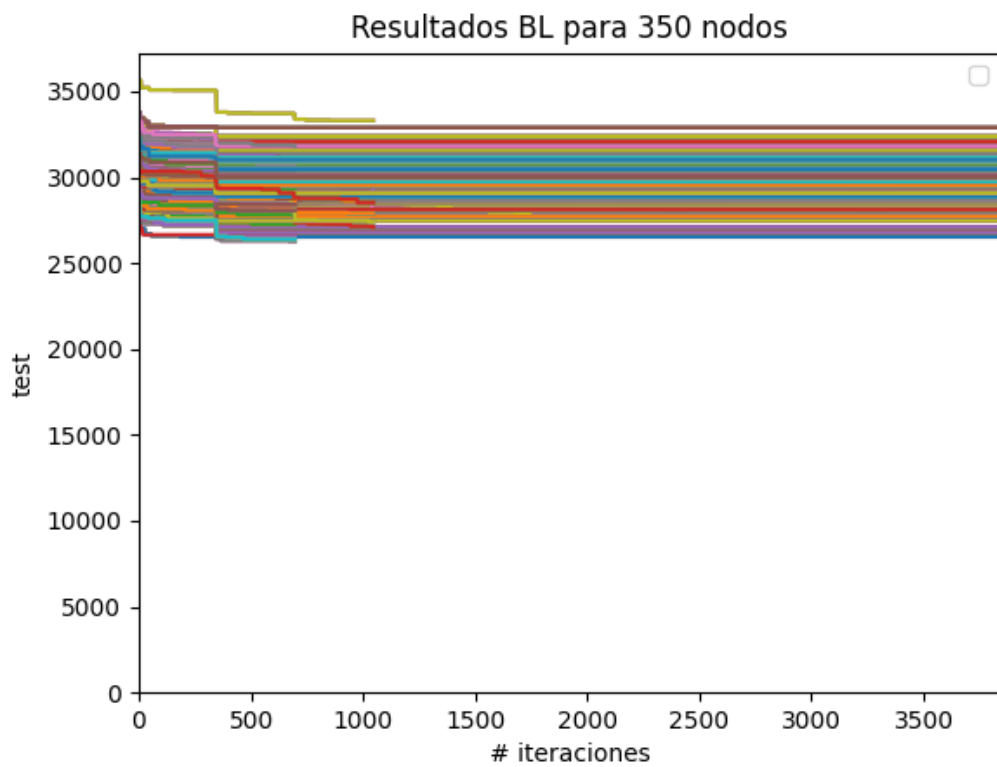
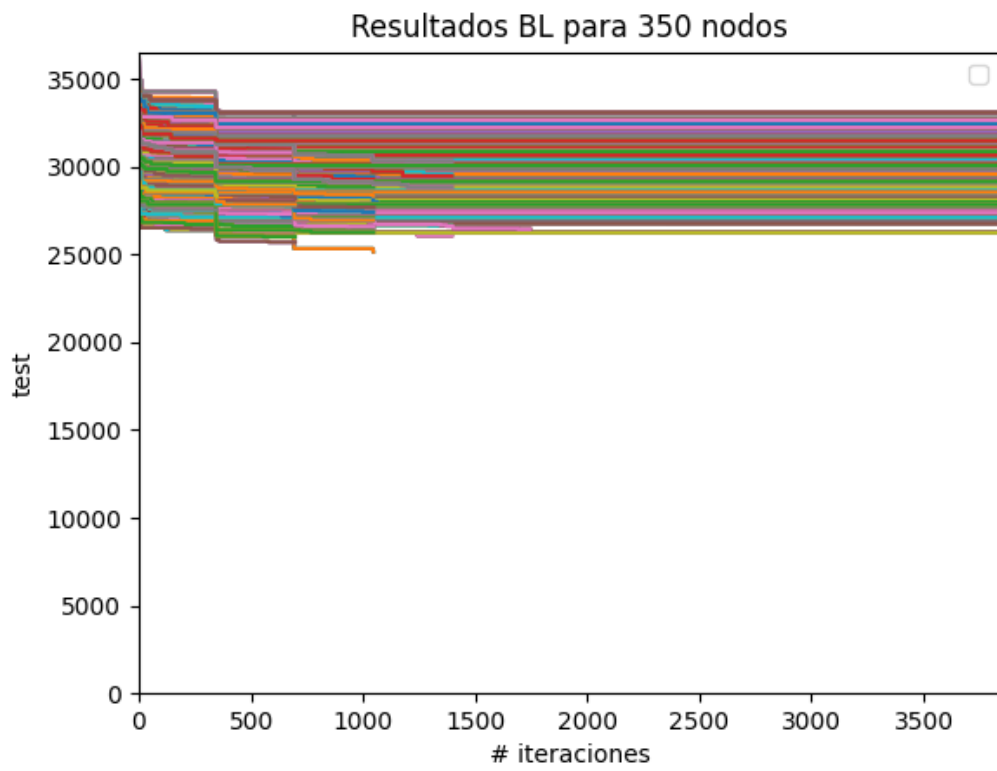
limite corte de mejora 0.9 y 1.2 respectivamente



