Prueba de Oposición

Agustín Calo

19 de septiembre de 2025

Problema

Valor del problema

Procedimiento

Usando SCIP

Problema a resolver

Ejercicio 9. [SCIP] Determine la mayor cantidad de alfiles que se pueden colocar en un tablero de ajedrez de 8×8, tal que no haya dos alfiles en la misma casilla y cada alfil sea amenazado como máximo por uno de los otros alfiles. Nota: Un alfil amenaza a otro si ambos se encuentran en dos casillas distintas de una misma diagonal. El tablero tiene por diagonales las 2 diagonales principales y las paralelas a ellas.

Figure: Extraído de la guía 1 de Introducción a la Investigación Operativa y Optimización, 2do cuatrimestre 2025

¿Qué nos aporta resolverlo?

En este caso estamos frente a un problema de optimización donde deseamos encontrar un máximo. A simple vista, plantea una situación poco práctica pero de este podemos obtener las siguientes intuiciones:

- Las capacidades del modelado matemático
- Idea de cómo modelar situaciones diversas
- Utilización del algoritmo simplex mediante SCIP

¿Cómo procedemos?

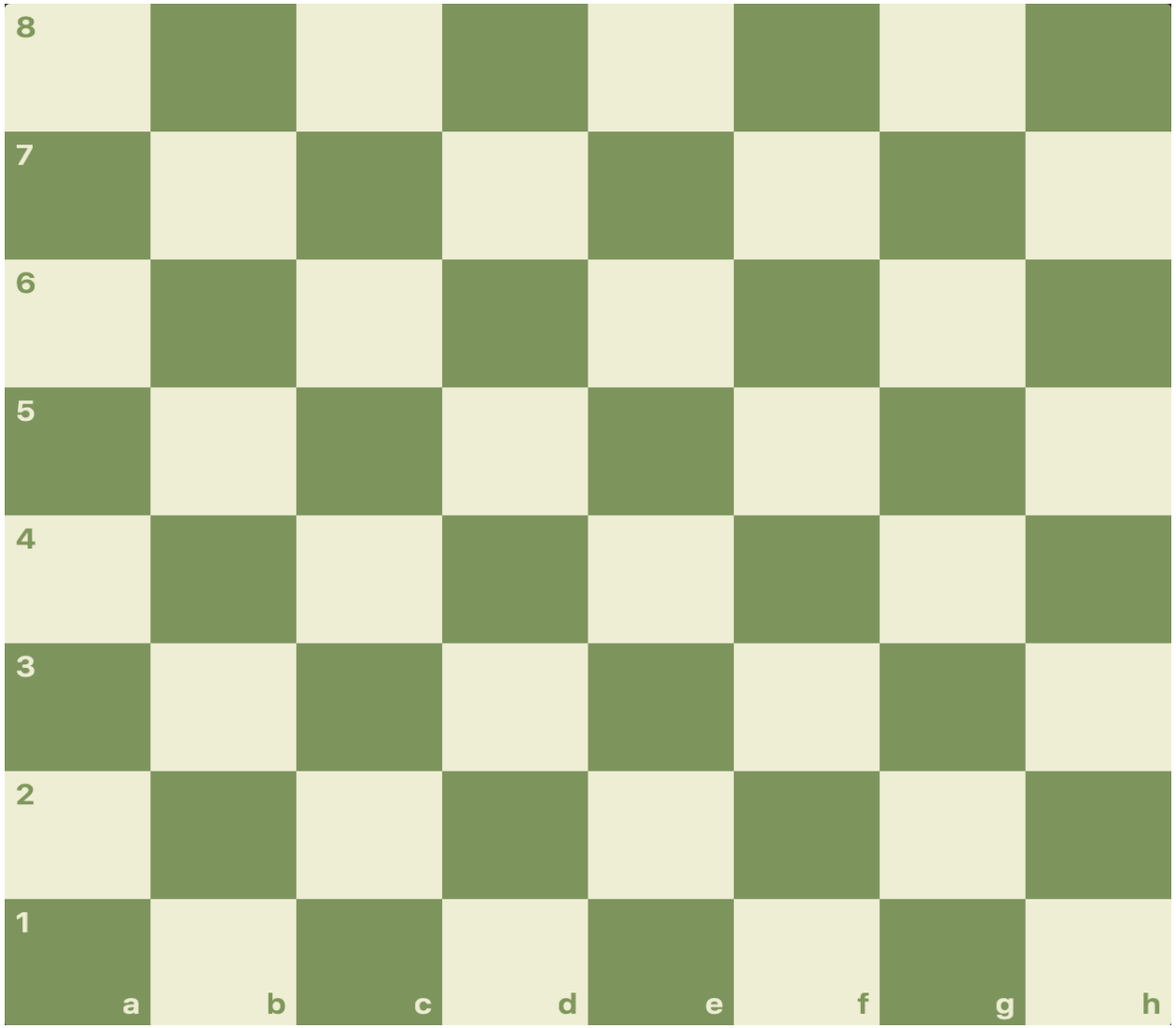
Queremos resolver este problema de manera eficiente. Para esto, hay solvers basados en el algoritmo simplex que lo van a a resolver rápidamente siempre y cuando le demos la información necesaria del problema. Acá es donde entra en juego el modelado matemático.

