



UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE CIENCIAS EXACTAS Y NATURALES
DEPARTAMENTO DE COMPUTACIÓN

Complejidad computacional en distintas formulaciones de ajedrez para un jugador

Tesis de Licenciatura en Ciencias de la Computación

Daniel Matías Salvia

Director: Dr. Ariel Arbiser

Buenos Aires, 2024

Índice general

Resumen	III
Abstract	v
1. Introducción	1
1.1. El juego del ajedrez	1
1.1.1. Antecesores	2
1.1.2. Ajedrez por computadora	3
1.1.3. Ajedrez solitario	4
1.2. Grafos	5
1.2.1. Problemas hamiltonianos	5
1.2.2. Complejidad	6
1.2.3. Estado del arte y bibliografía	6
2. Resultados principales	9
2.1. Método 1: Arcos	12
2.1.1. Descripción	12
2.1.2. Tablero básico	14
2.1.3. Tablero con piezas limpiadoras	15
2.1.4. Algoritmo	21
2.2. Método 2: Caminos Diagonales	22
2.2.1. Descripción	22
2.2.2. Tablero básico	23
2.2.3. Tablero con piezas limpiadoras	24
2.2.4. Algoritmo	27
2.2.5. Reducción de las dimensiones del tablero	28
2.3. Método 3: Caminos Rectos	28
2.3.1. Descripción	28
2.3.2. Tablero básico	31
2.3.3. Tablero con piezas limpiadoras	33
2.3.4. Algoritmo	34
2.4. Método 4: Triángulos	37
2.4.1. Descripción	37
2.4.2. Tablero básico	39
2.4.3. Tablero con piezas limpiadoras	40
2.4.4. Algoritmo	41
2.5. Soluciones de las posiciones generadas	43

2.6. Uso de los algoritmos de BFS/DFS	45
3. Variantes de interés	47
3.1. Caballo como PCN	47
3.2. Dama como PCN	48
3.3. Alferza como PCN	50
3.4. Elefante en vez de alfil como PCN	50
3.5. Rey como PCN	52
3.6. Variantes de ajedrez solitario	54
3.6.1. Ajedrez solitario en k movimientos	54
3.6.2. Ajedrez solitario “Captura-Prevención”	56
3.6.3. Otras posibilidades	57
4. Complejidad P en el ajedrez solitario	59
5. Conclusiones	63
A. Información de los métodos	65

Resumen

El objetivo de esta tesis es estudiar la complejidad computacional correspondiente al problema del ajedrez solitario, en que dada una posición el jugador sólo puede hacer jugadas de captura con el objetivo de dejar una sola pieza en el tablero. Demostramos que este problema pertenece a la clase de complejidad de los problemas NP-Completos. Mediante problemas de ciclos hamiltonianos en grafos no dirigidos, se investiga la NP-Compleitud del juego restringiendo el conjunto de piezas a diferentes posibilidades utilizando peones, alfiles y torres. Para esto se presentan cuatro formas de generar las posiciones con diferentes propiedades como las piezas usadas o las maneras de representar las aristas del grafo. Se compara un método con otro para analizar ventajas de cada uno, tales como tamaño del tablero y cantidad de piezas necesarias. También se estudia la NP-Compleitud de distintas variantes en que se modifican algunas reglas, así como los efectos de utilizar piezas del ajedrez antiguo. Se investiga asimismo la frontera P de este problema cuando se usa esencialmente un solo tipo de pieza.

Palabras claves: ajedrez, algoritmo, ciclo hamiltoniano, complejidad, grafo, NP-Completo, pieza.

Abstract

The purpose of this thesis is to study the computational complexity of the Solitaire Chess problem, in which given a chess position the player can only make capturing moves with the goal of leaving a single piece on the board, where captures are the only legal moves. We prove that Solitaire Chess belongs to the class of NP-Complete problems. By using hamiltonian cycle problems on undirected graphs, the NP-Completeness of the game is investigated by restricting the set of pieces to different possibilities using pawns, bishops and rooks. For this purpose, four methods are presented to generate the positions with different properties such as the pieces employed or the ways of mapping each edge of the graph. The methods are compared to illustrate their virtues such as board size and number of pieces required. We also study the NP-Completeness of different variants of the problem in which some rules are modified, as well as the effects of using old chess pieces. The P-frontier of this problem is also investigated when using essentially a single type of piece.

Keywords: algorithm, chess, complexity, graph, hamiltonian cycle, NP-Complete, piece.

Capítulo 1

Introducción

1.1. El juego del ajedrez

El ajedrez “moderno” es un juego de estrategia que se origina aparentemente entre los años 1470 y 1490 en Valencia, España. Se juega en un tablero 8x8 en donde cada fila tiene un número del 1 al 8, las columnas una letra de la A a la H y los casilleros son de color blanco o negro.

El jugador con las piezas blancas tiene sus piezas inicialmente en las filas 1 y 2 y es el que inicia la partida, mientras que el jugador con las piezas negras las tiene en la fila 7 y 8. Los jugadores se alternan haciendo un movimiento cada uno por turno. Cada pieza tiene un patrón diferente de movimiento.

- Alfil: puede moverse a casilleros o capturar piezas que se encuentren en sus mismas diagonales. A diferencia del peón, no se limita a casilleros diagonales inmediatos tanto para capturar o para moverse. Este patrón de movimiento implica que no puede suceder que un alfil que está en un casillero negro pase a estar en un casillero blanco.
- Torre: puede moverse y capturar piezas en línea horizontal o vertical.
- Dama: puede moverse igual que una torre o un alfil. Es considerada la pieza más fuerte del juego.

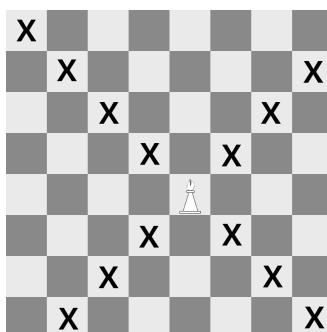


Figura 1.1: Patrón de movimiento del alfil

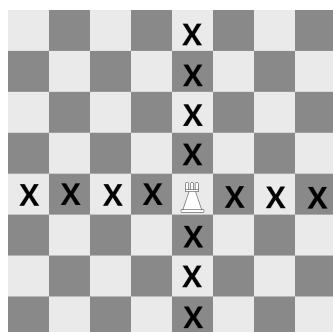


Figura 1.2: Patrón de movimiento de la torre

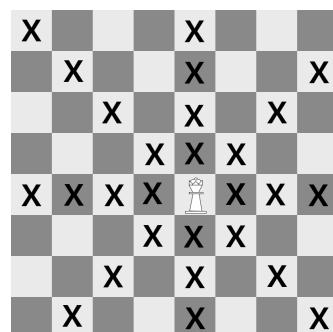


Figura 1.3: Patrón de movimiento de la dama

- Caballo: su movimiento es similar a una “L” en donde se moverá 2 casilleros en el eje vertical u horizontal y luego un casillero en el eje que no se movió inicialmente. A diferencia de las demás piezas, puede “saltear” otras piezas.
- Peón: esta pieza puede moverse un casillero hacia adelante en el eje vertical en cada turno y 2 en el caso de ser su primer movimiento (resaltado con color verde en la Fig. 1.4). Si hay una pieza enemiga en el casillero diagonal inmediato delante de este, puede capturarla (resaltado con color rojo en la Fig. 1.4). Además, si llega a la última fila puede convertirse en una dama, torre, alfil o caballo a elección del jugador (esta acción se la conoce como “promoción” o “coronación”). Por último, si un peón en su primer movimiento avanza 2 casilleros hacia adelante y hay un peón oponente ubicado en la misma fila y en una columna adyacente, el peón oponente puede capturar dicha pieza y moverse una fila atrás en la misma columna de la pieza capturada como próxima jugada. Esta acción se la llama “captura al paso”.
- Rey: puede moverse en cualquier dirección pero sólo a casilleros inmediatos. Esta pieza no puede ser capturada. Se dirá que un jugador está en “jaque” si su rey está siendo amenazado por una pieza del oponente. En ese caso, el jugador en jaque sólo podrá realizar movimientos que produzcan que su rey deje de estar amenazado. Si dichos movimientos no existen, se dirá que se produjo un “jaque mate” [1].

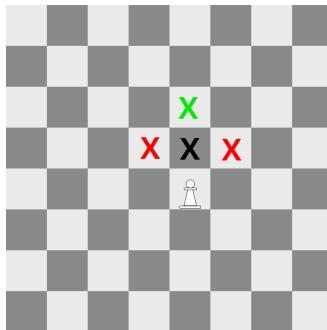


Figura 1.4: Patrón de movi-
miento del peón

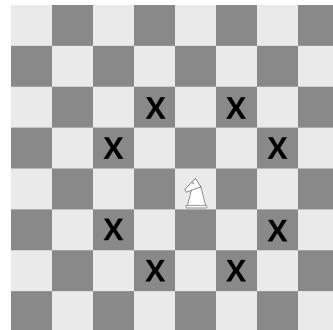


Figura 1.5: Patrón de movi-
miento de la caballo

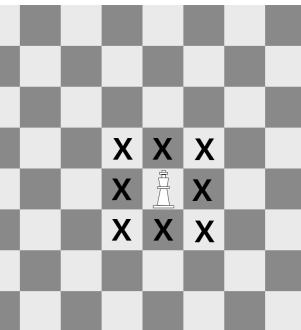


Figura 1.6: Patrón de movi-
miento del rey

Obviaremos algunos conceptos del ajedrez moderno como el “enroque”, la “coronación” o la “captura al paso” o las condiciones de victoria debido a que no fueron necesarias para este trabajo ni tampoco son parte del ajedrez solitario.

1.1.1. Antecesores

Este juego tiene antecesores como el “Chaturanga” originario de la India, donde participaban 4 jugadores y utilizaba menos piezas por jugador que el ajedrez original o el “Shatranj”, siendo este último el antecesor directo del ajedrez. En este último, si bien el tablero y las piezas eran similares, había ciertas diferencias con el movimiento de algunas piezas que cambiaban drásticamente el juego:

- En vez de alfiles había una pieza llamada elefante el cual a pesar de que su patrón de movimiento era similar, ambas se mueven en diagonal, se movía de a 2 casilleros por turno, saltando un casillero por movimiento en vez de tener un alcance ilimitado. Si

bien en el libro “History of chess” de H. J. R. Murray [2] se sugiere que pudo haber tenido dos patrones de movimientos más, nos concentraremos sólo en el mencionado.

- En vez de una dama había una alferza la cual únicamente se podía mover en sentido diagonal y de a un casillero. Esta pieza pasó a ser de la más débil a la más fuerte en el ajedrez moderno.

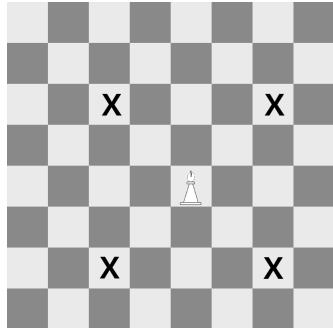


Figura 1.7: Patrón de movimiento del ele- fante

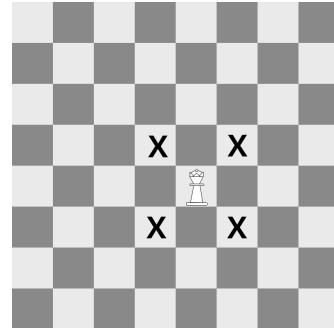


Figura 1.8: Patrón de movimiento de la al- ferza

1.1.2. Ajedrez por computadora

Con la aparición de las computadoras en la década del 50, se comenzó a crear máquinas y programas para jugar ajedrez, sea para entretenimiento o para investigación. El funcionamiento consiste en evaluar diferentes posibles jugadas futuras mediante una función y en base a esto ejecutar la jugada. Los programadores debían (y deben) analizar varios temas:

- Cómo representar el tablero.
- Cómo buscar los posibles movimientos a realizar y seleccionar los mejores.
- Cómo evaluar una posición.

En un artículo escrito por Claude Shannon [3] se llega a la conclusión de que una búsqueda de una buena jugada basada en la “fuerza bruta”, enumerar todas las posibles jugadas y luego evaluarlas, es inviable debido al tiempo que se tardaría en elegir un movimiento utilizando este tipo de búsqueda. En dicho artículo, Shannon propone algunas consideraciones al crear una función de evaluación para las jugadas con la idea de no analizar en profundidad aquellas cuyo valor es bajo y así reducir el tiempo de búsqueda para encontrar la mejor.

A lo largo de los años, diferentes compañías diseñaron programas que corrían en uno o varios procesadores al mismo tiempo, como por ejemplo: Deep Blue [4] o Deep Thought [5] para jugar contra profesionales del ajedrez. Hoy en día se utilizan principalmente como herramientas de análisis y no tanto como oponentes.

1.1.3. Ajedrez solitario

El juego “ajedrez solitario” [6, 7, 8] fue creado por Vesa Timonen. Se trata de una variante del ajedrez tradicional donde sólo hay un jugador, todas las piezas son del mismo color y el tablero puede tener dimensiones variadas, por ejemplo 3×3 , 4×4 u 8×8 entre otros. El juego consiste en descubrir los movimientos a realizar para que sólo quede una pieza en el tablero. En cada movimiento, se debe capturar una pieza. A diferencia del ajedrez tradicional, el rey puede ser capturado en cualquier momento y no hay jaques ni jugadas especiales como el enroque, coronación o captura al paso.

El problema básico de ajedrez solitario es: dada una posición, determinar si es posible una secuencia de capturas como la indicada.

Para determinar la complejidad de este problema, abordaremos el problema de la existencia de un camino o ciclo hamiltoniano en un grafo no dirigido y buscaremos una manera de representar dicho problema en una posición de ajedrez solitario.

A continuación se mostrarán 6 ejemplos de posiciones de ajedrez solitarios en donde 3 tendrán soluciones y los 3 restantes no.

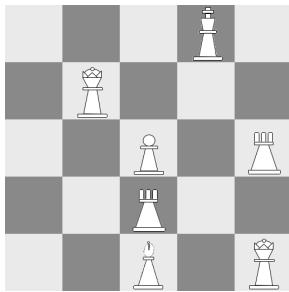


Figura 1.9: Posición de ajedrez solitario sin solución.

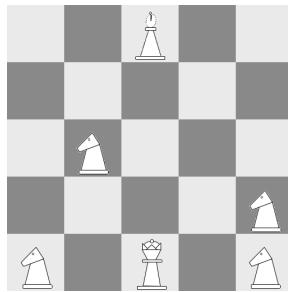


Figura 1.10: Posición de ajedrez solitario sin solución.

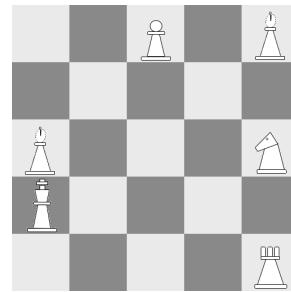


Figura 1.11: Posición de ajedrez solitario sin solución.

La posición descripta por la Fig. 1.9 no tiene solución debido a que ninguna pieza puede capturar al rey ubicado en la casilla d5 ni este tampoco puede capturar una pieza en un movimiento.

En la Fig. 1.10 se puede observar que las únicas piezas que pueden llegar al alfil en el casillero c5 son la dama o el caballo ubicado en el casillero b3. Si alguna de estas piezas no captura al alfil no será capturada por ninguna otra. Al mismo tiempo, si alguna pieza captura al alfil, no tendrá un movimiento válido por lo que esta posición no tiene solución.

Por último, en la Fig. 1.11 la única pieza que puede capturar al rey es la torre cuando esta se ubica en el casillero a3. Sin embargo, para que esto suceda anteriormente tiene que capturar al caballo en la casilla e3 y al alfil de la casilla a3 lo cual dejaría a las piezas en las casillas c5 y e5 sin poder ser capturadas, por lo que esta posición no tiene solución.

La posición de ajedrez solitario vista en la Fig. 1.12 tiene varias soluciones. Una consiste en capturar con el caballo d5 al peón c3 y utilizar la torre a2 para capturar a la torre c2. Luego es trivial que la dama en la casilla b4 puede capturar las piezas restantes.

Una solución consiste en de la posición correspondiente a la Fig. 1.13 consiste en que la torre c5 capture todos los peones. Luego, cuando esté en la casilla a1 el alfil debe capturarla y finalmente este último será capturado por el caballo.

Finalmente, en la Fig. 1.14 una solución consiste en que el alfil capture al peón a3.

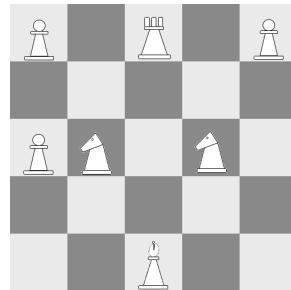
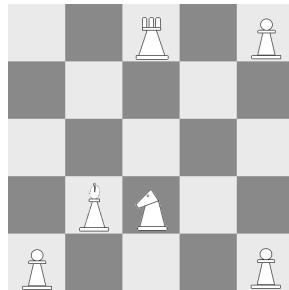
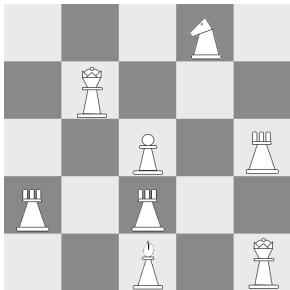


Figura 1.12: Posición de aje- mate solitario con solución. Figura 1.13: Posición de aje- mate solitario con solución. Figura 1.14: Posición de aje- mate solitario con solución.

Luego el caballo b3 capturará al peón a5 y el caballo d3 al peón e5. Por último la torre capturará las piezas restantes, pasando por las casillas e5, a5 y a3.

1.2. Grafos

Un grafo consiste en un par ordenado de conjuntos. El primero contendrá los nodos o vértices mientras que el segundo las aristas o enlaces que los unen. Para una introducción general sobre la teoría de grafos, ver [9], [10] y [11]. Normalmente se los representa con puntos (vértices o nodos) conectados entre ellos por líneas (aristas o enlaces). Para representar una arista se usará la notación $\mathbf{X} \Leftrightarrow \mathbf{Y}$ donde \mathbf{X} e \mathbf{Y} son 2 vértices. Si bien existen grafos no dirigido (dada una arista $\mathbf{X} \Leftrightarrow \mathbf{Y}$ se puede ir de \mathbf{X} a \mathbf{Y} o viceversa) o dirigidos (dada una arista, este indica si se puede ir de \mathbf{X} a \mathbf{Y} o \mathbf{Y} a \mathbf{X} pero no ambas al menos que exista otra) en este trabajo sólo nos limitaremos a los grafos no dirigidos.

Se dice que un nodo tiene grado g si hay g aristas incidentes al vértice. Al mismo tiempo, un grafo es regular si todos los vértices de este tienen el mismo grado y también se dice que es completo si todos los vértices de un grafo están unidos entre ellos. Diremos que un grafo es **casi 3-regular** si todos los nodos son de grado 3 excepto por uno que es de grado 2.

Existen numerosos problemas sobre grafos, ver [9], [10] y [11]. En este trabajo nos concentraremos en el problema de la existencia de un camino hamiltoniano (HP) o ciclo hamiltoniano (HC).

1.2.1. Problemas hamiltonianos

El problema del camino hamiltoniano (HP) sobre un grafo consiste en determinar si existe un camino que pase por todos los nodos sin repetir ninguno. El problema del ciclo hamiltoniano (HC) sobre un grafo consiste en determinar si existe un camino que pase por todos los nodos del grafo, sin repetir ninguno excepto el primero que será igual al último.

Ambos problemas están relacionados:

- Por un lado, dado un grafo G del cual se quiere encontrar un HP, se puede buscar un HC en un grafo H que consiste en el grafo G y un nodo extra que esté conectado a todos los nodos de este.

- Por otro lado, dado un grafo con un HC, al remover una arista de este ciclo obtendremos un HP ya que podríamos tomar como primer y último nodo los nodos incidentes en la arista removida.

1.2.2. Complejidad

Los problemas de determinación de la existencia de algún HP o HC pertenecen a la categoría de problemas NP-Complejos [12]. Este es el subconjunto de los problemas de decisión pertenecientes a la clase de complejidad NP en donde todo problema en NP puede ser reducido a cada problema en NP-Completo. La reducción consiste en vincular dos problemas en donde la existencia de un algoritmo para solucionar uno implica la existencia de otro algoritmo para resolver el segundo en un tiempo polinomial.

Un problema es NP-Completo (Cook-NP-Completo) si se cumplen dos condiciones:

- Puede ser resuelto en un tiempo polinomial por una máquina de Turing no determinista. Es decir, pertenece al conjunto NP. Equivalentemente, dada una solución, esta puede ser verificada en tiempo polinomial por una máquina de Turing determinista.
- Todo problema perteneciente a NP puede ser reducido a este en tiempo polinomial.

Al mismo tiempo, para demostrar que un problema es NP-Hard sólo hay que verificar la segunda condición.

Existen diversos problemas NP-Complejos como por ejemplo:

- El problema de satisfacibilidad booleana (SAT) en donde dada una expresión booleana con variables y sin cuantificadores, existe una asignación de valores para las variables en donde la expresión es verdadera
- El problema de la mochila (Knapsack) en donde dados una mochila con un peso máximo posible y un conjunto de objetos, cada uno con su valor y peso, se busca un subconjunto de objetos que no excedan el máximo de la mochila y tenga el mayor valor posible.
- El problema de la cobertura de vértices (Vertex Cover) en el que, dado un grafo, se busca un conjunto de vértices de cardinal mínimo tal que cada una de las aristas del grafo incida en al menos un vértice del conjunto.

La manera más sencilla para demostrar que un problema está en NP-Completo es primero demostrar que el problema pertenece al conjunto de los problemas NP y luego transformar un problema que ya este en NP-Completo a este en tiempo polinomial.