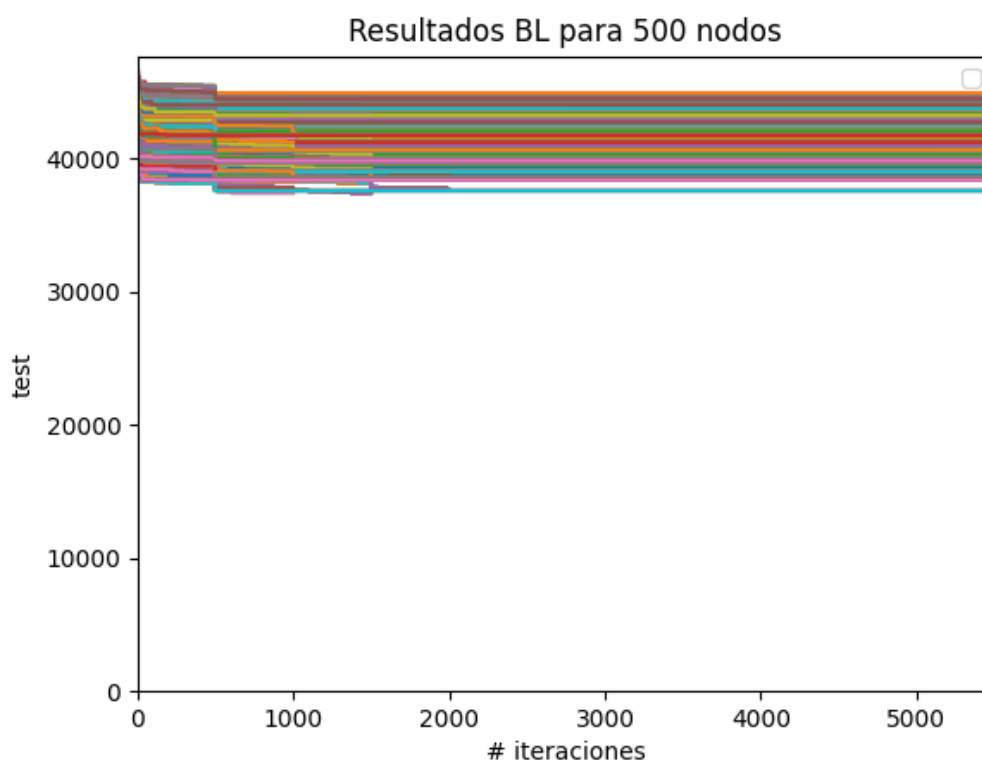
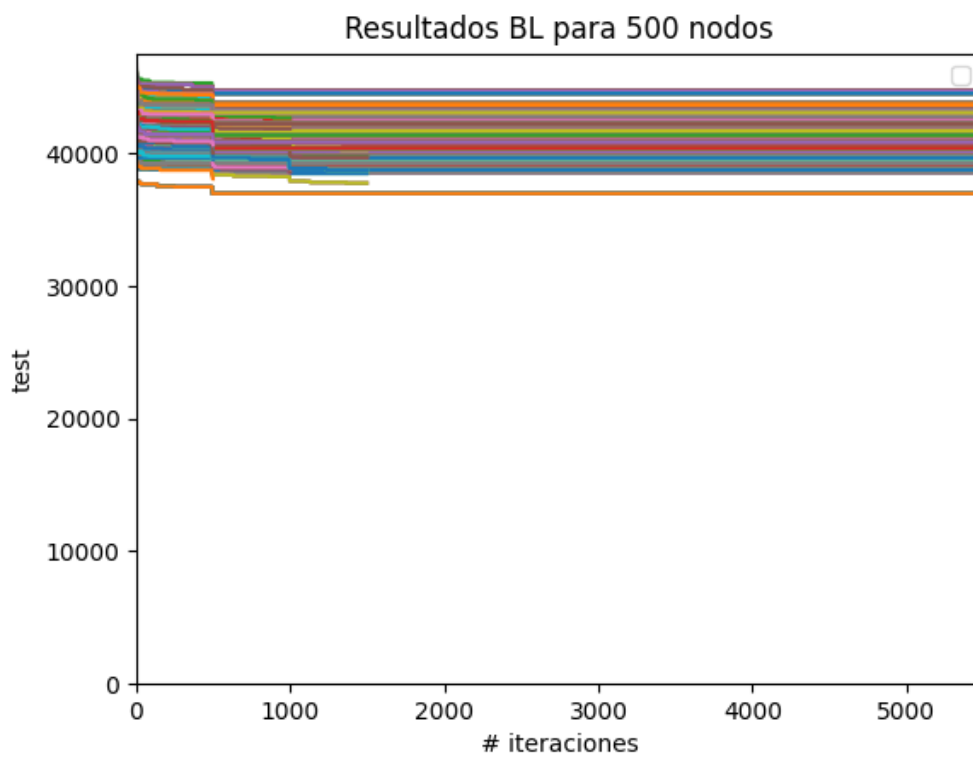
limite corte de mejora 0.9 y 1.2 respectivamente



limite corte de mejora 0.9 y 1.2 respectivamente



limite corte de mejora 0.9 y 1.2 respectivamente



limite corte de mejora 0.9 y 1.2 respectivamente

Se puede observar, si se traza una línea imaginaria, que la cantidad de iteraciones necesarias para encontrar el mínimo en todos los casos para búsqueda local corresponde 1 a 1 a la cantidad de nodos ya que luego de ese punto todas las curvas se aplanan y no logran bajar más.

Conclusiones

Es por eso que gracias a la experimentación y ayudados por los gráficos y cálculos auxiliares podemos visualizar rápidamente como se comporta la metaheurística GRASP y se concluye que un buen criterio aproximado para elegir los cortes de iteraciones y cantidad de búsquedas locales son

GRASP

Del promedio de la experimentación da aproximadamente un 66% de la cantidad de nodos del grafo, es decir respetando el siguiente cálculo

$$\text{iteraciones} = 0.65 * \text{cantidad_nodos}$$

Búsqueda local

Del promedio de la experimentación da aproximadamente un 66% de la cantidad de nodos del grafo, es decir respetando el siguiente cálculo

$$\text{iteraciones} = 1 * \text{cantidad_nodos}$$