¿Qué necesitamos para Modelar?

En primera instacia debemos definir como representar a nuestra situación como un conjunto de combinaciones lineales de variables que van a representar algún aspecto del problema.
Por ende, queremos definir:

- Parámetros: valores fijos en el problema.
- Variables: aquellos valores que varian y van a representar la solución.
- Restricciones: combinaciones lineales acotadas por un valor que representan que se puede y que no en el modelo.
- Función objetivo: nos indica que queremos optimizar.

Primer paso: Modelar



Primer paso: Variables

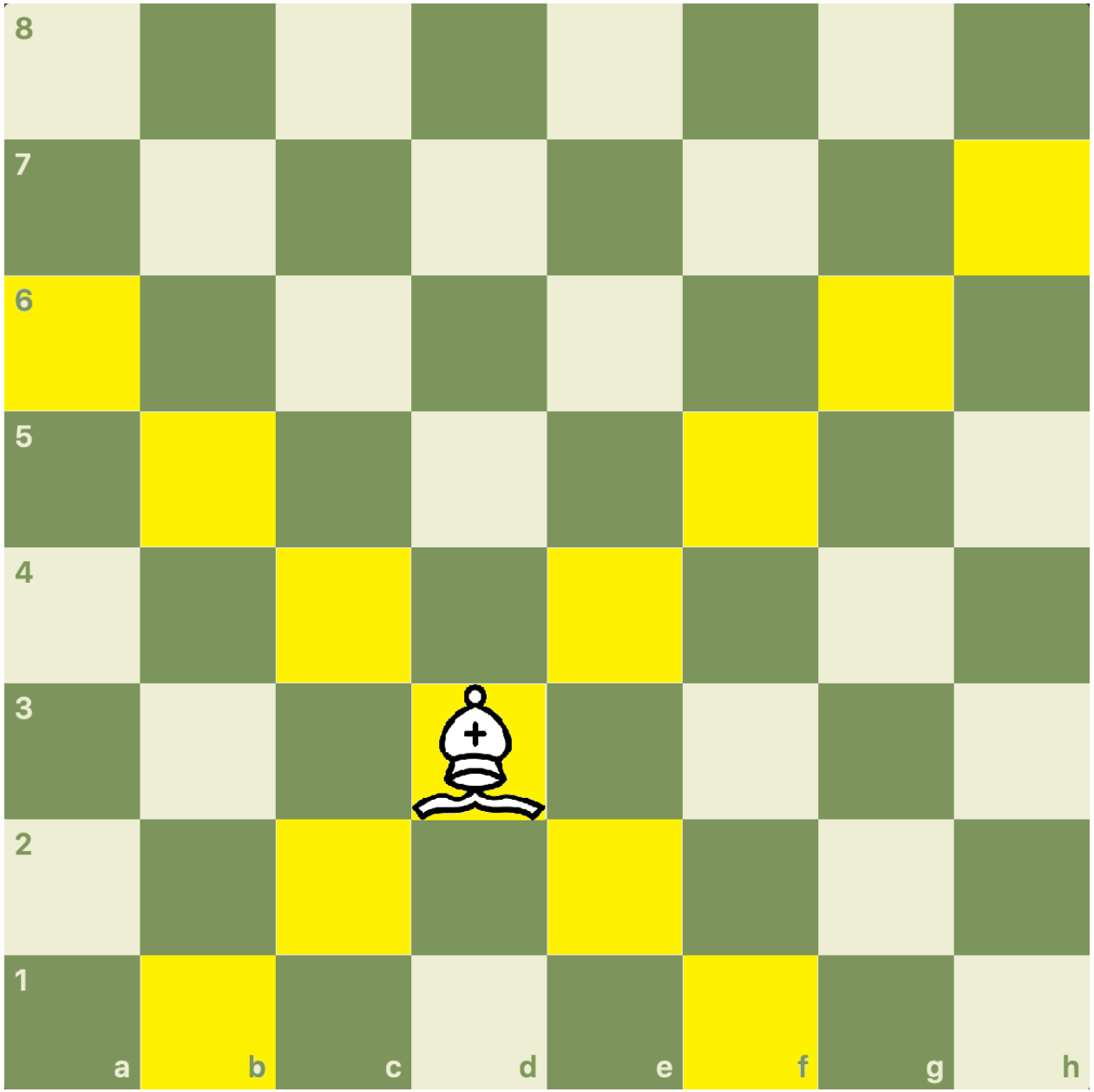
Como podemos apreciar, un tablero es un conjunto de celdas que pueden ser identificadas individualmente por su fila y columna. Siendo que es de 8x8, por simplicidad, vamos a numerar las filas y las columnas del 0 al 7 y denotarlas como tuplas (fila, columna). De esta forma, podemos representar si hay un alfil en la posición (i,j) con un avariable $x_{i,j}$ que vale 1 en caso de que haya y 0 caso contrario.
Como no queremos represntar nada más, podemos decir que tenemos un conjunto de 8x8 variables.