1.2.3. Estado del arte y bibliografía

Hemos comprobado que paralela e independientemente al desarrollo de esta tesis, otros autores han encarado el mismo problema de complejidad en distintas variantes del ajedrez solitario, de maneras diversas. En [14] se prueba la NP-Complejidad de una variante en la que cada pieza tiene un máximo de 2 capturas permitidas, utilizando reducciones desde problemas distintos a los tratados aquí. Posteriormente, en [15] se prueba una variedad de resultados concernientes a problemas NP-Hard del ajedrez solitario procurando el uso de una cantidad mínima de tipos de pieza. Estos autores, para varios de los casos, reducen

problemas NP-Completos sobre grid graphs, esto es grafos en que los nodos que forman una grilla con ejes dados por las adyacencias horizontales y verticales (se suelen representar como puntos del conjunto $\mathbf{Z} \times \mathbf{Z}$), de alguna manera con geometría más “cercana” a lo deseado. Son más variantes, pero para varias parten esencialmente de formulaciones propias de un tablero. Nuestras opciones son reducir problemas más “distantes”, que involucran grafos casi 3-regulares y que a priori no se representan dentro de grid graphs.

En [14] se analiza la variante de ajedrez solitario con sólo torres y donde cada una pueda hacer como máximo 2 capturas y concluye que dicha variante es NP-Completa. Nosotros analizamos dicha variante en la sección 4 (junto a la variante de sólo caballos) pero permitiendo que cada pieza pueda realizar una cantidad ilimitada de capturas y demostramos que dicha variante es polinomial. También, en ambos casos se analiza una variante con sólo peones pero con la misma diferencia en la condición.

Muy especialmente, a lo largo de [15] se utiliza la reducción de problemas de HP, a diferencia de nosotros que utilizamos desde el comienzo HC en una sola formulación. También se prueba que es NP-Hard la variante con peones y una sola torre, para lo cual reducen un problema de NP-completitud del HP sobre grafos con grado acotado por 3 que requiere además comenzar y terminar en nodos de grado 1. Si bien el método es bastante distinto de los que mostramos, la construcción asemeja la del método Arcos (sección 2.1), pero no distinguen peones nodos de peones articuladores, y con cierta complicación sutil del algoritmo preciso. Por otra parte, la adaptación de una prueba de NP-Hard sobre esta familia de grafos termina siendo más complicada, tomando una reducción presentada por otros autores.

También en [15] se utiliza la reducción de problemas de una variante de 3-SAT (3-SAT monótono planar), al que se le agrega la condición de que los literales ocurran como máximo dos veces, para demostrar que algunas variantes de ajedrez solitario con dos tipos de piezas es NP-Hard. Sin embargo, sólo se utiliza la reducción de problemas de HP para demostrar que el ajedrez solitario es NP-Completo para cualquier conjunto de dos piezas distintas. Por último, las pruebas [15] no incluyen detalles o algoritmos detallados, mostrando en su lugar guías y bosquejos.

Nuestra presentación parte de una motivación previa de representar grafos determinados en posiciones de ajedrez con varias piezas. Analizamos también casos de un solo tipo de pieza y algunos con 2 restringidos (sección 4). En [15] también incluyen casos de una pieza y se aborda la variante en que de cierta pieza haya una sola en el tablero y que deba ser la que sobreviva al final de las capturas, y lo mismo con dos tipos de pieza donde uno de estos aparece una sola vez y debe ser la que sobreviva (en [7] y [8] aparece esto en problemas que incluyen un rey). Finalmente, estos autores no tratan formulaciones para dos jugadores, como “Captura-Prevención” (sección 3.6.2), ni otras con límites en la cantidad de movimientos.

Capítulo 2

Resultados principales

El objetivo central de esta tesis es buscar distintas maneras de representar un grafo mediante un tablero de $N \times M$, utilizando diferentes piezas de ajedrez, ya sean de la actualidad o históricas, de modo que las posibles secuencias en las que se capturan las piezas, ubicadas en casillas que denotan los nodos del grafo, representen los distintos caminos o ciclos hamiltonianos de dicho grafo. De esta manera se busca demostrar que el problema del ajedrez solitario pertenece a la categoría de problemas NP-Completos bajo la condición que las piezas que están presentes en la posición del ajedrez solitario sean únicamente alfiles, peones y torres, o bien otro subconjunto de piezas específico preferentemente minimal, y luego utilizar dicha demostración para probar que algunas de sus variantes son NP-Completos o NP-Hard.

A lo largo de las construcciones que veremos, vamos a utilizar 4 tipos de piezas que cumplirán distintos roles. A saber:

- **Peones nodos (PN):** utilizadas para representar los nodos del grafo. Generalmente estos peones estarán ubicados de una manera en la que no puedan capturar ninguna pieza ,es decir, siempre estarán en la misma casilla hasta ser capturados.
- **Pieza captura nodos (PCN):** esta pieza capturará las diferentes piezas nodos. Esta pieza será un alfil o una torre, según el método a usar.
- **Peones articuladores (PA):** utilizadas para representar aristas. Estas piezas tienen un alcance de movimiento reducido ya que no queremos que puedan capturar otro tipo de piezas como piezas captura nodos.
- **Piezas limpiadoras (PL):** utilizadas para capturar las piezas articuladoras restantes una vez que se hayan capturado todos los PN. Es necesario que si esta pieza está en algún casillero en que estuvo algún PA no tenga un movimiento que le permita capturar un PN.

Una primera intuición podría ser que una PCN con un alto alcance de movimiento puede ser beneficioso al crear una posición en base a un grafo. Sin embargo esto es erróneo dado que impondría más limitaciones. Por ejemplo en el caso de utilizar una torre como PCN, tenemos que asegurarnos de que un PA correspondiente a una arista X que va de $P1$ a $P2$ (**X: $P1 \Leftrightarrow P2$**) no esté en la misma fila o columna que un PA correspondiente a una arista **Y: $P3 \Leftrightarrow P4$** ya que permitiría moverse de un PN incidente en la primera arista a otro PN incidente en la segunda arista los cuales pueden no ser adyacentes en el

grafo. Si reemplazamos esta torre por una dama, no sólo tendríamos que asegurarnos de que no esté en una misma fila o columna sino también que no comparta una diagonal, lo cual limitaría aún más los casilleros en los que podríamos ubicar otras piezas.

Si la única pieza que captura las piezas nodos es la PCN, entonces el orden en el que capture las piezas está dado por el HP. Esto se debe a que si la representación es correcta, la transición de un PN a otro se corresponde con un camino en el grafo. Por lo tanto, el recorrido o camino que incluye a todos los PN utilizando únicamente el movimiento de la PCN es igual a un recorrido que visite todo los nodos del grafo, es decir un HP. Luego, se necesitarán más movimientos para eliminar todas las piezas restantes de la posición.

Para hablar de las posiciones de las piezas, se indicarán las coordenadas x e y siendo x la columna e y la fila, notando la casilla como (x, y) . Al mismo tiempo, llamaremos a las casillas de color blanco a toda casilla para la cual $x + y$ es par y casillas de color negro toda casilla tal que $x + y$ es impar.

También, llamaremos diagonal principal y diagonal secundaria de una pieza P a las diagonales compuestas por las casillas $(x_p + a, y_p + a)$ y $(x_p + a, y_p - a)$ con $a \in \mathbf{Z}$ respectivamente. Nótese que cuando $a = 0$ la casilla resultante es aquella donde está la pieza P .

Por último, el término n denotará el número de nodos pertenecientes al grafo y m el número de aristas de este.

Para demostrar que el problema del ajedrez solitario pertenece a la categoría de problemas NP-Complejos el camino usual es reducir un problema de dicha categoría a este. Inicialmente se dará la definición de grafo casi 3-regular y luego se demostrará que el problema de la existencia de un HC en esta familia de grafos es NP-Completo.

Definición 2.0.1 (Grafo casi 3-regular). Un grafo no dirigido $G = (V, E)$ es casi 3-regular $\iff \exists x \in V$ tal que $gr(x) = 2 \wedge \forall y \in V, y \neq x, gr(y) = 3$

En otras palabras, diremos que un grafo es casi 3-regular si todos los nodos del grafo son de grado 3 con la excepción de uno de grado 2. Al mismo tiempo, la cantidad de aristas que tiene un grafo casi 3 regular es de $7 + 3(n - 5)/2$ con $n \geq 5$ e impar por el lema del apretón de manos [9], esto es, dado un grafo sin bucles ni dirigido, la suma de los grados de los vértices de este es igual al doble de la cantidad de aristas.

Lema 2.0.1. No existen grafos casi 3-regulares con una cantidad par de nodos.

Demostración. Sea $G = (V, E)$ un grafo casi 3-regular y $|E|$ la cantidad de aristas en el grafo, por el lema del apretón de manos la suma de los grados de un grafo sin bucles y no dirigido es igual a $2|E|$. Al mismo tiempo, la suma de los grados de un grafo casi 3-regular es de $3(n - 1) + 2$ por lo cual $3(n - 1) + 2 = 2|E|$. Cuando n es par la suma de los grados es impar, entonces no existe $|E| \in \mathbb{N}$ tal que se satisfaga la igualdad. \square

Lema 2.0.2. El problema de la existencia de un HC en un grafo G 3-regular es NP-Completo.

Demostración. Ver [13]. \square

Lema 2.0.3. Dado un grafo 3-regular G , y una arista e de G , el problema de la existencia de un HC en G que incluya a la arista e es NP-Completo.

Demostración. Reduciremos el problema anterior a este. Sea $G = (V, E)$ un grafo 3-regular, construiremos G' reemplazando un nodo por el subgrafo de la Fig. 2.2. Luego, G tiene un HC si y sólo si G' también lo tiene ya que se puede entrar y salir del subgrafo de la misma manera que en el nodo reemplazado en G y recorriendo todos los nodos agregados. Es más, si G' tiene un HC, dada cualquier arista de color verde, existe un HC en G' que recorre dicha arista. Esto nos da una reducción polinomial del problema de la existencia de un HC en un grafo 3-regular al problema mencionado.

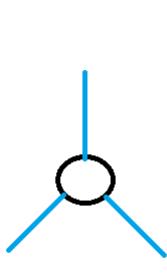


Figura 2.1: Nodo arbitrario

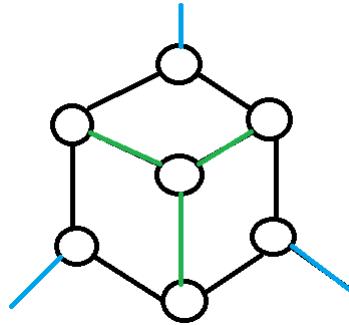


Figura 2.2: Subgrafo utilizado para reemplazar el nodo

Falta demostrar que el problema del enunciado pertenece a la categoría de problemas NP. Dada una secuencia de nodos, se puede verificar la existencia de una arista entre 2 nodos de manera polinomial según la forma de representar el grafo. Luego, la secuencia de nodos contendrá $n + 1$ elementos, por lo que verificar si una secuencia de nodos es un HC es polinomial. Al mismo tiempo, verificar si la secuencia de nodos contiene la arista e es lineal en función de la cantidad de nodos. Como el problema del enunciado pertenece a la categoría de problemas NP y redujimos un problema NP-Completo a este, entonces el problema del enunciado es NP-Completo.

□

Lema 2.0.4. El problema de la existencia de un HC en un grafo G casi 3-regular es NP-Completo.

Demostración. Sea $G = (V, E)$ un grafo 3-regular y $e = (x_1, x_2) \in E$. Sea $x \notin V$, construimos un grafo $G' = (V', E')$ con $V' = V \cup \{x\}$, $E' = (E - \{(x_1, x_2)\}) \cup \{(x, x_1), (x, x_2)\}$. G' es casi 3-regular ya que $gr(x) = 2$ mientras que $gr(x_1) = gr(x_2) = 3$. Verificamos la existencia de un HC para G' .

- Si existe un HC, entonces existe un HC en G que incluye a e . Esto se debe a que como $gr(x) = 2$ los ejes (x, x_1) y (x, x_2) deben estar en tal HC.
- Si no existe un HC, entonces no existe en G un HC que incluya a e .

Esto nos da una reducción polinomial del problema de la existencia de un HC en un grafo G que contenga una arista e al problema mencionado. Falta demostrar que el problema del enunciado pertenece a la categoría de problemas NP. La demostración de esto es similar a la anterior, con la particularidad de que no tenemos que verificar si una arista en particular pertenece a la secuencia. Por lo tanto, el problema de existencia de un HC en un grafo G casi 3-regular es NP-Completo.

□

Lema 2.0.5. El problema del ajedrez solitario en un tablero de $N \times M$ para N y M dados está en la categoría de problemas NP.

Demuestra. Hay que demostrar que un posible candidato a solución del problema de ajedrez solitario es verificable en un tiempo polinomial. Sea p la cantidad de piezas en la posición, se puede verificar en tiempo lineal si un movimiento es válido. Dado que $p \leq N * M$, si existe una solución en dicha posición entonces la secuencia de movimientos será de longitud $p - 1$, por lo que dicha longitud es lineal en función de la cantidad de piezas en la posición. Entonces, como la cantidad de movimientos es lineal en función de p y la legalidad de cada movimiento es verificable en un tiempo lineal en p , se puede verificar el candidato a solución en un tiempo polinomial y, por lo tanto, el problema del ajedrez solitario es NP. \square

Para demostrar que el problema de ajedrez solitario en un tablero $N \times M$ utilizando como piezas sólo torres, alfiles y peones es NP-Completo definiremos 4 maneras diferentes de generar una posición de ajedrez solitario en base a un grafo casi 3-regular.

2.1. Método 1: Arcos

En este método se usa una torre como PCN y, 2 alfiles, una torre y peones como PL (PLA, PLT y PLP respectivamente). El propósito de estas PLP es que los alfiles puedan moverse diagonalmente para poder capturar los PA luego de que la PCN capture todos los PN.

2.1.1. Descripción

Los PN estarán ubicados en una diagonal con orientación de (x,y) hacia $(x - a, y + a)$ o $(x + a, y - a)$ con $a > 0$, dejando 2 casilleros vacíos de por medio. Es decir, dado un PN_i en la casilla (x_i, y_i) , puede haber un PN en los casilleros $(x_i - 3, y_i + 3)$ o $(x_i + 3, y_i - 3)$. Según el grado del nodo, este puede tener en su misma fila y columna entre 2 y 3 PA:

- Si el nodo es de grado 2, habrá un PA en la casilla $(x_i, y_i + a)$ con $a > 0$ y otro en la casilla $(x_i + 1, y_i)$ o $(x_i - 1, y_i)$ (Fig. 2.3 y Fig. 2.4).
- Si el nodo es de grado 3, habrá un PA en la casilla $(x_i, y_i + a)$ con $a > 0$, $(x_i + 1, y_i)$ y $(x_i - 1, y_i)$ (Fig. 2.5)

Diremos que PN_i está conectado por derecha o izquierda a otro si la conexión es mediante el PA que está en los casilleros $(x_i + 1, y_i)$ o $(x_i - 1, y_i)$ respectivamente.

Para reducir las dimensiones del tablero se debe ubicar el PN de menor grado al final con el PA adyacente a su izquierda o al principio con su PA adyacente a la derecha para no generar una columna extra. De ahora en más se considerará el primer caso.

Debido a que la distancia entre los PN es de 3 casilleros en la coordenada x , inicialmente tendremos un tablero con un ancho de $3n - 1$ casilleros con n la cantidad de PN. Al mismo tiempo, como la diferencia entre las coordenadas y es la misma, tendremos un tablero con un alto de $3(n - 1) + 1$ casilleros.

Hay diferentes maneras de conectar los distintos nodos en el tablero. Sean PN_j y PN_k PN y (x_j, y_j) y (x_k, y_k) sus respectivos casilleros, las formas de conectarlos son:

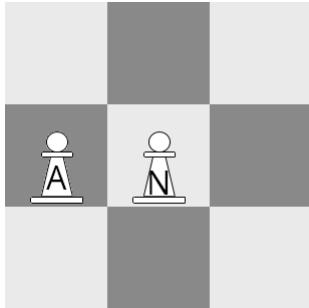


Figura 2.3: Possible alineación horizontal de un PN de grado 2

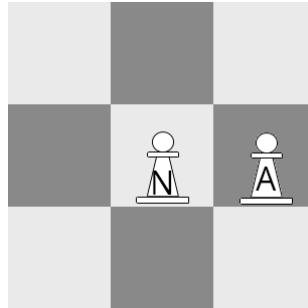


Figura 2.4: Possible alineación horizontal de un PN de grado 2

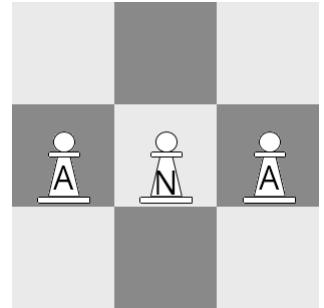


Figura 2.5: Alineación horizontal de un PN de grado 3

- Utilizando 2 PA donde uno estará en la misma columna que PN_j , otro en la de PN_k y ambos en la misma fila. Es decir un PA estará en la casilla $(x_j, y_j + a_1)$ y $(x_k, y_k + a_2)$ con $a_1 > 0$, $a_2 > 0$ y $y_k + a_2 = y_j + a_1$.
- Utilizando 3 PA los cuales estarán en la casilla $(x_j + h, y_j)$, $(x_j + h, y_j + a_1)$ y $(x_k, y_k + a_2)$ con $h = \pm 1$, $a_1 > 0$, $a_2 > 0$ y $y_k + a_2 = y_j + a_1$. Según si $h = 1$ o $h = -1$ se dirá que PN_j está conectado por derecha o por izquierda a PN_k respectivamente.
- Utilizando 4 PA los cuales estarán en las casillas $(x_j + h_1, y_j)$, $(x_j + h_1, y_j + a_1)$, $(x_k + h_2, y_k)$ y $(x_k + h_2, y_k + a_2)$ con $h_1 = \pm 1$, $h_2 = \pm 1$, $a_1 > 0$, $a_2 > 0$ y $y_k + a_2 = y_j + a_1$. Según si $h_1 = 1$ o $h_1 = -1$ se dirá que PN_j está conectado por derecha o por izquierda a PN_k . Al mismo tiempo, en base al valor de h_2 se dirá lo mismo sobre la conexión de PN_k a PN_j .

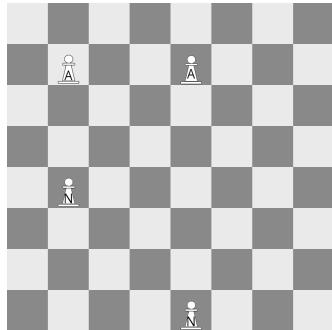


Figura 2.6: Conexión de PN a PN sin utilizar PA horizontales

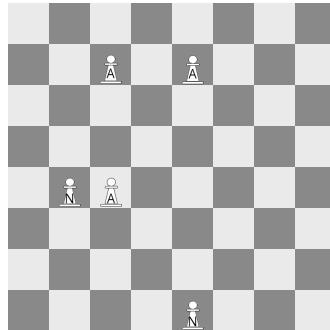


Figura 2.7: Conexión de PN a PN mediante un PA horizontal

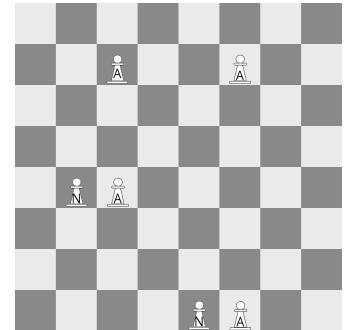


Figura 2.8: Conexión de PN a PN mediante 2 PA horizontales

Como se mencionó anteriormente se quiere evitar que las PL puedan capturar los PN. Para esto, como el alfil limpiador va a capturar a los PA, se requiere que para todo PA PA_a en la casilla (x_a, y_a) no exista ningún PN en los casilleros $(x_a + d, y_a + d)$, $(x_a - d, y_a + d)$, $(x_a - d, y_a - d)$ y $(x_a + d, y_a - d)$ con $d > 0$. Es decir, que no haya ningún PN alineado diagonalmente a algún PA.

Para lograr esto, los PA no adyacentes a los PN tendrán un valor en la coordenada y mayor o igual a $3(n - 1) + 3n - 1 = 6n - 4$. Esto se debe a que si tomamos el PN_{max_y}

el PN con el valor más alto en la coordenada y , PN_{max_x} el PN con el valor más alto en la coordenada x y los casilleros $(x_{max_y} + a, y_{max_y} + a)$ con $0 \leq a \leq x_{max_x}$ encontraremos casilleros que pertenecen a la diagonal principal de PN_{max_y} hasta la columna a la que pertenece PN_{max_x} . Luego, como los demás PN tienen un valor menor en la coordenada y que PN_{max_y} , sus diagonales principales estarán por debajo de esta. Por lo tanto toda pieza con un valor de la coordenada y mayor que $y_{max_y} + x_{max_x}$ no estará en las diagonales principales de ningún PN.

