

Bono Tablero Reinas

El tablero de reinas se hizo para un tablero de 10000x10000, y se entrenó 5 veces para conocer el número de conflictos que se generaban y las mejores combinaciones de parámetros para el recocido simulado. A continuación se detallan las pruebas realizadas y la curva de evolución de conflictos según el número de iteraciones, debido a la densidad del tablero (10000 x 10000) las ubicaciones no pueden ser claramente visibles, motivo por el cual no se adjuntan en este informe. Para la implementación del recocido simulado propuesto, se modeló la implementación de acuerdo al siguiente pseudo código:

Algoritmo RecocidoSimulado_N Reinas(n, T0, Tmin, α , max_iter):

Entrada:

n ← tamaño del tablero (n x n)
T0 ← temperatura inicial
Tmin ← temperatura mínima
 α ← factor de enfriamiento ($0 < \alpha < 1$)
max_iter ← máximo número de iteraciones sin mejora

Salida:

mejor_estado ← configuración de reinas con menor número de conflictos
mejor_costo ← número de conflictos de mejor_estado
historial_costos ← evolución del costo por iteración

```
1. estado_actual ← generar_tablero_inicial_aleatorio(n)
2. costo_actual ← calcular_conflictos(estado_actual)
3. mejor_estado ← estado_actual
4. mejor_costo ← costo_actual
5. T ← T0
6. iter_sin_mejora ← 0
7. historial_costos ← [costo_actual]
8. i ← 0
9. i_max ← 5000 // Límite adicional de iteraciones totales
10. Mientras T > Tmin y iter_sin_mejora < max_iter y i < i_max hacer:
    11. i ← i + 1
    12. estado_vecino ← generar_vecino_por_swap(estado_actual)
    13. costo_vecino ← calcular_conflictos(estado_vecino)
    14.  $\Delta E$  ← costo_vecino - costo_actual
    15. Si  $\Delta E < 0$  o  $\exp(-\Delta E / T) > \text{número\_aleatorio}(0, 1)$ : ← (Criterio
de aceptación)
        16. estado_actual ← estado_vecino
        17. costo_actual ← costo_vecino
        18. Si costo_actual < mejor_costo:
            19. mejor_estado ← estado_actual
            20. mejor_costo ← costo_actual
            21. iter_sin_mejora ← 0
        22. Sino:
            23. iter_sin_mejora ← iter_sin_mejora + 1
    24. Sino:
        25. iter_sin_mejora ← iter_sin_mejora + 1
```

```
26.  $T \leftarrow T \times \alpha \leftarrow$  (Enfriamiento)
27. Añadir costo_actual a historial_costos
28. Si mejor_costo == 0:
29. Terminar bucle
30. Retornar (mejor_estado, mejor_costo, historial_costos)
```

Test 1:

Entrada:

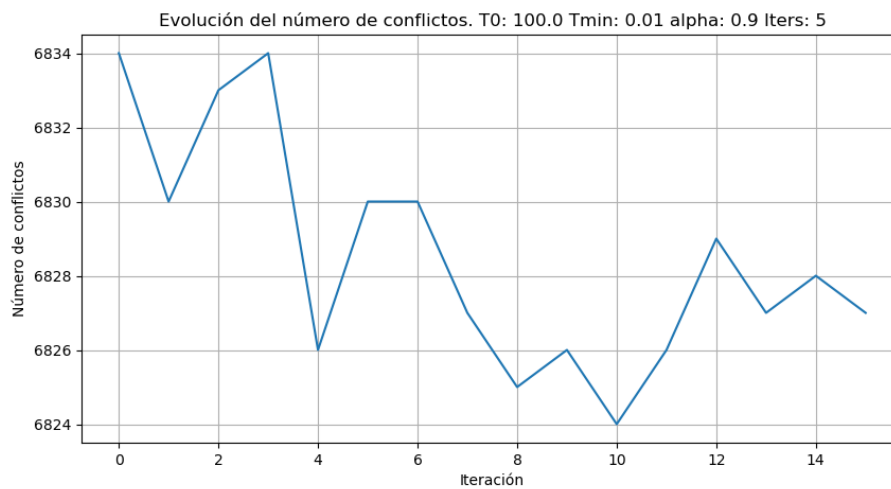
$T_0 = 100$

$T_f = 0.01$

$\alpha = 0.9$

Max Iter¹: 5

Salida: Evaluación de conflictos test 1



Test 2:

Entrada:

$T_0 = 100$

$T_f = 0.01$

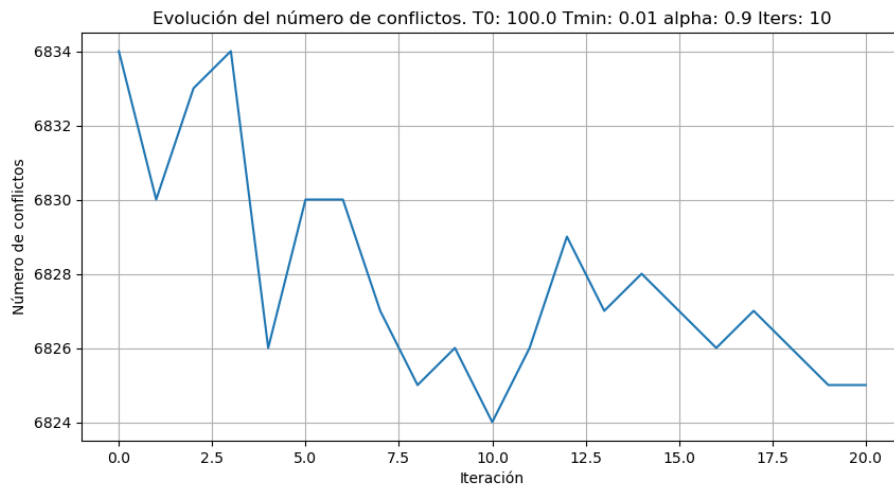
$\alpha = 0.9$

Max Iter²: 10

Salida: Evaluación de conflictos test 2

¹ Estas iteraciones se refiere al número máximo de iteraciones para seguir a la siguiente evaluación si el algoritmo no encuentra ninguna mejora en la solución.

² Estas iteraciones se refiere al número máximo de iteraciones para seguir a la siguiente evaluación si el algoritmo no encuentra ninguna mejora en la solución.



Test 3:

Entrada:

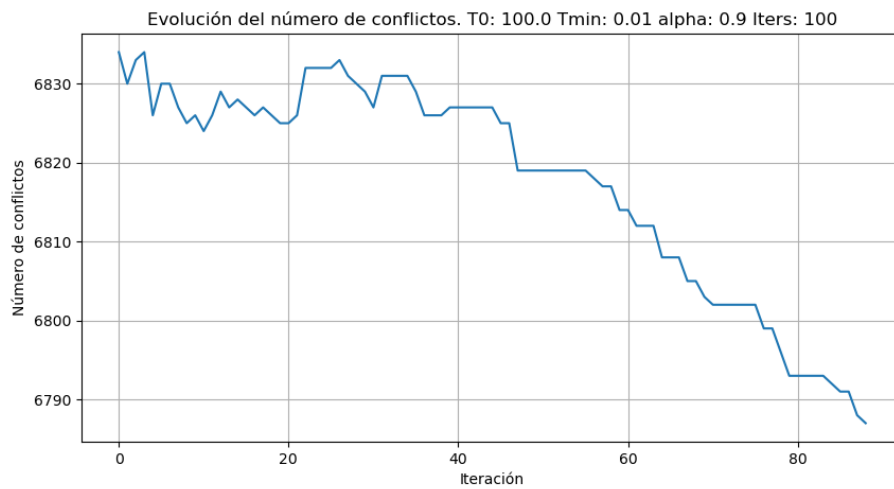
$T_0 = 100$

$T_f = 0.01$

$\alpha = 0.9$

Max Iter³: 100

Salida: Evaluación de conflictos test 3



³ Estas iteraciones se refiere al número máximo de iteraciones para seguir a la siguiente evaluación si el algoritmo no encuentra ninguna mejora en la solución.