Segundo paso: Restricciones

En esta parte se nos presenta la mayor dificultad, el cómo representar linealmente que un alfil esté amenazado. Para esto nos vamos a valer de las siguientes implicaciones:

1. Necesariamente no puede haber más de dos alfiles por diagonal.
2. Seguro que si hay un alfil en la posición (i,j) entre las dos diagonales que pasan por ahí no pueden contener a más de un alfil entre ambas.

¿Qué es una diagonal en este caso?



¿Qué es una diagonal en este caso?

Se puede apreciar que un alfil en la posción i,j tiene dos diagonales asociadas una que va de izquierda a derecha y otra de derecha a izquierda, las llamaremos diagonal creciente y decreciente respectivamente.
Nos podemos preguntar, ¿hay algún invariante respecto a la posción y los casilleros en las diagonales?
La respuesta es sí y hay varias formas de representarlo. En este caso utilizamos que en la diagonal creciente, siempre se suma o se resta de a uno en ambas cordenadas, es decir, los casilleros son de la forma: $(i \pm k, j \pm k)$. Por otro lado, las diagonales decrecientes las podemos pensar como que se suman uno en una posición y restan uno en otro, lo que se ve como: $(i \pm l, j \mp l)$.

¿Qué es una diagonal en este caso?

Es necesario también preguntarse ¿qué restricciones cumple esos k y l?

Para la diagonal creciente:

1. $0 \leq i - k \leq 7$
2. $0 \leq j - k \leq 7$

Para la decreciente:

1. $0 \leq i - l \leq 7$
2. $0 \leq j + l \leq 7$

Restricciones

Ahora si, planteamos nuestras dos restricciones:
$\forall i, j \in \{0, 1, 2, ...7\}$

1. $x_{i,j} + \sum_k x_{i-k,j-k} + \sum_l x_{i-l,j+l} \leq 4$
2. $\sum_k x_{i-k,j-k} \leq 2$
3. $\sum_l x_{i-l,j+l} \leq 2$

Cerrando el modelo

Ahora, una vez que tenemos las variables que representan a los alfiles del problema y las condiciones que restringen sus posiciones, podemos escribir la función obejtivo:

$$máx \sum_{i,j} x_{i,j}$$

con

$$i, j \in \{0, 1, ..., 7\}$$

Implementar nuestro modelo para obtener la solución

Ahora entra la parte computacional del problema. Para esto vamos a ocupar uno de los solvers por excelencia para problemas de programación lineal, SCIP. Este nos permite a través de sus funciones principales copiar nuestro modelo en la compu y que este se encargue de encontrar la solución usando un algoritmo basado en simplex.

¿Cómo se ve el código?

Ahora entra la parte computacional del problema. Para esto vamos a utilizar uno de los solvers por excelencia para problemas de programación lineal, SCIP. Este nos permite a través de sus funciones principales copiar nuestro modelo en la computadora y que este se encargue de encontrar la solución usando un algoritmo basado en simplex.
Las funciones principales son:

- addVar(): recibe quienes son las variables del probelma.
- addCons(): se encarga de almacenar las restricciones.
- setObjective(): toma la función objetivo.

¿Cómo se ve el código?

```
model9 = Model('Ajedrez')
```

```
x = np.empty((8, 8), dtype='object')

for i in range(0,8):
  for k in range(0,8):
    x[i, k] = model9.addVar(f'x_{i}_{k}', vtype='B')
```

```
for i in range(0,8):
  for j in range(0,8):
    a=max(i-7,j-7)
    b=min(i,j)
    c=max(i-7,-j)
    d=min(i,7-j)
    diagP=quicksum(x[i-k,j-k] for k in range(a,b+1))
    diagN=quicksum(x[i-l,j+l] for l in range(c,d+1))
    model9.addCons(diagN<=2)
    model9.addCons(diagP<=2)
    model9.addCons((diagP+diagN)+x[i,j]<=4)
```

```
model9.setObjective(sum(x[i,j] for i in range (0,8)for j in range (0,8)), sense='maximize')
```

```
model9.redirectOutput()
model9.optimize()
sol = model9.getBestSol()
model9.writeSol(sol, 'Aje.sol')
```

La solución