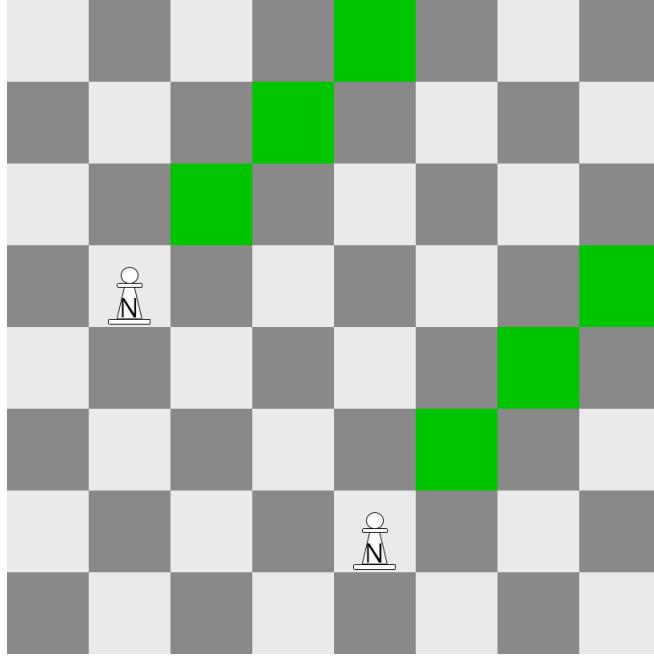


Figura 2.9: Diagonales principales de 2 PN

Por último tendremos las PL que estarán por fuera de las dimensiones especificadas hasta ahora. Las PLP estarán alineadas diagonalmente entre ellas para que la PLA pueda moverse entre las diferentes diagonales y capturar los PA. Un alfil recorrerá los casilleros negros y otro los blancos. Es necesario que estas PLA terminen alineadas horizontalmente para que una PLT pueda capturar a ambos luego de que hayan capturado todos los PA restantes y la PCN. Es importante que la PCN esté en una casilla donde pueda ser eliminada por alguna PLA, caso contrario no podrá ser capturada y terminaríamos con 2 piezas que no podrán ser capturadas.

2.1.2. Tablero básico

Por un lado, el tablero sin las PL tendrá una cota superior de la altura de $6n + m - 4 = 6n + 3(n - 5)/2 + 3 = 15n - 9/2$ ya que se necesita que los PA no estén alineados con PN diagonalmente y, para establecer las conexiones, se debe ubicar los PA no adyacentes a los PN en filas diferentes para no establecer conexiones entre PN no deseadas. Por otro lado la cota superior del ancho será de $3n - 1$ debido a la distancia entre los PN y los PA adyacentes a estos.

Al mismo tiempo, una cota superior de la cantidad de piezas será de $n + 2m + 2n - 1 = 6n - 2$ como máximo. Esto se debe a que habrá un PN por cada nodo en el grafo (n), 2

PA por cada arista ($2m$) ubicados en las filas superiores a los PN y $2n - 1$ PA adyacentes a los PN.

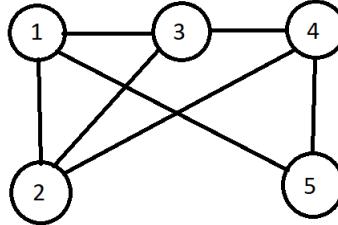


Figura 2.10: Grafo casi 3-regular con 5 nodos.

En la Fig. 2.11 se puede observar la posición generada del grafo de la Fig. 2.10.

Si bien se ubicó inicialmente todos los PN, el grafo se recorrió utilizando el algoritmo “Breadth-First Search” (BFS) para establecer las conexiones entre los diferentes PN. Aún así, no es imperativo el uso de este algoritmo. La línea negra representa el valor en la coordenada y máximo en la que los PA no adyacentes a los PN no deben estar para que al agregar las PL no exista una diagonal en donde haya un PA y un PN al mismo tiempo. Es importante aclarar que si el PA adyacente al PN_5 estuviera a la derecha en lugar de la izquierda, la línea negra estaría una fila más arriba y aumentaría el alto y ancho del tablero en 1.

2.1.3. Tablero con piezas limpiadoras

Las piezas limpiadoras se ubicarán por afuera del ancho y alto definido anteriormente, evitando que estén alineadas con los PN de manera horizontal, vertical o diagonal. Sin embargo, estarán alineadas diagonalmente a los PA. Considerando esto, algunas PL tendrán un valor en la coordenada y mayor que los PA y otras menor que los PN.

Para definir las diagonales en las que estarán ubicadas las PLP inicialmente se deberá trazar 2 diagonales:

- La primera diagonal pasará por el casillero $(1, 6(n-1)+4)$ o, en caso de que exista, por el PA en la primera columna cuyo valor en la coordenada y es mayor al mencionado. Llamaremos a esta diagonal $diag1$ y el casillero utilizado tendrá las coordenadas (x_{d1}, y_{d1}) .
- La segunda diagonal pasará por el casillero donde estará el PA de menor valor en la coordenada y , es decir $(3n - 2, 1)$. Este PA (PA_{min_y}) será adyacente al PN de menor valor en la coordenada y (PN_{min_y}). Llamaremos a esta diagonal $diag2$ y el casillero utilizado tendrá las coordenadas (x_{d2}, y_{d2}) .

La pendiente de ambas diagonales será de 1. Es decir los casilleros por las que pasa $diag1$ serán $(d + x_{d1}, d + y_{d1})$ mientras que $diag2$ pasará por $(x_{d2} + d, d + y_{d2})$ con $d \in \mathbb{Z}$. Luego, utilizando ambas diagonales se deben crear 2 diagonales nuevas para la ubicación de las PLP:

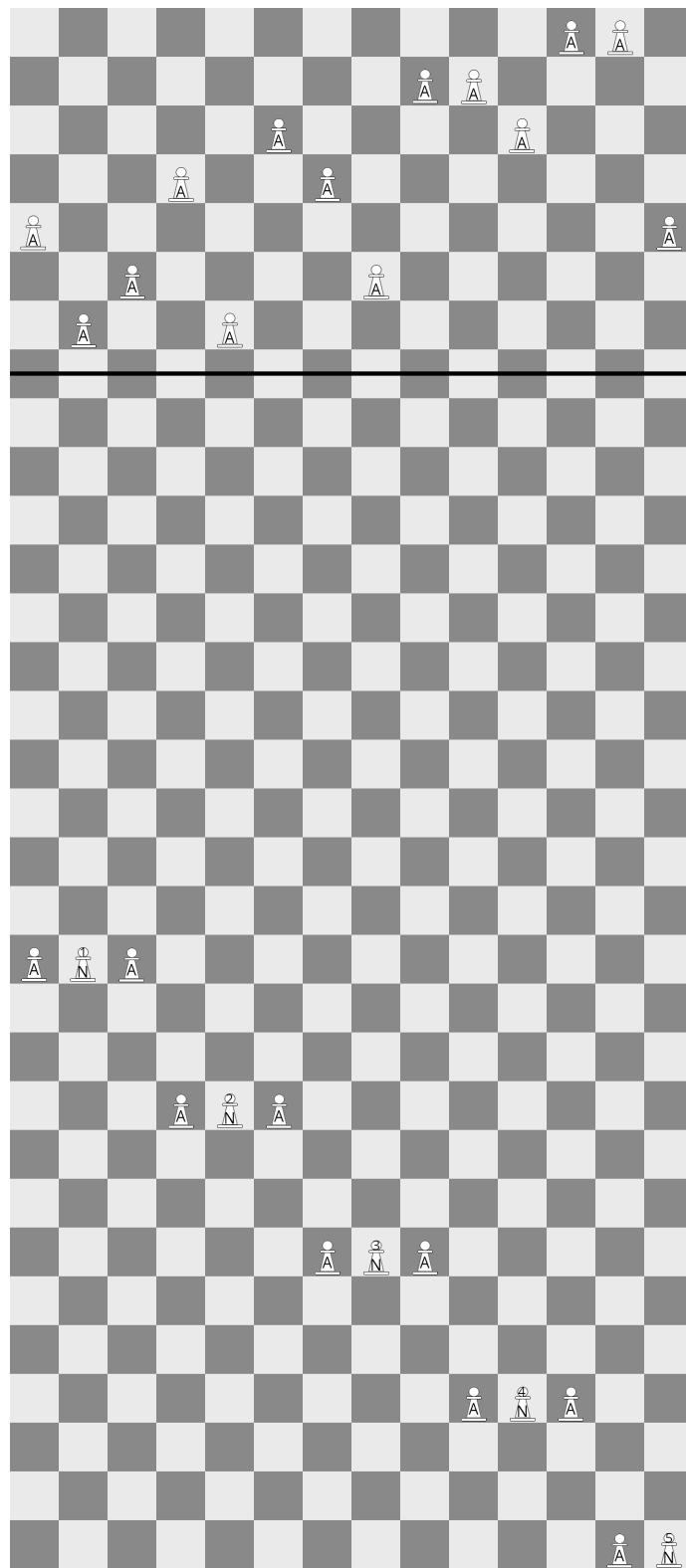


Figura 2.11: Posición generada a través del método 1 para el grafo de la Fig. 2.10.

- La primera diagonal se ubicará en el lado derecho de la posición e irá desde un casillero perteneciente a $diag1$ hasta un casillero perteneciente a la $diag2$ en caso de que ambas recorran casilleros del mismo color o hasta que ocurra una intersección con la última diagonal mencionada. La pendiente de la diagonal generada será de -1 y no debe existir ninguna pieza en la misma fila o columna de los casilleros que recorre. En caso de que no recorran casillas del mismo color, se elegirá un casillero cercano a dicha intersección tal que se genere la diagonal con las características mencionadas.
- La segunda diagonal se ubicará en el lado izquierdo de la posición e irá desde un casillero perteneciente a $diag1$ hasta un casillero perteneciente a la $diag2$ en caso de que ambas recorran casilleros del mismo color o hasta que ocurra una intersección con la última diagonal mencionada. Esta diagonal tendrá las mismas características que la diagonal anteriormente definida. En caso de que no recorran casillas del mismo color se elegirá el casillero de igual manera que el punto anterior.

En estas últimas diagonales definidas y su diagonal adyacente más “lejana” de la posición original se ubicarán las PLP. Una diagonal recorrerá casilleros de color blanco mientras que otra casilleros de color negro. Por cada PA deberá existir 2 PLP donde cada uno estará ubicado en alguna de las diagonales mencionadas y compartirán la diagonal con el PA correspondiente. Por último, se deberán ubicar las 2 PLA, una estará en un casillero de color blanco y otra de color negro. Se debe ubicarlas en la posición de las PLP con máximo y (ver la Fig. 2.12)

Si en las diagonales que recorren los casilleros del mismo color hay una cantidad par de PL, hay que agregar una PLP en cada extremo inferior de las diagonales con la condición de que estos deben estar alineados diagonalmente. En caso de no agregarlos, los alfiles quedarían en la misma diagonal por la que comenzarán luego de capturar todos los PA. Al mismo tiempo, si las nuevas PLP están alineadas horizontal o verticalmente con algún PA, se deberá mover todas las PLP ubicadas en el casillero (x, y) de la diagonal secundaria perteneciente a la nueva pieza, al casillero $(x+1, y+1)$ o $(x-1, y-1)$, según si la diagonal está en el lado superior derecho o en el lado inferior izquierdo de la posición.

Por último, será necesario ubicar 2 PLP alineadas horizontalmente entre ellas y cada una alineada a una diagonal inferior de las PLP ubicadas en casillas de diferente color y una PLT alineada horizontalmente a estos. De esta manera, luego de que se capturen todos los PA, las PLA podrán estar alineadas horizontalmente entre ellas para ser capturadas por la torre. Estas PLP pueden estar una fila por debajo de las PLP con mínimo y o una fila arriba de las PLP con máximo y perteneciente a las diagonales mencionadas. La PLT deberá estar en la misma fila, sin estar alineada vertical u horizontalmente a otra pieza que no sean estas PL.

Considerando esto, una cota superior para la cantidad de piezas será de $n + 6m + 4 + 6n - 3 = 7n + 6(7 + 3(n - 5)/2) + 1 = 16n - 2$; justificamos a continuación los distintos términos de esta fórmula:

- n debido a que existirá un PN por cada nodo.
- $6m+4$ debido a que por cada arista existirá 2 PA (sin contar los adyacentes a los PN), asumiendo que cada PA necesitará de 2 PLP para que la PLA pueda capturarla y que se necesitará 4 PLP porque todas las diagonales secundarias tendrán una cantidad par de PLP.

- $6n - 3$ debido a que cada PN tendrá 2 PA adyacentes excepto uno que será de grado 2 y que por cada PA adyacente hará falta 2 PLP para que pueda ser capturado por la PLA.

Uno podría creer que se puede reducir la cantidad de piezas limpiadoras si los PN están en casilleros de un color mientras que los PA están en casilleros de otro color. Esto haría que las PL estén en casilleros del mismo color que los PA y reduciría el alto del tablero ya que no tendríamos que considerar PA alineados diagonalmente con PN y, por lo tanto, no tendríamos que distanciar los PN de los PA no adyacentes. Sin embargo, esto complicaría (y probablemente también imposibilitaría) la generación de una posición que represente a un grafo. Esto se debe a que puede haber casos en los que no será posible representar la arista que une 2 PN. Por ejemplo si un PN en la posición (x_0, y_0) no tiene PA en los casilleros $(x_0, y_0 + a)$ y otro PN en la posición (x_1, y_1) no tiene PA en los casilleros $(x_1 + 1, y_1 + a)$ con $a > 0$. En este caso, es imposible unir a ambos ya que no existiría una forma de poner 2 PA alineados horizontalmente para crear la conexión y que estén en casilleros del mismo color que los demás PA.

Para las dimensiones del tablero, podemos pensar en la diagonal definida anteriormente por los PA de mayor y menor valor en la coordenada y :

- Para el ancho del tablero se debe calcular la distancia entre la PLP con el menor valor en la coordenada x (PLP_{min_x}) y el PA utilizado para trazar la diagonal superior donde se ubican los PLP. Al mismo tiempo se debe calcular la distancia entre la PLP de mayor valor en la coordenada x (PLP_{max_x}) y el PA adyacente al PN con menor valor en la coordenada y (PA_{min_y}).
 - La distancia del primer caso es igual a la distancia en el eje y entre el PA utilizado y PN_{min_y} debido a que se trazó una diagonal y se ubicó a PLP_{min_x} una fila por debajo de PA_{min_y} .
 - Para el segundo caso la distancia es igual a la distancia entre PA_{min_y} y el PA_{max_y} debido a que también se trazó una diagonal y se ubicó a PLP_{max_x} una fila por arriba.
 - Hay que considerar que la altura del PA utilizado para trazar la diagonal $diag1$ mencionada anteriormente no afectará en el resultado de la suma de ambas distancias. Esto se debe a que si el PA está ubicado en una fila superior, se reduce el tamaño de la diagonal que inicia desde (x_{d1}, y_{d1}) hasta la PL ubicada en el lado superior derecho de $diag1$ pero aumentará la de la diagonal que inicia desde el casillero mencionado hasta la PL ubicada en el lado inferior izquierdo de la misma. Si la pieza está ubicada en filas superiores habrá menos piezas en casilleros con valores de y mayor que este pero tendrá más piezas con valores menores. Por lo tanto, si el PA no adyacente de la primera columna está en el casillero $(x, 6(n-1)+4)$ la suma de la distancia será $6(n-1)+4+3n-1 = 9n-3$.
- Para el alto del tablero se debe calcular la distancia entre la PLP con el menor valor en la coordenada y (PLP_{min_y}) y PA_{min_y} . Al mismo tiempo se debe calcular la distancia entre la PLP de mayor valor en la coordenada x y PA_{max_y} . En ambos casos la distancia es igual a $m+3n-1$. ya que al ubicar PLP_{min_x} o PLP_{max_x} se debe tener un espacio inferior o superior respectivamente para ubicar los PLP alineados a los PA tanto para los adyacentes a los PN y los utilizados para representar las aristas.

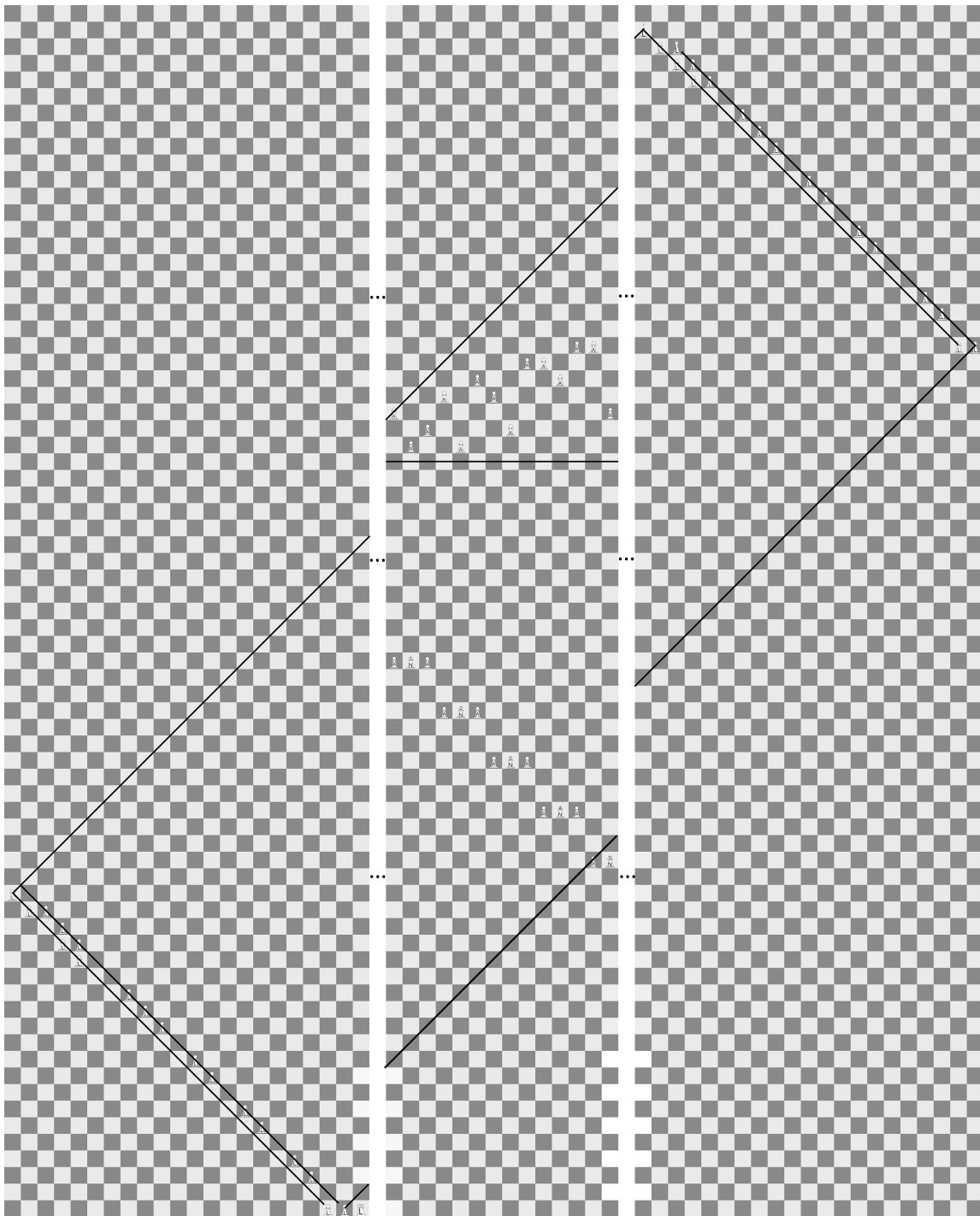


Figura 2.12: Posición generada a través del método 1 para el grafo de la Fig. 2.10 con piezas limpiadoras. Las líneas diagonales corresponden a la diagonal en la que se encuentra PA_{min_y} y aquella donde todos los PA se ubican en o por debajo de esta. Las diagonales adyacentes ubicadas en el lado superior derecho e inferior izquierdo corresponden a las posibles ubicaciones donde debe haber una PLP. La línea horizontal indica el valor en la coordenada y mínima donde deben estar los PA superiores.

Entonces, una cota superior para el ancho del tablero será de
 $3n - 1 + 6(n - 1) + 4 + 6n + 3(n - 5)/2 + 3 \leq 15n + 3(n - 5)/2 = 33/2 * n - 11/2$.
Al mismo tiempo, una cota superior para el alto del tablero será de
 $6n + 3 + 3(n - 5)/2 + 2m + 6n - 2 \leq 15n + 3(n - 5)/2 + 15 = 33/2 * n + 19/2$.

2.1.4. Algoritmo

1 **Función** agregarPLMet1(T : Posición de ajedrez solitario):

2 $PA_1 \leftarrow PA$ ubicado en la columna que está a la izquierda de PN_0 y tiene el mayor valor en la coordenada y

3 $PA_2 \leftarrow PA$ ubicado en la columna que está a la izquierda de PN_{n-1} y tiene el menor valor en la coordenada y

4 **Si** $6(n-1) + 4 > PA_{1_y}$ **entonces**

5 $diag_1 \leftarrow$ diagonal principal que pasa por el casillero $(1, 6(n-1) + 4)$

6 **Si no**

7 $diag_1 \leftarrow$ diagonal principal que pasa por el casillero $(1, PA_{1_y})$

8 **FinSi**

9 $diag_2 \leftarrow$ diagonal principal que pasa por el casillero donde está PA_{min_y}

10 Trazar una diagonal secundaria del lado derecho desde $diag_1$ hasta $diag_2$, de modo que no haya ninguna pieza en las filas y columnas por las que pasa y pase por casilleros del mismo color

11 Trazar una diagonal secundaria del lado izquierdo que cumpla las condiciones mencionadas en el paso anterior

12 **Para todo** PA_a

13 Ubicar los PLP alineados diagonalmente a PA_a y en las diagonales secundarias trazadas en los pasos anteriores o sus diagonales secundarias adyacentes de otro color

14 **Fin**

15 **Para toda** diagonal secundaria trazada con una cantidad par de PLP

16 $PL_{min_y} \leftarrow PL$ de la diagonal secundaria con menor valor en la coordenada y

17 Aregar 1 PLP en el casillero $(x_{min_y} + 1, y_{min_y} - 1)$

18 **Si** el PLP agregado está en una misma fila o columna que un PA o PN y la diagonal secundaria está del lado superior derecho **entonces**

19 **Para todo** PLP ubicado en el casillero (x, y)

20 Reubicarlo en el casillero $(x + 1, y + 1)$

21 **Fin**

22 **FinSi**

23 **Si** el PLP agregado esta en una misma fila o columna que un PA o PN y la diagonal secundaria esta del lado inferior izquierdo **entonces**

24 **Para todo** PLP ubicado en el casillero (x, y)

25 Reubicarlo al casillero $(x - 1, y - 1)$

26 **Fin**

27 **FinSi**

28 **Fin**

29 Reemplazar dos PLP que tengan el mayor valor en la coordenada y y ubicadas en casilleros de diferente color por una PLA.