Test 4:

Entrada:

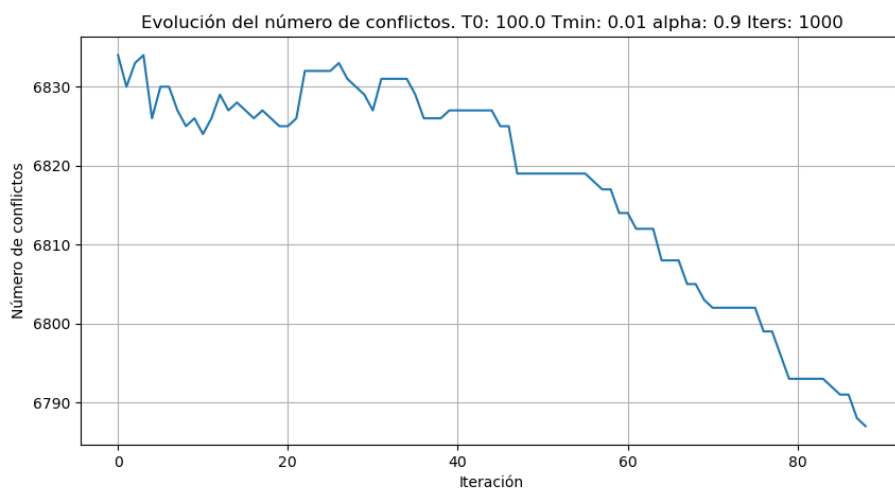
$$T_0 = 100$$

$$T_f = 0.01$$

$$\alpha = 0.9$$

$$\text{Max Iter}^4: 1000$$

Salida: Evaluación de conflictos test 4



Test 5:

Entrada:

$$T_0 = 200$$

$$T_f = 0.01$$

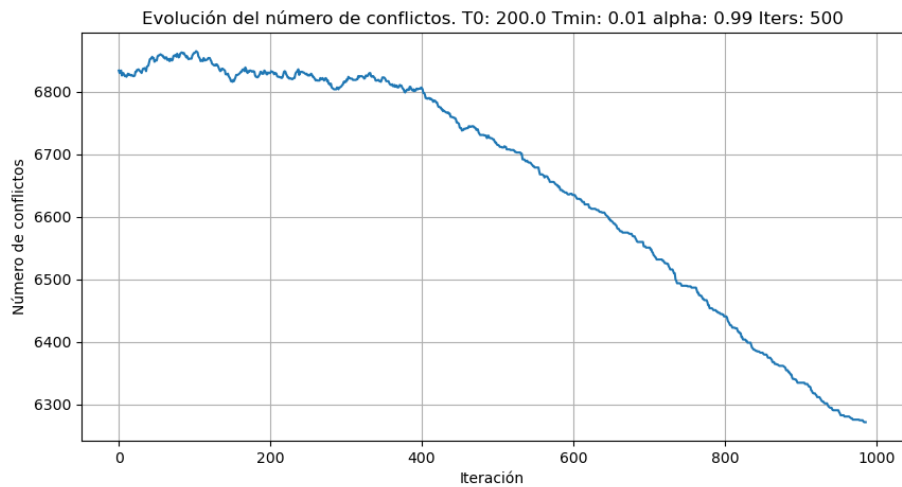
$$\alpha = 0.99$$

$$\text{Max Iter}^5: 500$$

Salida: Evaluación de conflictos test 5

⁴ Estas iteraciones se refiere al número máximo de iteraciones para seguir a la siguiente evaluación si el algoritmo no encuentra ninguna mejora en la solución.

⁵ Estas iteraciones se refiere al número máximo de iteraciones para seguir a la siguiente evaluación si el algoritmo no encuentra ninguna mejora en la solución.



Basado en los resultados obtenidos se observa que para el caso de estudio, el aumento de la temperatura inicial (refiérase al test 1 y test 5 por ejemplo) representa un ajuste importante para mejorar el modelo, es decir, reducir la cantidad de conflictos. Esto se evidencia con la mejora de la resolución de conflictos si la temperatura inicial es mayor. Así mismo, una tasa de enfriamiento que no sea muy agresiva (por ejemplo $\alpha=0.99$) favorece una mejora en el modelo, a comparación de una tasa de enfriamiento con mayor reducción de la temperatura.

Con respecto a la modelación, se observa que aunque el aumento de la temperatura y el menor enfriamiento lleven a un mejor resultado, esto se hace a expensas computacionales, debido a que aumentar los parámetros de temperatura final y tasa de enfriamiento modifican los criterios de parada del algoritmo, por lo que controlar estos parámetros para instancias tan grandes como las evaluadas se hace muy relevante para obtener una respuesta en un tiempo computacional adecuado.