30 Ubicar 2 PLP alineadas horizontalmente entre ellas y en las diagonales secundarias inferiores

31 Ubicar 1 PLT alineada a las 2 últimas PLP agregadas horizontalmente y que no esté alineada con un PA o PN

1 **Función** *unirPNMet1*(*T*: Posición de ajedrez solitario, *e*: Arista):

2 **Si** no hay un PA en la misma columna en la cual está el PN_a **entonces**

3 Ubicar un PA en el casillero $(x_a, y_a + v)$ donde $v > 3(n - 1)$ y no se haya ubicado anteriormente otro PA en la fila $y_a + v$.

4 **Si no**

5 Ubicar un PA en la casilla $(x_a - 1, y_a)$ o $(x_a + 1, y_a)$ según sea posible y otro PA en la casilla $(x_a - 1, y_a + v)$ o $(x_a + 1, y_a + v)$ respectivamente

6 **FinSi**

7 **Si** no hay un PA en la misma columna en la cual está el PN_b **entonces**

8 Ubicar un PA en el casillero $(x_b, y_b + v')$ donde $v' > 3(n - 1)$ y $y_a + v = y_b + v'$.

9 **Si no**

10 Ubicar un PA en la casilla $(x_b - 1, y_b)$ o $(x_b + 1, y_b)$ según sea posible y otro PA en la casilla $(x_b - 1, y_b + v')$ o $(x_b + 1, y_b + v')$.

11 **FinSi**

Algoritmo 1: Arcos

1 **Función** *GenerarTableroMet1*(*G*: Grafo casi 3-regular):

2 **Salida:** Posición de ajedrez solitario *T*

3 Ubicar los PN correspondientes a cada nodo de *G*. Entre uno y otro debe haber una distancia de 3 casilleros en la coordenada *x* y 3 casilleros en la coordenada *y*. Sea el primer PN_0 en la casilla (x_0, y_0) , ubicar los nodos restantes en los casilleros $(x_0 + 3d, y_0 - 3d)$ con $1 \leq d < n$

4 Dejar $3(n - 1)$ filas libres por arriba del PN con máximo *y*

5 **Para** toda arista *e* : $A \Leftrightarrow B$ en el grafo conectar el PN_a con PN_b . Para esto:

6 | *unirPNMet1*(*T*, *e*)

7 **Fin**

8 *agregarPLMet1*(*T*)

9 **Retornar** *T*

2.2. Método 2: Caminos Diagonales

Para este método utilizaremos un alfil como PCN y, una torre y peones como PL (PLT y PLP respectivamente).

2.2.1. Descripción

Los PN se ubicarán en una misma columna, dejando 5 casilleros vacíos de por medio. Es decir, dado un PN_1 en la casilla (x_1, y_1) , puede ubicarse otro PN en las coordenadas $(x_1, y_1 + 6)$ o $(x_1, y_1 - 6)$. Luego, sea (x, y) el casillero de un PN, este puede tener un PA en la casilla $(x - 2, y + 2)$ o $(x + 2, y - 2)$ según el grado del nodo al que representa:

- Si el nodo es de grado 2, tendrá un PA en la ubicación $(x - 2, y + 2)$ o $(x + 2, y - 2)$.
- Si el nodo es de grado 3, tendrá un PA en la ubicación $(x - 2, y + 2)$ y $(x + 2, y - 2)$.

Estos PA se los llamarán PA auxiliares

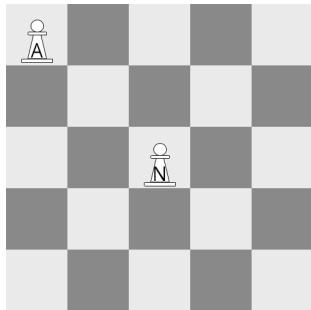


Figura 2.13: Possible alineación horizontal de un PN de grado 2.

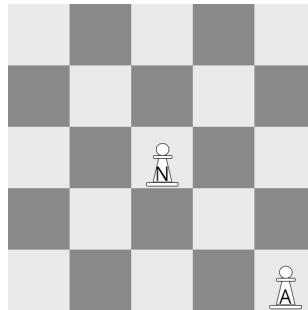


Figura 2.14: Possible alineación horizontal de un PN de grado 2

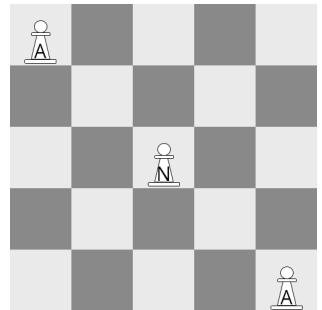


Figura 2.15: Alineación horizontal de un PN de grado 3

Para reducir las dimensiones del tablero se debe ubicar el PN de menor grado al final con el PA auxiliar a su izquierda o al principio con su PA auxiliar a la derecha para no generar dos filas extras. De ahora en más se considerará el primer caso.

Diremos que un PN está conectado por derecha o izquierda a otro si la conexión se hace mediante un PA auxiliar ubicado en la casilla $(x+2, y-2)$ o $(x-2, y+2)$ respectivamente.

Las conexiones entre los peones nodos estarán ubicadas en filas superiores a estas. Será mediante PA alineados diagonalmente a los PN o PA auxiliares a estos.

Para conectar los PN se utilizarán 2 PA o más dependiendo si se conecta por izquierda (ver la Fig. 2.17) o por derecha (2.18). Sea P_i con $1 \leq i \leq m$ el PN o PA auxiliar ubicado de arriba hacia abajo de la manera mencionada anteriormente, P_j y P_k , con $1 \leq j < k \leq n$, se seguirán los siguientes pasos:

1. Se ubicará un PA PA_{a1} en el casillero $(x_k + (t - y_k) + 1, t + 1)$ con $t \geq y_{max_y}$ y $P_{max_y} = PN_1$ o un PA auxiliar a PN_1 en base a cual tiene mayor valor en la coordenada y .
2. Se ubicará un PA PA_{a2} en el casillero $(x_j + d/2 + 1, y_j + d/2 + 1)$ con $d = (x_j + x_{a1}) + y_j - y_{a1}$.

Las PL consistirán en peones con la misma coordenada vertical u horizontal que los PA y una torre encargada de capturar todas las piezas restantes.

2.2.2. Tablero básico

La cota superior del ancho del tablero inicial (sin las PL) será de $6n + 2$. Esta expresión se obtiene al calcular la coordenada x del PA ubicado en la misma diagonal que P_m , es decir el PN con el menor valor en la coordenada y o el PA auxiliar derecho a este, y cuyo valor en la coordenada y es $y_1 + 1$. En caso de conectar con el PA auxiliar ubicado a la derecha de PN_{min_y} el ancho del tablero será mayor que si se conectara con PN_{min_y} debido a que el primero estará ubicado en filas inferiores.

Luego la cota superior para la altura será de $6n + 2 + 6n/2 + 1 = 9n + 3$ considerando si se debe conectar el PA auxiliar ubicado a la izquierda de PN_{max_y} con PN_{min_y} . Esto se debe a que utilizará, para representar la arista, un PA con el valor más alto en la coordenada y por la distancia existente entre ambos.

La cantidad de piezas utilizadas para representar las aristas del grafo es similar a la del método anterior. Es decir, por cada arista tenemos 2 PA y por cada nodo tendremos

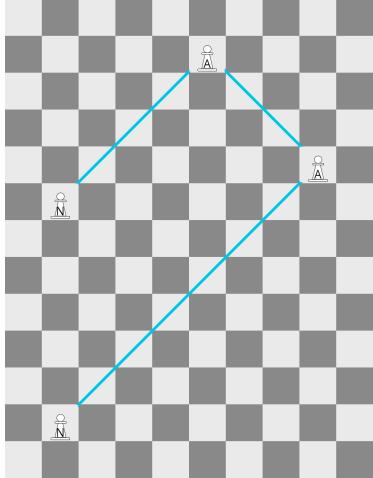


Figura 2.16: Conexión entre 2 PN.

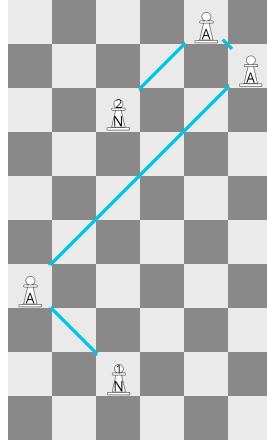


Figura 2.17: Conexión entre 2 PN donde PN_1 se conecta por izquierda a PN_2 .

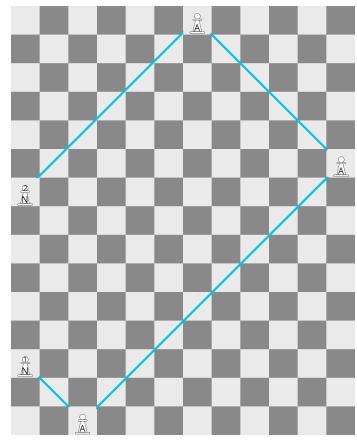


Figura 2.18: Conexión entre 2 PN donde PN_1 se conecta por derecha a PN_2 .

2 PA auxiliares excepto el que es de grado 2. Al mismo tiempo, cada nodo se representa con una pieza. Por lo tanto, una cota superior para la cantidad de piezas utilizadas en este método (sin considerar las PL) es $n + 2m + 2 * n - 1 = 6n - 2$.

Como ejemplo utilizaremos el mismo grafo ilustrado con el método 1 (Fig. 2.10).

Como se puede ver en la Fig. 2.19 los PN tiene el mismo valor en la coordenada x . El PN_5 tiene un único PA auxiliar en la casilla $(x_5 - 2, y_5 + 2)$ ya que si estuviera en la casilla $(x_5 + 2, y_5 - 2)$ se debería agregar 2 filas extras al tablero.

2.2.3. Tablero con piezas limpiadoras

Para los PA que tengan un valor en la coordenada y mayor que P_1 se tendrá un PLP a la derecha por fuera del tablero inicial y otro adentro pero en la misma columna que los PA auxiliares de la columna izquierda. La PLT deberá pasar por todas estas piezas, capturándolas y por último capturar a la PCN. Un grupo de PLP estarán en la misma columna que los PA auxiliares a la izquierda de los PN, es decir en los casilleros $(1, y_l)$ con $y_l \geq y_1$. Estos no afectarán las dimensiones del tablero. Sin embargo, se va a necesitar 2 PLP debajo de estos PA para que la PLT pueda ir de una columna de PA auxiliares a otra. Por lo tanto, se puede poner 2 PLP, una en cada columna de PA auxiliares, a 1 casillero por debajo de la primera fila de la posición generada sin PL para evitar que estén alineados con un PA diagonalmente. Es decir, en los casilleros $(1, -1)$ y $(5, -1)$ asumiendo que los PN están en la columna $x = 3$. Esto aumentaría la cantidad de filas en 1.

Al mismo tiempo, habrá un grupo de PLP ubicados una columna a la derecha del PA con máximo x . Estos se ubicarán en la columna $x_{max_x} + 1$ donde x_{max_x} corresponde al valor en la coordenada x de PA_{max_x} . Si bien habrá PA que estén en la misma diagonal que estos, no se establecerán conexiones inválidas. Esto se debe a que no puede haber 3 piezas con sus respectivos casilleros (x_1, y_1) , (x_2, y_2) y (x_3, y_3) donde $(x_1 - d_1, y_1 - d_1) = (x_2, y_2)$ y $(x_1 - d_2, y_1 - d_2) = (x_3, y_3)$ pero $(x_2 - d_4, y_2 - d_4) \neq (x_3, y_3)$ con $d_1, d_2, d_3, d_4 \in \mathbf{Z}$. Es decir, no puede haber una pieza que comparta la diagonal secundaria con 2 piezas pero que entre ellas no estén en la misma diagonal.

Los PLP ubicados a la izquierda no tendrán ningún PA ubicado en sus diagonales

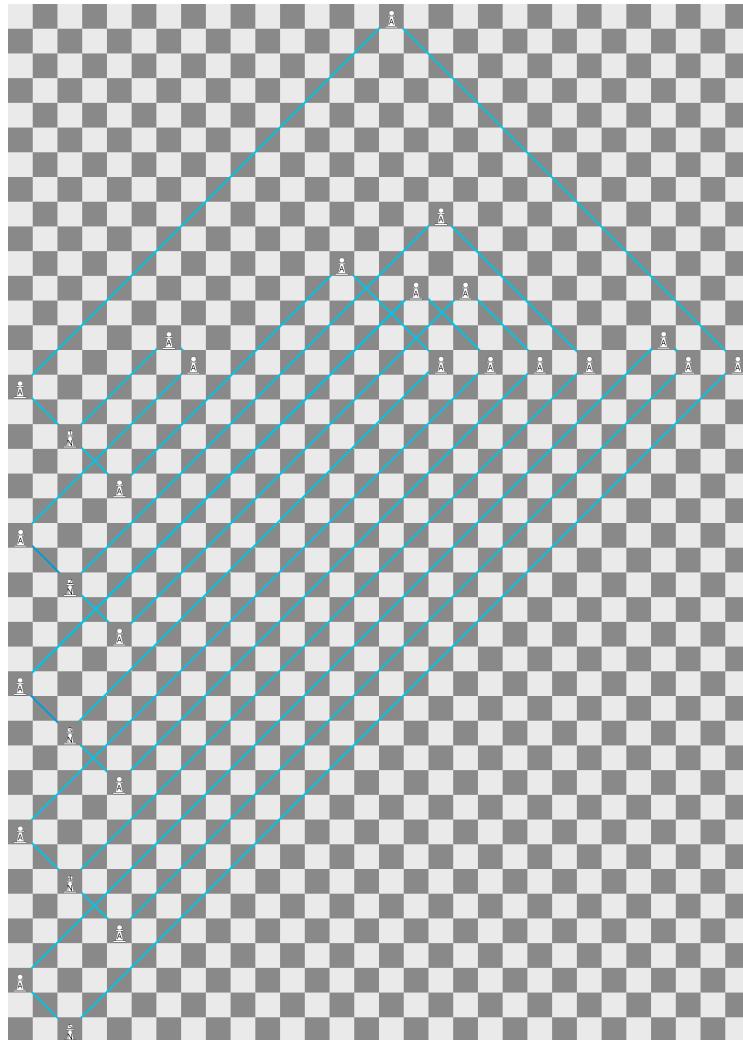


Figura 2.19: Posición generada a través del método 2 para el grafo de la Fig. 2.10

principales ni tampoco los PL ubicado a la derecha tendrán ningún PA en sus diagonales secundarias respectivamente. Esto se debe a que el PA con el mayor valor posible en la coordenada y , es decir aquel que se utilizaría para conectar P_1 con P_m , estará alineado con un PA auxiliar y el PA con el mayor valor en la coordenada x . Al mismo tiempo, al trazar una recta entre este PA y aquellos con los que está alineado se encontrará que todos los PA están ubicado debajo de estas rectas, por lo que no puede existir un PA por fuera alineado diagonalmente a las PLP de la manera mencionada anteriormente.

Para capturar los PA ubicados en los casilleros (x_a, y_a) tal que $y_a > y_1$ la PLT deberá utilizar estas 2 columnas descriptas. Las columnas permitirán a la PLT pasar por cada casillero en donde se ubiquen los PA mencionados, moviéndose de izquierda a derecha y viceversa para capturarlas. Por cada fila con un PA se requerirá 2 PLP donde cada una estará en una de las columnas mencionadas. Esto permitirá a la PLT capturar todos los PA ubicados en la fila y dirigirse hacia la siguiente.

Por último, existirá una PLT ubicada en la columna con los PA auxiliares a la derecha de los PN. Esta no debe estar alineada con ningún PN horizontal o verticalmente.

Considerando que todos los PA y PN estarán en casilleros del mismo color (caso contrario habría piezas que el PCN no podrá alcanzar debido a que sólo se mueve en casilleros del mismo color) se deberá ubicar la PLT en la coordenada $(5, y)$ con $y \geq y_1$ y el color del casillero en dicha coordenada debe ser diferente al de los PN y PA auxiliares ya que los PA ubicados a la derecha están en la columna 5. Dicho esto, no es necesario agregar más filas ni columnas.

Por lo tanto, una cota superior para la cantidad de piezas es $6n - 2 + 1 + 2 + 2m = 6n + 2(7 + 3(n - 5)/2) + 1 = 9n$. Asimismo las dimensiones del tablero serán de $6n + 3 \times 9n + 4$ ya que sólo se agregó una columna y una fila.

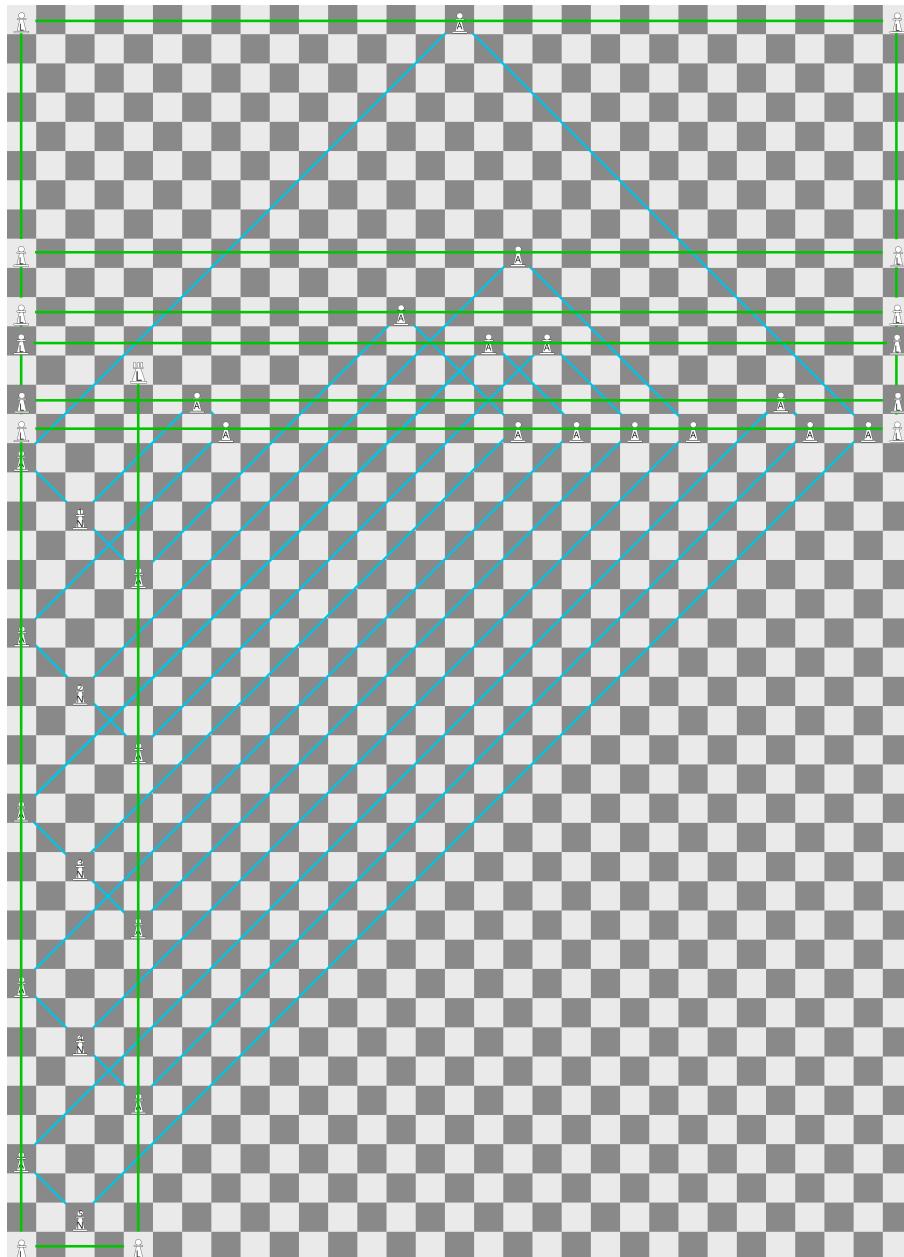


Figura 2.20: Posición generada a través del método 2 para el grafo de la Fig. 2.10 con piezas limpiadoras

2.2.4. Algoritmo

```

1 Función agregarPLMet2( $T$ : Posición de ajedrez solitario):
2    $P_{max_y} \leftarrow PN$  o  $PA$  auxiliar con mayor valor en la coordenada  $y$ 
3    $y \leftarrow y_{P_{max_y}}$ 
4    $x \leftarrow x_{PA_{max_x}}$ 
5   Para todo  $PA_a$  tal que  $y_a > y$ 
6     Si no hay un PLP en los casilleros  $(1, y_a)$  y  $(x + 1, y_a)$  entonces
7       Ubicar PLP en las casillas mencionadas
8     FinSi
9   Fin
10   $P_{min_y} \leftarrow PN$  o  $PA$  auxiliar con mayor valor en la coordenada  $y$ 
11   $y \leftarrow P_{min_y}$ 
12  Ubicar PLP en los casilleros  $(1, y - 1)$  y  $(5, y - 1)$ 
13  Ubicar una PLT en el casillero  $(5, y + 3)$ 

```

```

1 Función unirPNMet2( $T$ : Posición de ajedrez solitario,  $e$ : Arista):
2   Si no hay un  $PA$  en la casilla  $(y_{max_y} + 3 - y_b, y_{max_y} + 3)$  entonces
3     Ubicarlo en dicha casilla, la que denotaremos con  $(x_{ab}, y_{ab})$ 
4   Si no
5     Ubicar un  $PA$  en la casilla  $(x_b - 2, y_b + 2)$  o  $(x_b + 2, y_b - 2)$  y otro en la
       casilla  $(y_{max_y} - 1 - y_b, y_{max_y} + 3)$  o  $(y_{max_y} + 7 - y_b, y_{max_y} + 3)$ 
       respectivamente y denotaremos el casillero del último  $PA$  ubicado con
        $(x_{ab}, y_{ab})$ 
6   FinSi
7   Si no hay un  $PA$  en la diagonal principal que pasa por la casilla  $(x_a, y_a)$  ubicar
       un  $PA$  en la casilla  $(x_a + d/2 + 1, y_a + d/2 + 1)$  con  $d = (x_a + x_{ab}) + y_a - y_{ab}$ 
       entonces
8     ubicar un  $PA$  en la casilla  $(x_a + d/2 + 1, y_a + d/2 + 1)$  con
        $d = (x_a + x_{ab}) + y_a - y_{ab}$ 
9   Si no
10    Proceder según una de las siguientes alternativas:
        ■ Ubicar un  $PA$  en el casillero  $(x_a - 2, y_a + 2)$  y otro en
             $(x_a - 2 + d'/2 + 1, y_a + 2 + d'/2 + 1)$  con  $d' = (x_a - 2 + x_{ab}) + y_a + 2 - y_{ab}$ 
        ■ Ubicar un  $PA$  en el casillero  $(x_a + 2, y_a - 2)$  y otro en
             $(x_a + 2 + d'/2 + 1, y_a - 2 + d'/2 + 1)$  con  $d' = (x_a + 2 + x_{ab}) + y_a - 2 - y_{ab}$ .
11   FinSi

```

Algoritmo 2: Caminos Diagonales

```

1 Función GenerarTableroMet2(G: Grafo casi 3-regular):
2   Salida: Posición de ajedrez solitario T
3   Ubicar los PN correspondientes a cada nodo de G. Estos deben estar alineados
4   verticalmente con una distancia de 6 casilleros entre pares. Dado el primer
5   PN en la casilla  $(x_0, y_0)$  ubicar los PN restantes en los casilleros  $(x_0, y_0 - 6d)$ 
6   con  $1 \leq d < n$ 
7   Para toda arista  $e : A \Leftrightarrow B$  en G, conectar el  $PN_a$  con  $PN_b$  donde  $PN_b$  es el
8   nodo de menor valor en la coordenada  $y$  entre los nodos incidentes en la
9   arista  $e$ ,  $PN_a$  el nodo de mayor valor en la coordenada  $y$  entre los nodos
10  incidentes en la arista  $e$  y  $PN_{max_y}$  el PN de mayor valor en la coordenada  $y$ 
11  entre todos los PN, mediante los siguientes pasos:
12  | unirPNMet2(T,  $e$ )
13  Fin
14  agregarPLMet2(T,  $e$ )
15  Retornar T

```

2.2.5. Reducción de las dimensiones del tablero

Resulta interesante la posibilidad de representar el grafo con menos elementos, esto significa menos piezas o bien un tablero más reducido. Si bien esto no modifica esencialmente la demostración, manteniéndose el orden de complejidad lineal en el proceso de representación, surge este desafío al tratar el problema en una implementación práctica. Para reducir la altura del tablero, si PN_{max_y} es de grado 2 se puede ubicar el PA auxiliar en $(x + 2, y - 2)$. Asimismo si PN_{min_y} es de grado 2 se puede ubicar el PA auxiliar en $(x - 2, y + 2)$.

Al mismo tiempo, todas las aristas representadas tanto en la Fig. 2.19 como en la Fig. 2.20 tienen un PA 3 filas más arriba de PN_{max_y} . Esto se debe a que cada PN o un PA auxiliar a este pertenece a una diagonal principal única por lo que no sucederá que para una misma diagonal se encuentren 2 PN o PA auxiliares a estos. De esta manera, se reduce la altura a la que se tiene que ubicar uno de los PA utilizados para representar la arista y así se reduce la altura del tablero en general.

2.3. Método 3: Caminos Rectos

Para este método se utilizará una torre como PCN y un alfil y peones como PL (PLA Y PLP respectivamente).

2.3.1. Descripción

Los PN se ubicarán sucesivamente en una diagonal con orientación de (x, y) hacia $(x + a, y + a)$ o $(x - a, y - a)$ con $a > 0$, con 1 o 3 casillas vacías entre un PN y otro según la situación. Es decir, dado un PN_i en la casilla (x_i, y_i) , puede haber un PN en los casilleros $(x_i + 2, y_i + 2)$, $(x_i - 2, y_i - 2)$, $(x_i + 4, y_i + 4)$ o $(x_i - 4, y_i - 4)$. Para ubicar los PN se recorrerá el grafo utilizando un algoritmo similar al Depth-First Search (DFS) o Breadth-First Search (BFS), que consiste en los siguientes pasos:

- El primer PN se ubicará sin restricción.
- Si PN_j es vecino de PN_i en el grafo y se puede ubicar PN_j en el casillero (x_i+2, y_i+2) o $(x_i - 2, y_i - 2)$, entonces ubicarlo
- Si PN_j es vecino de PN_i en el grafo y **no** se puede ubicar PN_j a 2 casilleros de distancia de PN_i ni tampoco de los vecinos de PN_j , se agregará en el extremo superior derecho de la diagonal tras la aplicación del algoritmo de DFS o BFS.

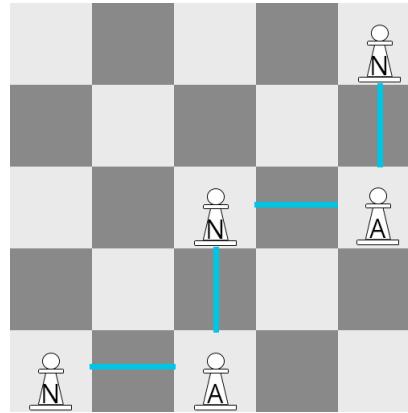


Figura 2.21: Posible alineación horizontal de un PN de grado 2

Sean PN_i con $1 \leq i \leq n$ los PN ubicados en una misma diagonal, en forma ascendente por numeración. Sean PN_j y PN_k con $1 \leq j < k \leq n$. Si la distancia entre PN_j y PN_k es de 2 casilleros en la diagonal sólo es necesario ubicar un PA en el casillero (x_k, y_j) . Es decir, debajo de PN_k y a la derecha de PN_j . Se dirá que un PN está ubicado en el borde derecho o izquierdo si hay un PN ubicado a 2 casilleros del lado izquierdo pero no del lado derecho y viceversa.

Los PN que no se pueden unir de la manera anterior debido a que su distancia es mayor que 2 casilleros se dirán que son “lejanos” entre ellos y en estos casos se tiene que proceder de acuerdo a las siguientes situaciones:

- Si PN_j está a $4d$ casilleros de distancia de PN_k se utilizará 2 PA en los casilleros $(x_j, y_j + 4d_1 + 2)$ y $(x_k, y_k + 4d_2 + 2)$ o $(x_j - 4d_1 - 2, y_j)$ y $(x_k - 4d_2 - 2, y_k)$ para establecer las conexiones, siendo $d, d_1, d_2 \in \mathbb{N}_0$, en el primer caso $y_j + 4d_1 = y_k + 4d_2$ y en el segundo caso $x_j - 4d_1 = x_k - 4d_2$ (ver la Fig. 2.22).
- Si PN_j está a $4d + 2$ casilleros de distancia de PN_k , se puede ubicar un PA en la casilla (x_j, y_k) o 3 PA en los casilleros $(x_k, y_k + 4d_2 + 2)$, $(x_j - 4d_1 - 2, y_j)$ y $(x_j - 4d_1 - 2, y_k + 4d_2 + 2)$ siendo $d_1, d_2 \in \mathbb{N}_0$. La elección de d_1 y d_2 se hará de modo que no se produzca una conexión con otro PA de otra arista (ver la Fig. 2.23).
- Si tanto PN_j como PN_k tienen su nodo más cercano a 4 casilleros de distancia se los puede conectar utilizando el lado derecho de la diagonal que contiene a los PN y la conexión es similar a los casos anteriores, tanto para cuando la distancia entre ambos PN es de $4d + 2$ como $4d$. La diferencia es al definir los casilleros en donde se

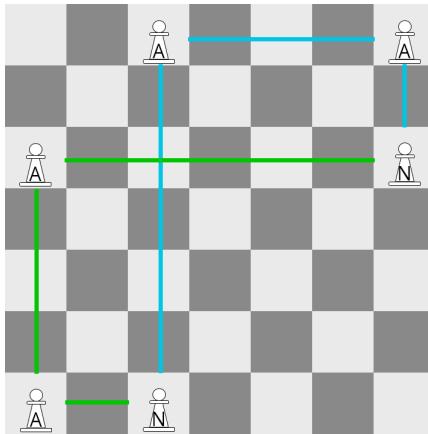


Figura 2.22: Dos formas de conexión entre 2 PN “lejanos” a $4d$ casilleros de distancia

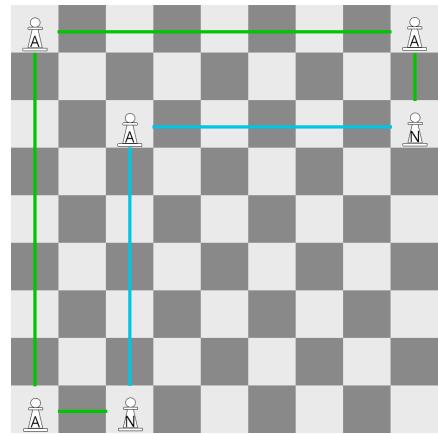


Figura 2.23: Dos formas de conexión entre 2 PN “lejanos” a $4d + 2$ casilleros de distancia

ubicarán PA, en vez de utilizar el $4d_1 + 2$ ($4d_2 + 2$) para el cálculo de la ubicación se utilizará $-(4d_1 + 2)$ ($-(4d_2 + 2)$) y (x_b, y_a) en vez de (x_a, y_b) (ver las Figs. 2.24 y 2.25).

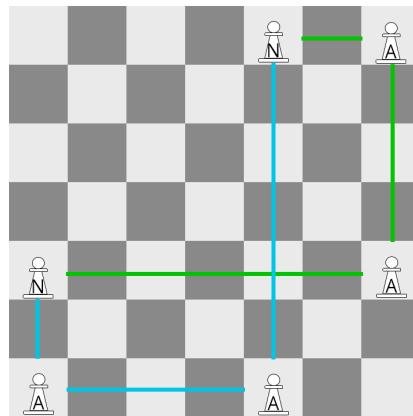


Figura 2.24: Dos formas de conexión entre 2 PN “lejanos” a $4d$ casilleros de distancia cuando el PN más cercano de ambos está a 4 casilleros de distancia.

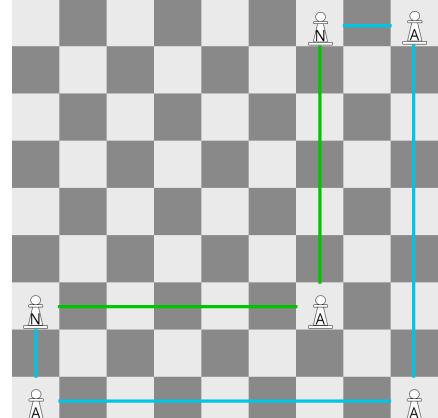


Figura 2.25: Dos formas de conexión entre 2 PN “lejanos” a $4d + 2$ casilleros de distancia cuando el PN más cercano de ambos está a 4 casilleros de distancia.

- Si PN_j está a 4 casilleros de distancia del nodo más cercano existen 2 formas de conexión adicionales.
 - Si PN_j está a $4d$ casilleros de distancia de PN_k se puede ubicar un PA en la casilla $(x_k - 4d - 2, y_k)$ y otro en $(x_j + 2, y_j)$ con $d \in \mathbb{N}_0$ y $x_k - 4d - 2 = x_j + 2$ (ver la Fig. 2.26).
 - Si PN_j está a $4d + 2$ casilleros de distancia de PN_k se puede ubicar los PA en los casilleros $(x_j, y_j - 2)$, $(x_j - 4d, y_j - 2)$ y $(x_j - 4d, y_k)$ con $d \in \mathbb{N}_0$ (ver la Fig.

2.27).

- Si PN_k está a 4 casilleros de distancia del nodo más cercano existen 2 formas de conexión adicionales.
 - Si PN_j está a $4d$ casilleros de distancia de PN_k se puede ubicar un PA en la casilla $(x_k, y_k - 2)$ y otro en $(x_j, y_j + 4d + 2)$ con $d \in \mathbb{N}_0$ y $y_k - 2 = y_j + 4d + 2$ (ver la Fig. 2.26).
 - Si PN_j está a $4d + 2$ casilleros de distancia de PN_k se puede ubicar los PA en los casilleros $(x_k + 2, y_k)$, $(x_k + 2, y_k + 4d)$ y $(x_j, y_k + 4d)$ con $d \in \mathbb{N}_0$ (ver la Fig. 2.27).

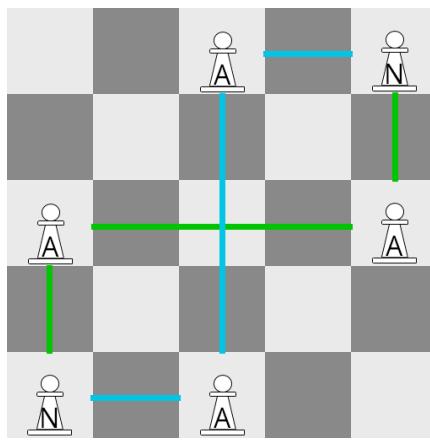


Figura 2.26: Dos formas de conexión entre 2 PN “lejanos” a $4d$ casilleros de distancia.

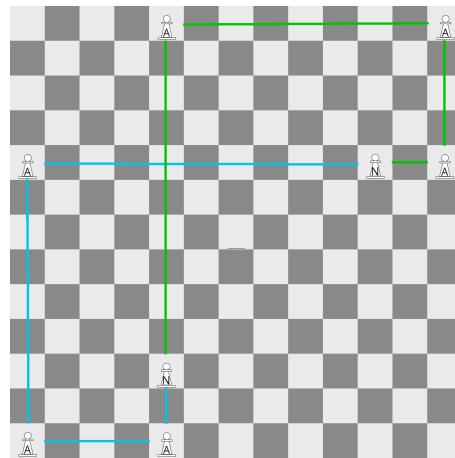


Figura 2.27: Dos formas de conexión entre 2 PN “lejanos” a $4d + 2$ casilleros de distancia.

En los casos de conexiones entre PN “lejanos” no quedarán PA alineados diagonalmente con ningún PN. Esto se debe a que como los PN tienen una distancia entre ellos de 2 o 4 casilleros en la diagonal se utiliza un PA con una distancia igual a $4d + 2$ respecto a los PN ya sea en el eje vertical u horizontal donde $d \in \mathbb{N}_0$, con el objetivo de evitar las diagonales donde están ubicados los PN. En particular, al sumar o restar 2 en la coordenada x o y del casillero donde se ubica un PN, obtendremos un casillero que no pertenece a las diagonales de estos y, al sumar o restar $4d + 2$ obtendremos otros casilleros que tampoco pertenecen a estas diagonales. Esto no sucede en el caso donde sólo sumemos o restemos $4d$ a las coordenadas mencionadas de los PN, ya que obtendríamos como resultado un posible casillero perteneciente a las diagonales de un PN.

2.3.2. Tablero básico

Para conectar los PN “lejanos” entre sí no necesariamente deben estar los PA en diagonales diferentes, sin embargo para obtener una cota de las dimensiones del tablero sin piezas limpiadoras consideraremos que sí lo están.

Para la cantidad de piezas definiremos una cota máxima debido a que existe la posibilidad de que algunas piezas no tengan una conexión con un PN cercano, es decir todas

sus conexiones son con PN “lejanos”. Por un lado tendremos n piezas correspondientes a los PN. Por otro lado asumiremos que para representar las aristas necesitaremos la mayor cantidad de PA posibles, o sea 3 PA. Como conclusión la cantidad de piezas será de $n + 3m = n + 9(n - 5)/2 + 21 = 11/2n - 3/2$ (n impar de acuerdo con el lema 2.0.1).

Para las dimensiones del tablero tendremos que considerar la diferencia entre las coordenadas y y entre las coordenadas x de las piezas con mayor y menor valor en ambas coordenadas. El peor caso será aquel donde para unir PN “lejanos” siempre hay que utilizar PA arriba o a la izquierda de estos, afectando el alto o el ancho del tablero respectivamente.

Para el alto del tablero, los PA que se utilizarán para conectar PN “lejanos” estarán en diferentes filas que otros PN y PA. Dicho esto, una cota superior del alto del tablero será $4n + 4(7 + 3(n - 5)/2) + 2 = 10n$. Esto se debe a la distancia máxima que puede haber entre un PN y otro y las formas de conexión de 2 PN cuando ambos tienen una distancia de $4d$ o $4d + 2$, sin contar las maneras de conectarlos cuando el PN más cercano está a 3 casilleros de distancia.

La cota del ancho del tablero es igual a la cota del alto del tablero¹. En vez de ubicar los PA por arriba de los PN se los ubicaría a la izquierda lo cual requeriría agregar más columnas. Al mismo tiempo, existen formas de conexión de PN “lejanos” donde un PA tendrá un valor en la coordenada x de $x_n - 4d - 2$ donde x_n es el valor en dicha coordenada de un PN y $d \geq 0$ y no tiene que existir otro PN o PA perteneciente a otra arista con el mismo valor en la coordenada x .

Para el mismo grafo utilizado en el método 1 (ver la Fig. 2.10), se genera la posición de la Fig. 2.28.

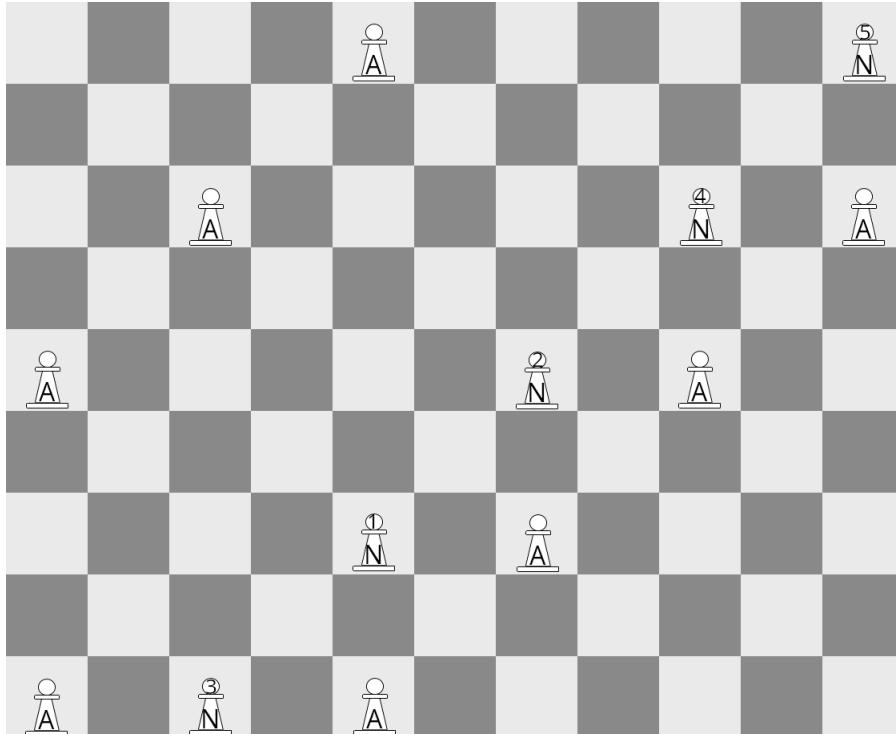


Figura 2.28: Posición generada a través del método 3 para el grafo de la Fig. 2.10

¹Esto puede entenderse dada la simetría en el tratamiento de las coordenadas x e y en este método particular.

Todos los PN están en la misma diagonal y ningún PA está en la misma diagonal que algún PN.

2.3.3. Tablero con piezas limpiadoras

Las PLP estarán en la misma diagonal que los PA para que puedan ser capturadas por el alfil. Por cada diagonal donde hay un PA habrá 2 PLP, uno del lado izquierdo y otro del lado derecho del PA, para que el alfil limpiador pueda capturarlos a todos, haciendo un recorrido “en zigzag”. Por lo tanto, una cota superior de la cantidad de piezas será de $n + 9m + 1 = n + 27(n - 5)/2 + 64 = 29/2n - 7/2$ asumiendo que cada PA estará en una diagonal diferente y cada arista utilizará 3 PA.

Estas PLP estarán por fuera de las dimensiones de la posición ya construida para evitar conexiones inválidas entre PN. Para encontrar la ubicación de estas PLP, se seguirán los siguientes pasos:

1. Se trazará una diagonal por cada PA y que tenga una pendiente de 1. Se llamará $diag_{der}$ a la diagonal ubicada a la derecha de los PN y más alejada de estos y, asimismo, $diag_{izq}$ a la diagonal ubicada a la izquierda de los PN y más alejada.
2. Luego se trazarán dos diagonales, una del lado superior derecho y otra del lado inferior izquierdo, ambas con pendiente -1 . Estas atravesarán las diagonales trazadas anteriormente y no pasarán por ninguna fila o columna en las que haya un PN. Por un lado, la diagonal ubicada en el lado superior iniciará desde un casillero con un valor de $x_{max_x} + 1$ en la coordenada x . Por otro lado, la diagonal ubicada en el lado inferior iniciará desde un casillero con un valor de $y_{min_y} - 1$ en la coordenada y . Los casilleros donde iniciarán estas nuevas diagonales deberán pertenecer a $diag_{izq}$. En los casilleros que están en la intersección entre dos diagonales se ubicarán las PLP.
3. Se reemplazará cualquier PLP por la PLA.

Considerando que las PLP deben tener distinto valor en las coordenadas x e y que los PN y que habrá 2 PLP por cada diagonal donde haya un PA, la cota superior del alto y ancho del tablero será de $10n + 2(12m + 1) = 28n - 9$. Esto se debe a que:

- El término $10n$ corresponde al tamaño del tablero sin PL.
- El término $12m + 1$ hace referencia a la distancia en el eje x o y entre $diag_{der}$ y $diag_{izq}$. Entre cada diagonal hay una distancia mínima de 4 casilleros teniendo en cuenta las distintas formas de conexión, donde para ubicar un PA se le suma a la coordenada x o y de un PN el valor de $4d + 2$ con $d \geq 0$. Al mismo tiempo, $4m$ es una cota superior a la distancia en el eje x o y entre la diagonal de los PN ($diag_{PN}$) y $diag_{izq}$. Se suma 1 para poder separar las PL de los PN a fin de que no estén en la misma fila o columna. Por último, la distancia entre $diag_{PN}$ y $diag_{izq}$ afectará hasta dónde deben llegar las diagonales ubicadas del lado superior derecho e inferior izquierdo y qué tan lejos estarán los casilleros donde inician estas diagonales de la posición generada sin piezas limpiadoras.

Se debe aclarar que estas cotas se obtienen asumiendo que cada PA está ubicado en una diagonal diferente. En la práctica esto no sucede, por ejemplo todos los PN cercanos se unen utilizando PA en una misma diagonal y los PA utilizados para representar aristas de PN “lejanos” pueden compartir diagonales entre diferentes conexiones.

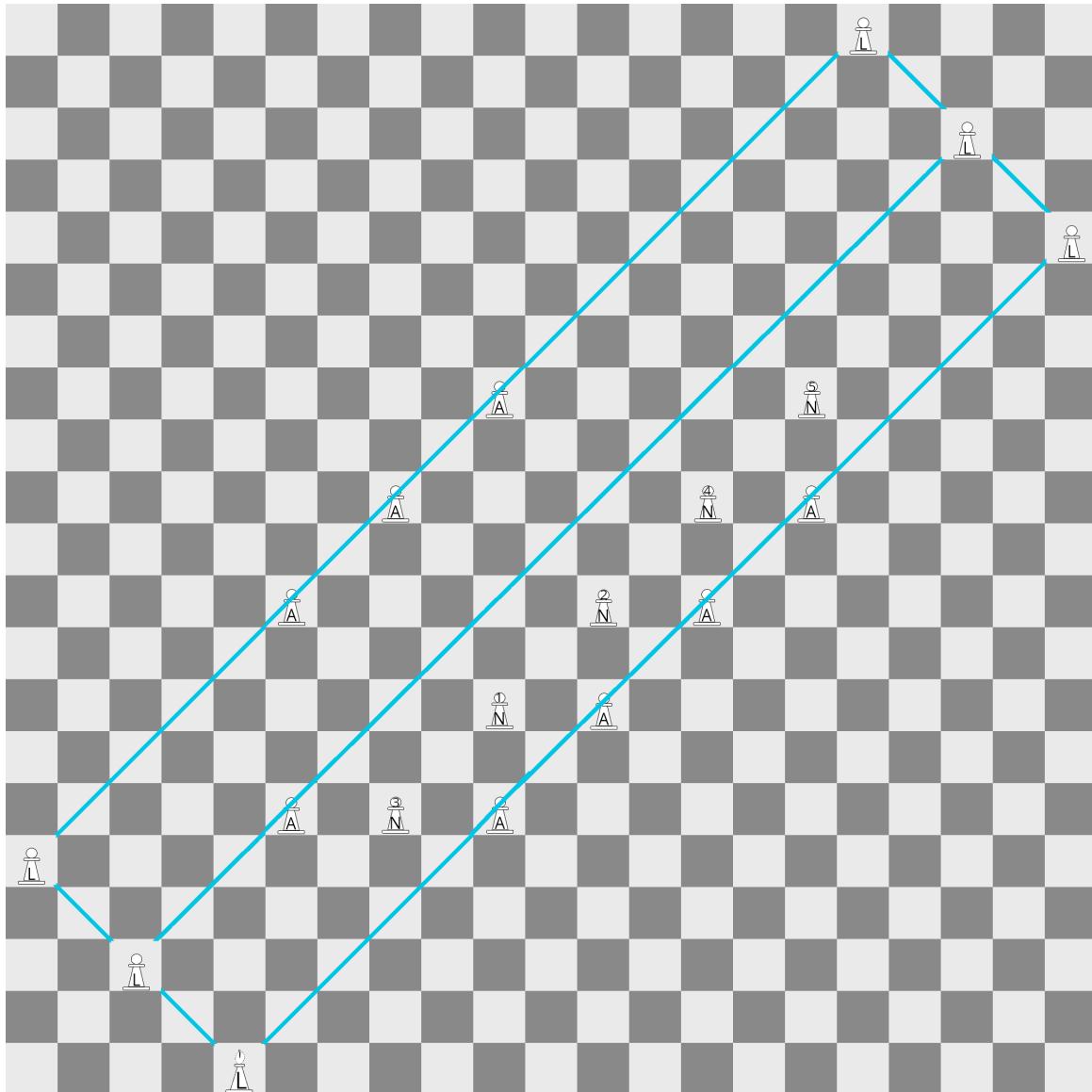


Figura 2.29: Posición generada a través del método 3 para el grafo de la Fig. 2.10 con PL

2.3.4. Algoritmo

Al generar las aristas, si existen condiciones especiales sobre PN incidentes (por ejemplo 1 o ambos tienen el PN más cercano a 4 casilleros de distancia) se debe utilizar la forma de conexión para estos casos. Este orden se debe a las posibles conexiones que tienen los distintos PN y la cantidad de conexiones que debe hacer cada uno.

La idea es recorrer todas las aristas, empezando desde un nodo inicial PN_a y luego expandirse hacia sus nodos adyacentes. Si se encuentra un nodo PN_b el cual no está ubicado aún en la posición pero sí su vecino PN_a y este es de grado 2 en la posición, se lo ubicará en la pila para luego insertar estos en la posición y establecer las conexiones restantes.

```

1 Función agregarPLMet3( $T$ : Posición de ajedrez solitario):
2   Para todo  $PA$ 
3     | Trazar la diagonal principal del PA
4   Fin
5    $diag_{izq} \leftarrow$  diagonal principal sin otra a su izquierda.
6    $diag_{der} \leftarrow$  diagonal principal sin otra a su derecha.
7   Trazar dos diagonales con pendiente  $-1$  del lado superior e inferior tal que
     inician en  $diag_{izq}$ , terminen en  $diag_{der}$  y ninguno de sus casilleros
     pertenezcan a una fila o columna donde hay un  $PA$  o  $PN$ 
8   Ubicar PLP en los casilleros que están en la intersección entre las últimas dos
     diagonales trazadas y las diagonales principales
9   Reemplazar cualquier PLP por una PLA

```

Algoritmo 3: Caminos Rectos

```

1 Función GenerarTableroMet3(G: Grafo casi 3-regular):
2   Variables: Conjunto vacío C y una pila vacía P
3   Salida: Posición de ajedrez solitario T
4   Ubicar un nodo de G en la casilla  $(x_0, y_0)$ 
5   Ubicar en P todas las aristas del nodo
6   Mientras haya aristas en P  $A \Leftrightarrow B$  y siendo  $PN_a$  y  $PN_b$  los PN
7     correspondientes a los nodos A y B respectivamente hacer
8     Si  $PN_a$  es de grado 2 o  $PN_b$  está ubicado en la posición entonces
9       Agregar  $A \Leftrightarrow B$  a C
10      FinSi
11    Fin
12  Para todo nodo n que no esté en la posición pero si en el grafo
13     $PN_{max_y} \leftarrow$  PN con mayor valor en la coordenada y
14    Ubicar el PN que representará a n en el casillero  $(x_{max_y} + 4, y_{max_y} + 4)$ 
15  Fin
16  Para toda arista e :  $A \Leftrightarrow B$  en C conectar los PN del siguiente modo:
17    Se establecerán las conexiones con la siguiente prioridad:
18    1. Conexiones donde ambos PN están ubicados en los bordes. Es decir, tienen un PN
19      a 2 casilleros de distancia de un lado pero no del otro.
20    2. Conexiones que involucren un PN ubicado en un borde y un PN a 4 casilleros de
       distancia al PN más cercano.
21    3. Conexiones donde hay al menos un PN a 4 casilleros de distancia del PN más
       cercano.
22    4. Conexiones restantes.
23    unirPNLejanosMet3(T, e)
24  Fin
25  agregarPLMet3(T)
26  Retornar T

```

Siguiendo estos pasos toda arista del grafo puede ser representada en la posición. Considerando PN_a , PN_b y PN_c tres PN, sucederá lo siguiente:

- Si ambos están a 2 casilleros de distancia es trivial la conexión a establecer.
- Si ambos no están ubicado en los bordes, no tienen el PN más cercano a 4 casilleros de distancia y no están a 2 casilleros de distancia entre ellos, hay dos maneras de conectarlos. Esto se debe a que si hay una conexión entre ambos en el grafo y se cumplen las condiciones mencionadas, entonces las otras conexiones que tienen ambos PN son con sus vecinos y están ubicados a 2 casilleros de distancia de estos.
- Si un PN_a está ubicado en el borde y el PN_b no está en el borde restante y también tiene un PN a 2 casilleros de distancia, implica que este último tiene dos maneras de

establecer la conexión mientras que el PN_a tiene como mínimo una manera. Esto se debe a que PN_a puede tener como vecinos en el grafo a PN_b y PN_c y en la posición están ambos a $4d$ o $4d + 2$ casilleros de distancia siendo $d \geq 1$ o únicamente tiene como vecino a PN_b en caso de ser de grado 2. En cada caso existen dos maneras diferentes de conectarlos. Al mismo tiempo, PN_b tiene dos PN a 2 casilleros de distancia y su única conexión a un PN “lejano” va a ser con PN_a .

- Si ambos están ubicados en los bordes y no tienen al PN más cercano a 4 casilleros de distancia entonces cada uno tiene una conexión con un PN a 2 casilleros de distancia y tienen 2 formas de establecer una conexión, ya sea cuando el PN esta a $4d$ y $4d + 2$ casilleros de distancia.
- Si sólo uno tiene el PN más cercano a 4 casilleros de distancia este puede establecer la conexión de una manera especial y no se verá afectado al establecer las otras conexiones, debido a que la ubicación del primer PA saliente de este no corresponde con ninguna de las otras formas de establecer la conexión en los casos en que el nodo más cercano está a 2 casilleros.
- Si ambos tienen el PN más cercano a 4 casilleros de distancia, se puede utilizar las conexiones especiales para estos casos. Estas no afectarán en las posibles formas de conectar con otros PN debido a que las otras maneras de conexión de PN “lejanos” no utilizan el lado derecho de la diagonal de PN.

2.4. Método 4: Triángulos

Este método utilizará un alfil como PCN y torres y peones como PL (PLT y PLP respectivamente).

2.4.1. Descripción

Los PN se ubicarán en una misma fila con 3 o 7 casilleros vacíos entre ellos. Es decir, se ubicarán en una misma fila con dirección de (x, y) hacia $(x + a, y)$ con $a > 0$. Esta última distancia sólo se utilizará cuando no se pueda ubicar el PN a una distancia de 3 casilleros de alguno de sus vecinos, esto es, dado un PN_i en la casilla (x_i, y_i) puede haber un PN en los casilleros $(x_i + 4, y_i)$, $(x_i - 4, y_i)$, $(x_i + 8, y_i)$ o $(x_i - 8, y_i)$. Un par de PN serán considerado “cercanos” si hay una distancia de 4 casilleros entre ellos y “lejanos” si hay una distancia mayor que 4 casilleros. Se dirá que un PN está ubicado en el borde derecho si hay un PN ubicado a 4 casilleros del lado izquierdo pero no del lado derecho y un PN está ubicado en el borde izquierdo si hay un PN ubicado a 4 casilleros del lado derecho pero no del lado izquierdo.

Sean PN_i con $1 \leq i \leq n$ los PN ubicados de izquierda a derecha en la misma fila. Dados PN_j y PN_k con $1 \leq j < k \leq n$, habrá diferentes maneras de representar la arista:

- Si las coordenadas de PN_j y PN_k son (x_j, y_j) y $(x_j + 4, y_j)$ respectivamente se utilizará un PA en la casilla $(x_j + 2, y_j + 2)$ (ver la Fig. 2.30)
- Si las coordenadas de PN_j y PN_k son (x_j, y) y (x_k, y) respectivamente donde $x_k \neq x_j + 4$ se utilizará 2 PA. El primer PA estará en la casilla $(x_j - 1, y - 1)$ y el segundo en $(x_j - 1 + \lfloor (x_k - x_j + 1)/2 \rfloor, y - (x_k - x_j)/2 - 1)$ o el primero estará en $(x_k + 1, y - 1)$

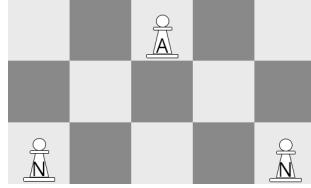


Figura 2.30: Conexión de 2 PN

y el segundo en $(x_j + \lceil (x_k + 1 - x_j)/2 \rceil, y_j - (x_k - x_j)/2 - 1)$. Cuando la conexión se hace de esta manera se dirá que PN_j está conectado por izquierda o PN_k está conectado por derecha respectivamente (ver las Figs. 2.31 y 2.32).

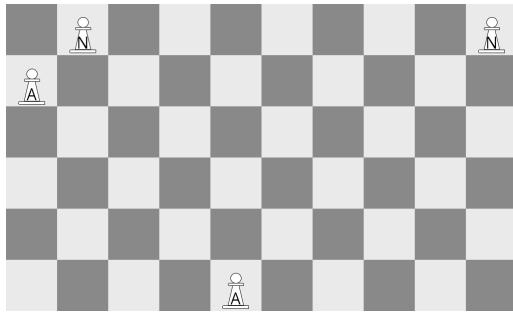


Figura 2.31: Conexión de 2 PN ‘lejanos’ utilizando un PA a la izquierda

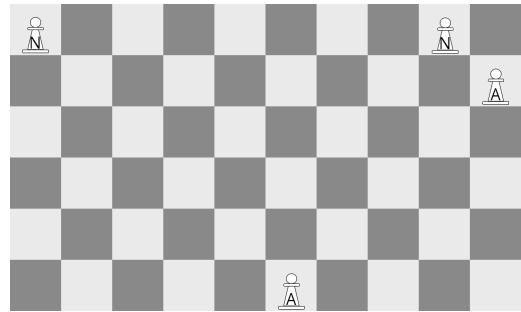


Figura 2.32: Conexión de 2 PN ‘lejanos’ utilizando un PA a la derecha

- Si PN_j tiene el nodo más cercano a una distancia de 8 casilleros hacia PN_k o viceversa, establecemos una conexión utilizando un primer PA en $(x_j + 1, y_j + 1)$ y otro en $(x_j + 1 + \lceil (x_k - x_j - 1)/2 \rceil, y_j + 1 - \lfloor (x_k - x_j + 1)/2 \rfloor)$ o un primer PA en $(x_j - 1, y_j + 1)$ y otro en $(x_k - 1 - \lceil (x_k - x_j - 1)/2 \rceil, y_k + 1 - \lceil (x_k - x_j - 1)/2 \rceil)$ respectivamente (ver Figs. 2.33 y 2.34).

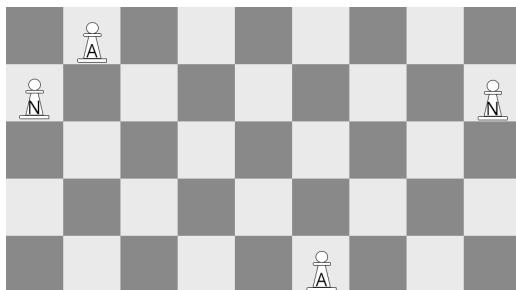


Figura 2.33: Conexión de 2 PN ‘lejanos’ donde el nodo más cercano a uno está a 8 casilleros de distancia

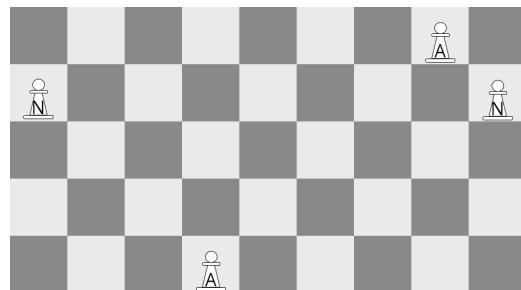


Figura 2.34: Conexión de 2 PN ‘lejanos’ donde el nodo más cercano a uno está a 8 casilleros de distancia

- Si PN_j y PN_k están en el borde izquierdo y derecho respectivamente o tienen el PN más cercano a 8 casilleros de distancia se utilizará un PA en la casilla $(x_j - 1, y_j + 1)$, otro en $(x_k + 2, y_k + 2)$ y en $(x_j - 1 + \lceil (x_k + 3 - x_j)/2 \rceil, y_j + 1 + \lceil (x_k + 3 - x_j)/2 \rceil)$ (ver Fig. 2.35).

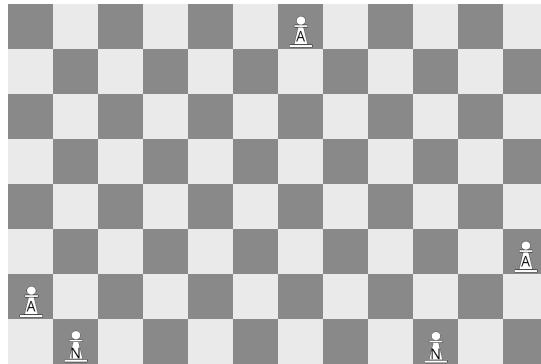


Figura 2.35: Conexión de 2 PN “lejanos” utilizando el método alternativo en ambos lados

Para conexiones entre PN “lejanos” los PA no quedarán alineados horizontal o verticalmente con PN por las siguientes razones:

- Algunos PA se agregan alineados diagonalmente a los PN a 1 o 2 casilleros de distancia para conectarlo con un PN “lejano”. Llamaremos a estos PA “auxiliares”.
- Al agregar los PA utilizados para conectar los PN “lejanos” no se los ubica en la columna donde la distancia entre los PN es la mitad sino en la columna siguiente o anterior. Como la distancia entre los PN es par, el casillero ubicado en la fila de los PN y en la columna que marca la mitad de la distancia entre ambos PN será del mismo color que los casilleros donde están estos. Al moverse una columna más a la derecha o izquierda, corresponderá un casillero de un color diferente por lo cual en dicho casillero no habrá un PN.

2.4.2. Tablero básico

Excepto por los PA auxiliares que se ubican con el fin de conectar PN de un extremo a otro, todos los PA se ubican entre los PN de los extremos. Por lo tanto, una cota superior del ancho del tablero será $8(n - 1) + 3$ debido a que la distancia máxima que puede haber entre un PN y otro es de 8 y por la necesidad de agregar un PA a la izquierda y a la derecha de los extremos para representar una arista entre los PN en los bordes.

Para el alto del tablero se tiene que tener en cuenta las diferentes formas de conexiones. Una forma de conexión se da cuando se unen 2 PN “lejanos” agregando PA en las filas superiores de los PN. Otra forma de conexión se da cuando se agregan PA en las filas inferiores. Por lo anterior, una cota superior será de

$$8(n - 1)/2 - 2 + \lceil (8(n - 1) + 3)/2 \rceil + 2 = 4(n - 1) + \lceil (8(n - 1) + 3)/2 \rceil$$

asumiendo que el PN_1 está en la coordenada (1,1), que la distancia entre ambos extremos es de $8(n - 1)$ y considerando la fila donde estarán los PN.

Por otro lado, la cantidad de piezas es de $n + 3m = n + 9(n - 5)/2 + 21 = 11n/2 - 3/2$ debido a que para cada nodo se utilizará un PN y la cantidad máxima de PA por arista es de 3.

Ejemplificando con el mismo grafo utilizado en el método 1 (ver la Fig. 2.10), se genera la posición correspondiente a la Fig. 2.36. Todos los PN están en la misma fila y no hay PA en las columnas de estos.

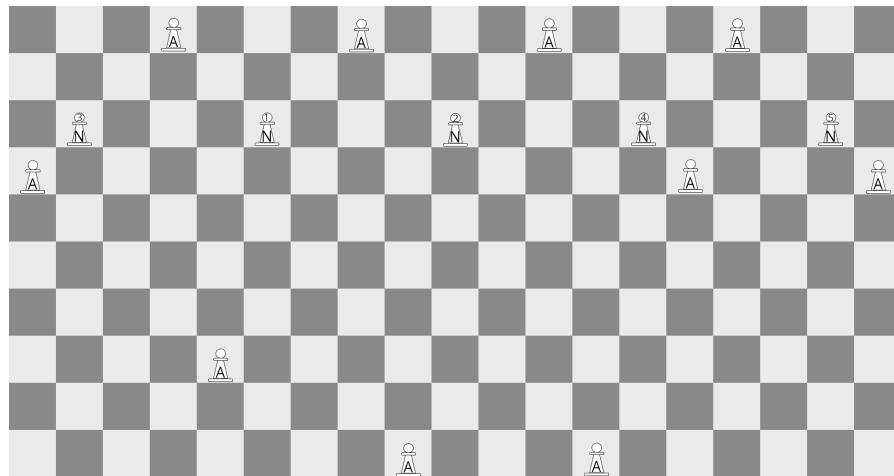


Figura 2.36: Posición generada a través del método 4 para el grafo de la Fig. 2.10

2.4.3. Tablero con piezas limpiadoras

Las PLP estarán ubicadas en la mismas filas que las PA y a una columna de distancia del PA_{min_x} y PA_{max_x} , agregando dos columnas nuevas. Lo anterior tiene la función de que la PLT pueda pasar por cada PLP capturando cada PA. En caso de que la PCN capture una PLP no tendrá otro movimiento posible que no implique pasar por el casillero en el que estaba anteriormente debido a que, para todo PLP, este no tendrá 2 piezas alineadas diagonalmente y en diferentes diagonales dadas las ubicaciones de los mismos.

Considerando esto, la cota superior del alto del tablero no se modifica. Sin embargo, la cota superior para el ancho del tablero será de $8(n - 1) + 5$.

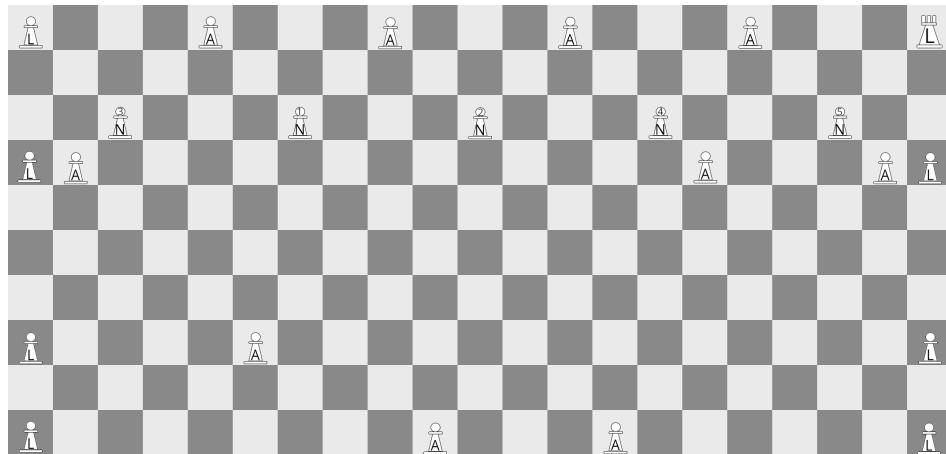


Figura 2.37: Posición generada a través del método 4 para el grafo de la Fig. 2.10 con PL

La ubicación de la PLT debe ser alguna donde existiría una PLP para poder pasar por todos los PA. En el peor caso cada PA que conecta PN “lejanos” estará en diferentes filas. Para el cálculo de la cantidad de PL hay que tener en cuenta que:

- Habrá 2 filas por arriba y 1 fila por debajo de los PN que tendrán PA utilizados para conectar PN “lejanos”.

- Se considerará el caso donde todos los PN están a 8 casilleros de distancia y se calculará una cota máxima de la cantidad de PL.

Por lo tanto, la cota máxima de la cantidad de PL dependerá directamente de la cantidad de aristas. Por ende, dicha cota máxima será de $6+2m = 3n+5$ y la cota superior sobre la cantidad de piezas totales será de $n+5m+5 = n+15(n-5)/2+40 = 17/2n+5/2$.

2.4.4. Algoritmo

Si existen condiciones especiales sobre los PN incidentes (por ejemplo uno o ambos tienen el PN más cercano a 8 casilleros de distancia) se utilizará la forma de conexión para estos casos, debido a que al no utilizarlas podría dificultar o imposibilitar la generación de la posición. Este orden se debe a las posibles conexiones que tienen los distintos PN y la cantidad de conexiones asociadas a cada uno.²

1 **Función** unirPNLejanosMet4(*T*: Posición de ajedrez solitario, *e*: Arista):

2 **Si** PN_a y PN_b están ubicados en los bordes o ambos tienen al nodo más cercano a 8 casilleros de distancia **entonces**

3 Ubicar un PA en los casilleros $(x_a - 1, y_a + 1)$, $(x_b + 2, y_b + 2)$ y $(x_a - 1 + \lceil (x_b + 3 - x_a)/2 \rceil, y_a + 1 + \lceil (x_b + 3 - x_a)/2 \rceil)$

4 **Si no si** PN_a tiene el nodo más cercano a una distancia de 8 casilleros hacia PN_b y PN_b tiene un PN a 4 casilleros de distancia **entonces**

5 Ubicar un PA en los casilleros $(x_a + 1, y_a + 1)$ y $(x_j + 1 + \lceil (x_k - x_j - 1)/2 \rceil, y_j + 1 - \lfloor (x_k - x_j + 1)/2 \rfloor)$

6 **Si no si** PN_b tiene el nodo más cercano a una distancia de 8 casilleros hacia PN_a y PN_a tiene un PN a 4 casilleros de distancia **entonces**

7 Ubicar un PA en los casilleros $(x_b - 1, y_b + 1)$ y $(x_k - 1 - \lceil (x_k - x_j - 1)/2 \rceil, y_k + 1 - \lceil (x_k - x_j - 1)/2 \rceil)$

8 **Si no**

9 Ubicar un PA en los casilleros $(x_a - 1, y_a - 1)$ y $(x_j - 1 + \lfloor (x_k - x_j + 1)/2 \rfloor, y - (x_k - x_j)/2 - 1)$ o $(x_b + 1, y_b + 1)$ y $(x_j + \lceil (x_k + 1 - x_j)/2 \rceil, y - (x_k - x_j)/2 - 1)$ según sea posible.

10 **FinSi**

1 **Función** agregarPLMet4(*T*: Posición de ajedrez solitario):

2 $x_{min_x} \leftarrow y_{P_{max_x}}$ con P_{max_y} un PA o PN con el mayor valor en la coordenada *y*

3 $x_{max_x} \leftarrow x_{P_{min_x}}$ con P_{min_y} un PA o PN con el menor valor en la coordenada *y*

4 **Para todo** PA_a

5 **Si** no hay PLP en los casilleros $(x_{min_y} - 1, y_a)$ y $(x_{min_y} + 1, y_a)$ **entonces**

6 Ubicar PLP en las casillas mencionadas

7 **FinSi**

8 **Fin**

9 Reemplazar una PLP por una PLT

²Dados PN_1, PN_2, PN_3, PN_4 PN donde PN_2 y PN_3 están a 4 casilleros de PN_1 y PN_4 está a 8 casilleros, si establecemos las conexiones de PN_2 y PN_3 a PN_1 como si estuvieran a más casilleros de distancia no podríamos establecer la conexión de PN_4 con PN_1 debido a que se utilizó las dos maneras de existentes para conectar PN “lejanos”.

Algoritmo 4: Triángulos

```

1 Función GenerarTableroMet4(G: Grafo casi 3-regular):
2   Variables: Conjunto vacío C y una pila vacía P
3   Salida: Posición de ajedrez solitario T
4   Ubicar un nodo de G en el tablero en la casilla  $(x_0, y_0)$ 
5   Ubicar en P todas las aristas del nodo
6   Mientras haya aristas  $A \Leftrightarrow B$  en P y siendo  $PN_a$  y  $PN_b$  los PN
7     correspondiente a los nodos A y B respectivamente hacer
8     | Si  $PN_a$  es de grado 2 o  $PN_b$  está ubicado en la posición entonces
9     |   | Agregar  $A \Leftrightarrow B$  a C
10    | FinSi
11    | Si  $PN_b$  no está ubicado en la posición entonces
12    |   | Ubicar  $PN_b$  en la casilla  $(x_a + 4, y_a)$  y conectar mediante un PA en la
13    |   | casilla  $(x_a + 2, y_a + 2)$  o ubicar  $PN_b$  en la casilla  $(x_a - 4, y_a)$  y
14    |   | conectar mediante un PA en la casilla  $(x_a + 2, y_a + 2)$ 
15    | FinSi
16    | Fin
17    | Por cada nodo que no esté en la posición pero si en el grafo, ubicar los PN a 8
      | casilleros de distancia en la coordenada  $x$  del  $PN_{max_x}$ 
18    | Para toda arista  $A \Leftrightarrow B$  en C y siendo  $PN_a$  y  $PN_b$  los PN correspondiente a
      | los nodos A y B y  $x_{PN_a} < x_{PN_b}$  respectivamente conectar los PN del
      | siguiente modo:
19    |   | Se establecerán las conexiones con la siguiente prioridad:
20    |   |   1. Conexiones donde ambos PN están ubicados en los bordes. Es decir, tienen un PN
      |   |     a 2 casilleros de distancia de un lado pero no del otro.
21    |   |   2. Conexiones que involucren un PN ubicado en un borde y un PN a 4 casilleros de
      |   |     distancia al PN más cercano.
22    |   |   3. Conexiones donde hay al menos un PN a 4 casilleros de distancia del PN más
      |   |     cercano.
23    |   |   4. Conexiones restantes.
24    |   | unirPNLejanosMet4(T, e)
25    | Fin
26    | agregarPLMet4(T)
27    | Retornar T

```

Siguiendo estos pasos toda arista del grafo puede ser representada en la posición. Considerando PN_a y PN_b dos PN, esto se debe a lo siguiente:

- Si ambos están a 4 casilleros de distancia es trivial la conexión que se debe establecer.
- Si ambos PN no están ubicado en los bordes, no están a 4 casilleros de distancia y no tienen el PN más cercano en dirección a la pieza con la que se desea establecer la conexión a 8 casilleros de distancia, existen 2 maneras de conectarlos. Esto se debe a que si hay una arista entre ambos en el grafo y se cumplen las condiciones mencionadas, entonces las otras conexiones que tienen ambos PN son con sus vecinos y estos están ubicados a 4 casilleros de distancia con sus respectivos PN.
- Si un PN esta ubicado en un borde izquierdo y otro en el derecho o ambos están a 8

casilleros de distancia del PN más cercano, tienen una manera especial de conexión que no afecta a las otras formas de conexión existentes debido a la ubicación de los PA.

- Si sólo un PN tiene el PN más cercano a 8 casilleros de distancia con dirección al PN al que se desea conectar, se puede establecer la conexión utilizando un PA por arriba de la fila que contiene los PN y no lo afectará a la hora de establecer las otras conexiones debido a la ubicación de los PA. Sin embargo, hay que tener en cuenta que al PN que tiene a otro PN a 4 casilleros de distancia lo afectará como si fuera una conexión a un PN “lejano”.

2.5. Soluciones de las posiciones generadas

Para intentar solucionar el problema del ajedrez solitario en la posición generada por alguno de los métodos se debe realizar los siguientes cambios según el método utilizado. Sea PN_{g2} el PN de grado 2 ubicado en la casilla (x, y) :

- Si se utilizó el método 1, sea PA_1 el PA ubicado en la misma columna que PN_{g2} y PA_2 el PA ubicado en la misma fila que PA_1 , reemplazar PA_2 por la PCN, eliminar a PA_1 y agregar un PA en la ubicación $(x, y + 5)$.
- Si se utilizó el método 2, sea PA_1 el PA ubicado en la misma diagonal principal que PN_{g2} y PA_2 el PA ubicado en la diagonal principal o secundaria que PA_1 , reemplazar PA_2 por la PCN, eliminar PA_1 y agregar un PA en la ubicación $(x + 2, y + 2)$.
- Si se utilizó el método 3 o 4, sea PN_1 un PN vecino de PN_{g2} , e una arista a seleccionar tal que $e : PN_{g2} \Leftrightarrow PN_1$ y PA_1 un PA utilizado para simular e y alineado a PN_1 , los cambios dependerán de la cantidad de PA utilizados para simular la arista seleccionada y el método:
 - Si se utiliza más de un PA, independientemente del método utilizado se debe ubicar la PCN alineada únicamente a PA_1 y PN_1 y eliminar a PA_1 .
 - Si se utiliza sólo un PA y se generó la posición utilizando el método 3, sea PA_2 un PA diferente a PA_1 y alineada vertical u horizontalmente a PN_1 y P un PN o PA alineado vertical u horizontalmente a PA_2 , reemplazar P por la PCN y remover PN_1 y todos los PA alineados a PN_1 excepto PA_1
 - Si se utiliza sólo un PA y se generó la posición utilizando el método 4, sea PA_2 un PA diferente a PA_1 y alineada diagonalmente a PN_1 y P un PN o PA alineado diagonalmente a PA_2 , reemplazar P por la PCN y remover PN_1 y todos los PA alineados a PN_1 excepto PA_1

Al utilizar el método 1 o 2, el PA agregado se necesitará para que la PCN pueda moverse a un casillero donde la PLA o PLT pueda capturarla tras la captura del PN_{g2} por la PCN. Luego, para los métodos 3 y 4, en los casos donde se utiliza más de un PA para simular la arista, sólo se elimina el PA alineado a un vecino de PN_{g2} debido a que la PCN podrá capturar el PN_{g2} y luego un PA correspondiente a una arista. Esto no sucede en el caso donde sólo se utilice un único PA, por lo que se realizarán cambios para simular haber capturado un PN vecino de PN_{g2} , un PA correspondiente a una arista de dicho vecino (cuyos nodos incidentes no incluye a PN_{g2}) y haber llegado a otro PA correspondiente a

esta arista o un PN. Al mismo tiempo, se remueven todos los PA alineados a dicho vecino (a excepción de PA_1) para evitar mantener PA alineados que corresponden a diferentes aristas. Con esto, cuando la PCN capture a PN_{g2} tendrá un PA para capturar y permitirá a la PLA o PLT capturarla.

Siguiendo los pasos para obtener la posición, una PCN ubicada en el casillero de un PN_a puede moverse al casillero donde se ubica PN_b si y sólo si en el grafo existe una arista $A \Leftrightarrow B$. Por lo tanto, si se capturan todos los PN entonces existe un HC en el grafo.

Los cambios mencionados en los párrafos precedentes resultan cruciales. Por un lado, si reemplaza a un PN de grado 3 pueden quedar recorridos inválidos en la posición. Por ejemplo si en una posición generada por el método 1 reemplazamos un PN de grado 3 por la PCN en la casilla (x, y) y procedemos a mover esta pieza para capturar al PA ubicado en la misma columna, en el futuro podríamos capturar los PA en las casillas $(x - 1, y)$ y $(x + 1, y)$ para dirigirnos a un PN vecino del reemplazado, a pesar de que puede no ser vecino del último PN capturado. Por otro lado si reemplazamos al PN de grado 2, la PCN podría terminar en un casillero donde se ubicaba un PA correspondiente a una arista cuyos nodos incidentes no son el PN de grado 2, lo cual encontrariamos una solución de la posición pero no un HC.

El movimiento prematuro por parte de la PL principal (sea alfil o torre) antes de que la PCN capture todos los nodos no otorgará beneficio alguno. Es más, el movimiento de dicha pieza antes de que se capturen todos los PN puede llevar a no encontrar la solución de la posición debido a que esta sólo capturará PA lo cual reducirá los posibles movimientos que pueda hacer la PCN. Por lo tanto, podemos suponer que las PLA o PLT se moverán una vez que sean capturados todos los PN y la PCN esté ubicada en una casilla donde pueda ser capturada por estas.

Al mismo tiempo, la captura de una PL antes de capturar todos los PN dificultaría o imposibilitaría la captura de los PA debido a que estaríamos limitando el recorrido de la PLT o PLA según el método, lo cual también podría llevar a no encontrar una solución de la posición a pesar de su existencia.

A partir de las definiciones y los lemas anteriores podemos enunciar y demostrar el siguiente teorema.

Teorema 2.5.1. El problema del ajedrez solitario en un tablero de $N \times M$ para N y M dados utilizando como piezas sólo torres, alfiles y peones es NP-Completo.

Demostración. Reduciremos el problema del lema 2.0.4 a este. Sea G un grafo casi 3-regular. Utilizaremos cualquiera de los métodos 1 a 4 para construir una posición de ajedrez solitario en base a G en tiempo polinomial. Luego, verificaremos la existencia de una solución para la posición generada. Sabemos que:

- Las conexiones entre los PN se establecen según las aristas de G , por lo que dados PN_a y PN_b , existirá una serie de capturas por parte de la PCN desde uno de estos PN al otro si y sólo si existe un camino entre los nodos A y B en G .
- Los PN sólo pueden ser capturados por la PCN.
- El primer nodo eliminado, ya sea por ser capturado por la PCN o en virtud de las modificaciones para empezar a solucionar la posición generada, es un vecino del PN_{g2} , y el último a capturar será PN_{g2} . Esto se debe a que una vez capturado el PN_{g2} la PCN no podrá llegar a otro PN.

- Las piezas limpiadoras capturarán todos los PA y la PCN, por lo que existirá una solución si la PCN capturó todos los PN.

Por lo tanto, existirá una solución de la posición generada si y sólo si existe un HC en G. Esto se debe a que el recorrido realizado en la posición es replicable en G y viceversa ya que se construye la posición en base al grafo (donde las PL capturarán las piezas que no son PN).

Por último, por el lema 2.0.5 sabemos que el problema del ajedrez solitario está en la categoría de problemas NP y podemos reducir el problema de la existencia de un HC en un grafo casi 3-regular a este, entonces el problema pertenece a la categoría de problemas NP-Completos.

□

2.6. Uso de los algoritmos de BFS/DFS

Tanto en el método 3 como en el método 4 se utilizan los algoritmos de BFS o DFS para ubicar a los PN. Esto se debe a que se busca que la dimensión del tablero sea la menor posible y, al mismo tiempo, por la existencia de una forma de conexión que tiene como condición que ambos PN estén cerca. Si primero ubicamos un nodo y luego buscamos sus nodos adyacentes tendremos más posibilidad de ubicar estos cerca del primero que si ubicamos los nodos de manera aleatoria, lo cual dificultaría conectarlos mediante un único PA. En el caso de los métodos 1 y 2 no se utilizaron los algoritmos de BFS/DFS para ubicar los PN ya que no existía una necesidad de poner nodos adyacentes lo más cerca posible entre ellos para utilizar una forma de conexión en particular, debido a que dichos métodos establecen conexiones que no dependen de distancias entre nodos.

Originalmente la complejidad de los algoritmos es $O(n + m)$, como los grafos considerados son casi 3-regulares estos tienen $7 + 3(n - 5)/2$ aristas con $n \geq 5$, por lo que tendremos $O(n + 3(n - 5)/2 + 7) = O(n)$. Como conclusión, la complejidad de utilizar estos algoritmos es de $O(n)$.

Capítulo 3

Variantes de interés

Los métodos presentados para demostrar que el problema del ajedrez solitario es NP-Completo utilizan sólo peones, alfíles y torres. Esto se debe a dificultades cuando se utilizan otras piezas con una mayor libertad de movimiento como es el caso de la dama o con diferentes patrones de movimientos que no incluyen un alcance ilimitado, como por ejemplo el caballo. En el caso del caballo, se presentará un método para ilustrar las dificultades al utilizarla como PCN.

Además, usando el teorema demostrado previamente se analizará variantes del ajedrez solitario que incluyen reformular ciertas reglas del juego o agregar restricciones sobre el mismo y se intentará demostrar que dichas variantes también pertenecen a la categoría de problemas NP-Completos, por ejemplo remover la necesidad de capturar piezas en cada movimiento pero incluir una cantidad máxima de movimientos permitidos o una versión para 2 jugadores donde uno tiene que capturar todas las piezas mientras que el contrincante debe prevenirlo (“Captura-Prevención”).

3.1. Caballo como PCN

El caballo como PCN da dificultades al generar una posición en base a un grafo debido a su patrón de movimiento. Una manera de representar las aristas cuando se utiliza esta pieza es la siguiente:

- Sean $A \Leftrightarrow B$ una arista y A e B sus nodos incidentes donde A está en la posición pero no B y A no es de grado 4, ubicar un PA tal que el caballo podría capturarlo desde el casillero de PN_a y no exista otra pieza que pueda ser capturada por el caballo desde el nuevo PA . Luego, ubicar a PN_b en la misma fila o columna que PN_a que pueda ser capturado por el caballo desde la casilla del nuevo PA .
- Sean $A \Leftrightarrow B$ una arista y A e B sus nodos incidentes donde ambos están en el tablero, se creará una conexión utilizando el método anterior recursivamente hasta llegar al PN_b . Inicialmente se utilizará el PN_a y un PA y por último, un PA y el PN_b .

Considerando la forma en la que se representarán las aristas, existen dos dificultades al generar la posición:

1. Todo PN que se encuentre rodeado por otras piezas no podrá representar una nueva arista. Esto se debe a que al ubicar nuevos PA provenientes de dicho PN le permitiría

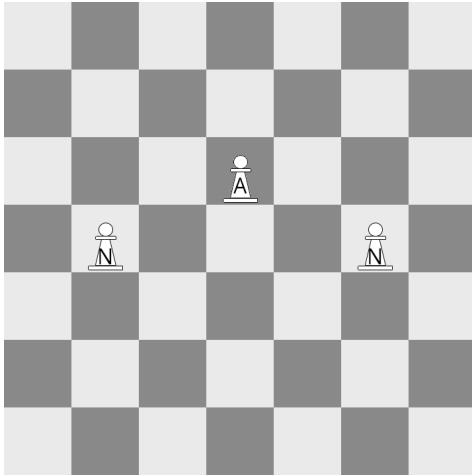


Figura 3.1: Conexión de 2 PN cercanos

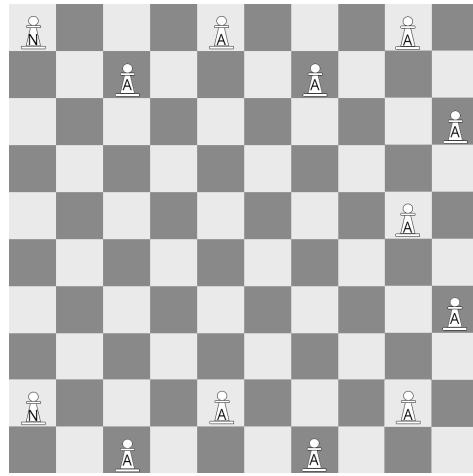


Figura 3.2: Conexión de 2 PN “lejanos”

a la PCN ir de una arista del PN a otra arista de otro PN. Por lo tanto, este método parece ser efectivo únicamente para generar grafos planares.

2. Los PA pueden quedar alineados tanto horizontal, vertical o diagonalmente con los PN. Esto genera dificultades al elegir PL ya que un alfil, una torre o una dama como PL podría ir de un PA a un PN.

Simulando el grafo correspondiente a la Fig. 2.10 obtenemos la posición representada en la Fig. 3.3. Se puede observar que existen diferentes PA en la misma fila/columna/diagonal que distintos PN. Por ejemplo el PN_5 está en la misma fila, diagonal y columna que diferentes PA utilizados para hacer la conexión “lejana” entre PN_3 , PN_4 y PN_5 con PN_2 , esto limita las piezas que se pueden utilizar como PL.

Además, no se pudo representar la conexión entre PN_5 y PN_4 debido a que el primero está encerrado entre PA. Cualquier PA que se agregue conectado a PN_5 establecería una conexión indeseada.

Debido a estos problemas no parece factible utilizar al caballo como PCN con el fin de representar el grafo en una posición. Si bien podríamos usar un algoritmo para detectar si un grafo es planar como por ejemplo el algoritmo de Hopcroft y Tarjan [16] y decidir en base al resultado que método utilizar, aún tendríamos dificultades para elegir las piezas limpiadoras. Aún más, encontrar la representación del grafo en una posición de modo que no se generen conexiones indeseadas puede llegar a ser costoso por la cantidad de posiciones que habría que probar con los PN, el cual puede provocar que a pesar de ser un grafo planar sea menos costoso utilizar otro método. La factibilidad del caballo como PCN puede ser investigada en el futuro. Dicha búsqueda no fue realizada en esta tesis debido a que requeriría utilizar otros métodos para representar grafos en posiciones de ajedrez.

3.2. Dama como PCN

Recordemos que la dama es una pieza cuyo patrón de movimiento es la unión de los patrones de movimiento de la torre y del alfil. Los métodos mencionados anteriormente tienen la particularidad de aprovechar la limitación de movimiento de la PCN para elegir

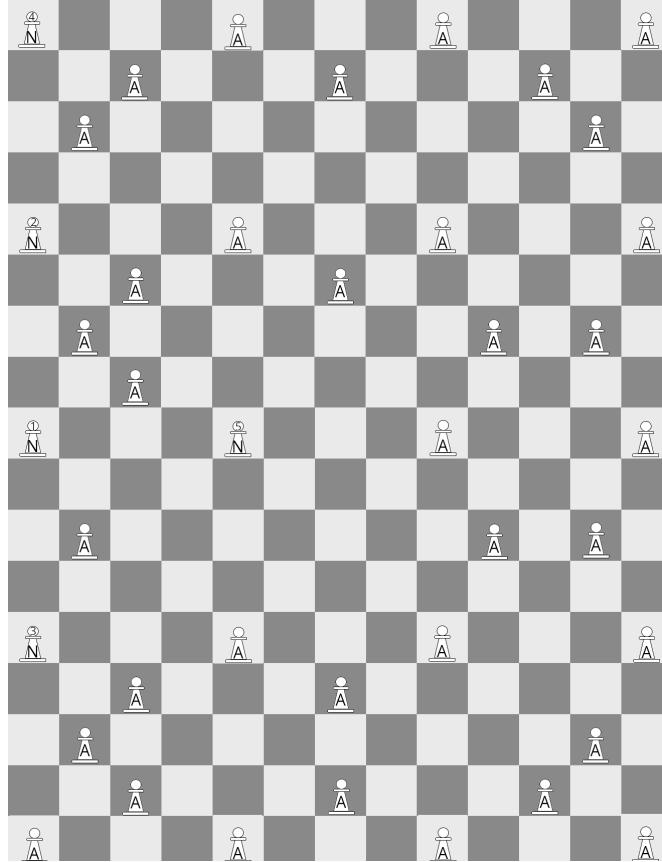


Figura 3.3: Posición generada utilizando un caballo como PCN para el grafo de la Fig. 2.10

la PL principal. Por ejemplo en los métodos 1 y 3 se utiliza una torre como PCN y se utiliza alfiles como PL principal, ya que podemos poner las PLP de tal manera que el alfil puede utilizarlas para realizar un recorrido y capturar todas las piezas restantes mientras que las torres no. Lo mismo ocurre con los métodos 2 y 4 en los que se utiliza el alfil como PCN y la torre como PL principal.

El problema de la dama es la libertad de movimiento que tiene ya que cuando se utiliza dicha pieza como PCN no podemos usar ni alfil ni torre como PL, esto implica que el reemplazo de una PCN por una dama en los métodos mencionados llevaría a una posición inválida, con conexiones de un PN a otro inexistentes en el grafo original.

Para evitar crear estas conexiones inexistentes se podría delegar la responsabilidad de capturar a los PA a estas, reemplazándolas por piezas con un mayor alcance de movimiento, es decir utilizar torres o alfiles como piezas articuladoras en vez de peones lo que nos permitiría hacer movimientos donde una capturaría a otra. Por lo tanto, si tenemos a la dama como PCN a punto de capturar el PN_1 y luego realizaría los movimientos para capturar a PN_2 , podríamos utilizar estas torres o alfiles para capturar todas las piezas que corresponden a las aristas de PN_1 que no se dirigen a PN_2 y luego capturar la pieza que está en la casilla de PN_1 . Entonces, cuando la dama capture la pieza en la posición PN_1 sólo podrá dirigirse al PN_2 y no quedarán piezas articuladoras que estén relacionadas con PN_1 .

Sin embargo, esto también permitiría que una pieza articuladora capture un PN y a un vecino de este. Por lo tanto el orden en el que se capturan los PN no será necesariamente el orden de un HP o HC debido a que una pieza articuladora capturó un PN que puede no ser vecino del último PN capturado por la PCN.

Del ajedrez actual, la única pieza con un patrón de movimiento que no es replicable por la dama es el caballo, pero como no tiene un alcance ilimitado, utilizarlo como pieza limpiadora implicaría utilizar un gran número de PLP, lo cual puede llevar a conexiones indeseadas.

3.3. Alferza como PCN

La alferza puede moverse de a un casillero en diagonal. Esto indica que para moverse requiere una pieza cercana lo que implica que el alfil, la dama o el peón puedan capturar a la PCN. Por otro lado, si utilizáramos torres como piezas articuladoras estos podrían llegar a estar alineadas con otras piezas articuladoras correspondiente a una arista diferente o a un PN, por lo que tendríamos conexiones inválidas al menos que estén en diferentes columnas y filas, pero no representaríamos de esta manera todas las conexiones que tiene un grafo dado que sólo podríamos representar 2 aristas para un PN. Esto se debe a que si un PN tiene 2 PA cercanos y ambos están en diferente fila y columna no se podrá ubicar un tercer PA cercano al PN que no comparta una fila o columna con los PA anteriores. Por lo tanto, esta pieza no debe ser utilizada como PCN por las dificultades mencionadas.

3.4. Elefante en vez de alfil como PCN

Como se mencionó anteriormente, el movimiento del elefante es similar al del alfil, sólo que se mueve de a 2 casilleros. Si reemplazáramos el alfil por el elefante y le agregamos la posibilidad de moverse de a un casillero, nos veríamos forzados a agregar más PA para representar las aristas entre un PN a otro debido a que la ventaja del alfil es su alcance de movimiento diagonal ilimitado.

Sin embargo, este reemplazo podría llevar a posiciones inválidas como se puede ver en las Figs 3.4 y 3.5, debido a que al agregar PA podríamos establecer conexiones inválidas entre PN.

Aquellos PN que están encerrados entre PA no se les podrá agregar una arista debido a que el elefante podría capturar desde un PA utilizado para representar una arista otra PA que se utiliza para representar otra arista. Esta dificultad se presenta tanto para el método 2 como para el método 4. Sin embargo, es posible utilizar al elefante como una PCN en casos en que los grafos son planares, debido a que al generar una posición, no podremos representar 2 aristas que se cruzan porque estableceríamos una conexión indeseada entre 2 peones nodos. Por lo tanto, si existe, podríamos utilizar la representación planar de un grafo y representarla en el tablero. Sin embargo, podemos caer en el mismo caso que al utilizar un caballo como PCN, donde verificar si un grafo es planar y representar dicho grafo puede ser más costoso que usar otro método.

Asimismo, si no modificamos su patrón de movimiento (sólo movimientos en diagonal de a 2 casilleros) igualmente tendríamos algunas de las conexiones inválidas que se generan al utilizar el método 2 y el método 4. En la Fig. 3.5 existen conexiones inválidas en la parte inferior del tablero que existirán usemos o no el movimiento adicional, debido a que tendríamos un PA de un PN conectado directamente a un PN o un PA de un PN conectado

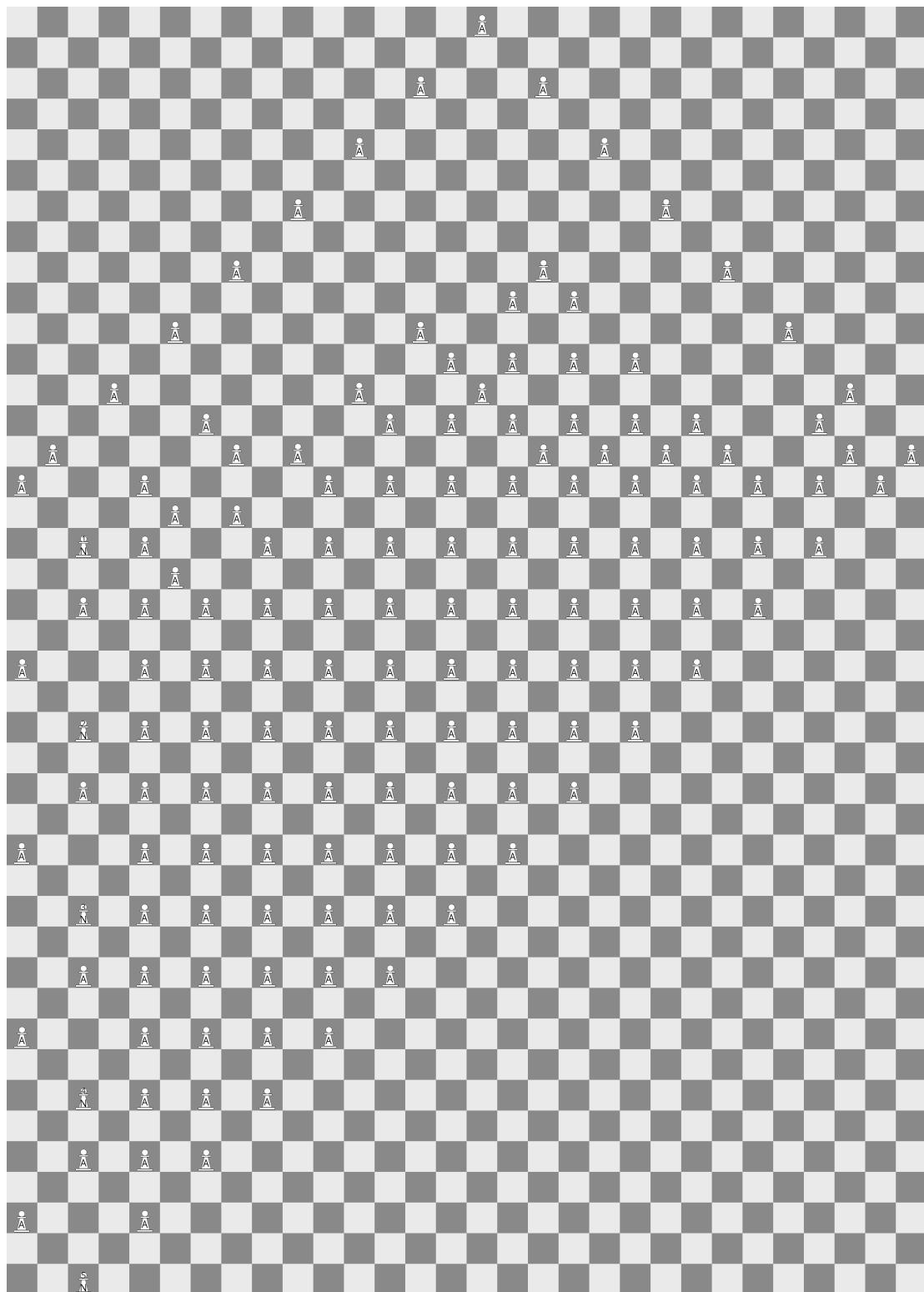


Figura 3.4: Posición generada utilizando la pieza elefante en vez del alfil en el método 2 para el grafo de la Fig. 2.10

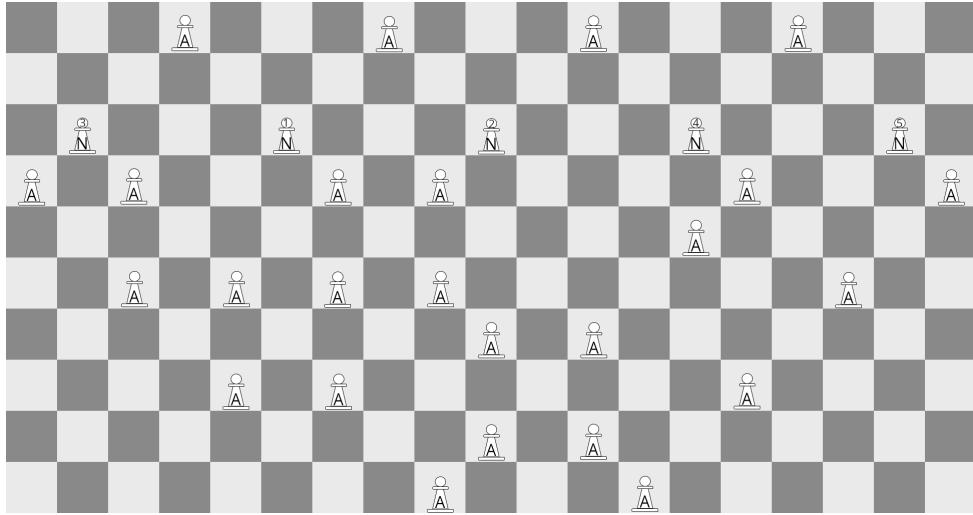


Figura 3.5: Posición generada utilizando la pieza elefante en vez del alfil en el método 4 para el grafo de la Fig. 2.10

a otro PA de otro PN. Esto nos indica que un cambio leve en el patrón de movimiento de la pieza no es suficiente para evitar todas las conexiones inválidas y puede requerir un cambio en la manera que se dan las conexiones entre los diferentes PN.

3.5. Rey como PCN

Si bien su patrón de movimiento es similar al de la alferza (se puede mover en diagonal, horizontal o verticalmente) puede ser utilizada para representar grafos en tableros de ajedrez. La única condición es que dicho grafo tenga una representación planar debido a que al utilizar esta pieza como PCN se debe aumentar la cantidad de PA por su patrón de movimiento. Por lo tanto, aquellos grafos donde se crucen las aristas no podrán ser representadas sin establecer conexiones inválidas. A continuación se enunciará el teorema y un bosquejo de la demostración.

Teorema 3.5.1. El problema del ajedrez solitario utilizando un rey, peones y alfiles en un tablero de $N \times M$ es NP-Completo.

La idea es reducir el problema de la existencia de un HP en un grafo planar [13] y casi 3-regular al problema enunciado. Dada la representación planar del grafo, generaremos una posición donde se respetarán las diferentes columnas y filas en las que se ubican los distintos nodos y el recorrido de las aristas. Por un lado, para ubicar los nodos existentes en la representación habrá un espacio entre las coordenadas horizontales en la posición si ambos están en diferentes columnas y entre las coordenadas verticales si están en diferentes filas. Por otro lado, para representar las aristas se tendrán en cuenta su recorrido y cuales nodos están cerca de estas.

A continuación, seguiremos los siguientes pasos para generar la posición del tablero:

- Recorreremos la representación planar del grafo de arriba hacia abajo ubicando los nodos. Dado 2 nodos N_1 y N_2 donde no existe un nodo intermedio entre ellos tanto en la coordenada x como en la coordenada y , debe haber como mínimo una distancia

de $8 + 8(a - 1)$ casilleros en la primera x y, en caso de que ambos tengan 2 valores diferentes en la coordenada y diferentes, una distancia de $8 + 8(a - 1)$ casilleros en la coordenada y con a la cantidad de aristas que pasan entre N_1 y N_2 . Al mismo tiempo, debe haber esta misma distancia entre un nodo y el borde del tablero. La ubicación de todos los PN deben ser en casilleros del mismo color.

- Para representar las aristas, se trazará un camino lo más fiel posible al existente en la representación planar utilizando PA. Es decir, tanto en la representación planar como en la posición el trayecto será el mismo, incluyendo por donde se extiende la arista y los giros que realiza. Estos PA estarán ubicados en casilleros de diferente color que los PN. Si en la representación planar se encuentran 2 aristas contiguas, en la posición habrá un casillero de distancia entre ambas.
- Por último, para capturar las piezas restantes luego de capturar los PN, se utilizarán una PLP y PLA de manera similar al método 1. Estableceremos 2 diagonales de PLP ubicados de tal manera que un alfil pueda capturar estas y todos los PA. No se podrá capturar un PN debido a que tanto los PN como los PA están en casilleros de diferente color.

A la hora de representar las aristas se debe respetar las siguientes condiciones al conectar los PN. Sea P_0 un PN con ubicación (x_0, y_0) :

- Si se ubica un PA en la ubicación $(x_0 + 1, y_0)$, el siguiente PA más cercano correspondiente a la misma conexión estará en $(x_0 + 2, y_0 - 1)$
- Si se ubica un PA en la ubicación $(x_0 - 1, y_0)$, el siguiente PA más cercano correspondiente a la misma conexión estará en $(x_0 - 2, y_0 + 1)$
- Si se ubica un PA en la ubicación $(x_0, y_0 + 1)$, el siguiente PA más cercano correspondiente a la misma conexión estará en $(x_0 + 1, y_0 + 2)$
- Si se ubica un PA en la ubicación $(x_0, y_0 - 1)$, el siguiente PA más cercano correspondiente a la misma conexión estará en $(x_0 - 1, y_0 - 2)$

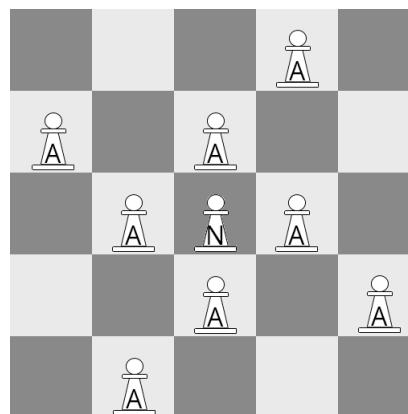


Figura 3.6: Posibles conexiones entrantes hacia un PN

Por último, cada PN sólo puede tener hasta 3 PA adyacentes debido a que al capturar un PN y dirigirse hacia otro, si dejamos 2 de sus PA adyacentes que corresponden a 2

aristas diferentes podríamos utilizarlos para llegar a un vecino del PN capturado (ver la Fig. 3.6).

Considerando esto, la cota superior para las dimensiones del tablero como la cantidad de piezas es polinomial. Esto nos da una reducción polinomial del problema del ajedrez solitario al problema mencionado.

Finalmente, verificaremos las posibles soluciones del problema en tiempo polinomial. Dado un movimiento, se puede verificar en tiempo lineal si dicho movimiento es válido y si captura una pieza. Luego, la secuencia de movimientos será de longitud como máximo igual al ancho de la posición multiplicado por su alto. Como la longitud máxima de la secuencia de movimientos es polinomial y cada movimiento es verificable en tiempo lineal entonces verificar toda la secuencia de movimientos es realizable en tiempo polinomial. Por lo tanto el problema mencionado es NP-Completo.

Es importante aclarar que el orden en el que se capturan los PN no es estrictamente un HP debido a que se puede capturar un PA de una arista e inmediatamente capturar otro PA de otra arista del mismo PN sin necesariamente capturarlo. A pesar de esto, se encuentra una solución a este problema si y sólo si existe un HP en el grafo original.

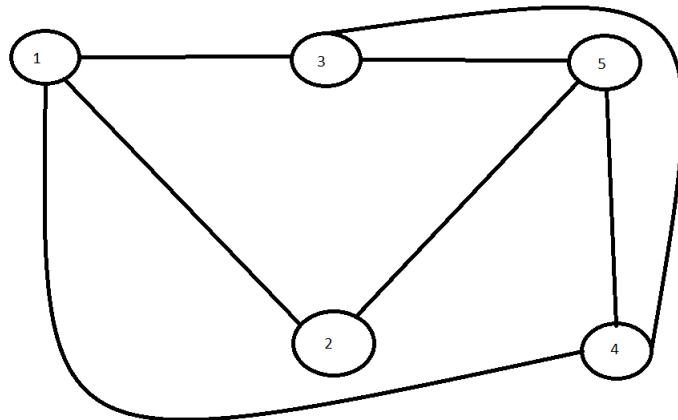


Figura 3.7: Ejemplo de grafo planar casi 3-regular.

3.6. Variantes de ajedrez solitario

El juego tratado en general es conocido y está extendido en distintos ámbitos, y en algunos de ellos se presenta con leves variantes. En esta sección tratamos algunas de las variaciones más naturales de la formulación original. Esto por un lado sirve para mostrar que mantiene la misma complejidad del original tan sólo modificando unas pocas reglas. Por otro lado, esto muestra una aplicación directa de la complejidad ya determinada en la formulación original.

3.6.1. Ajedrez solitario en k movimientos

En esta variante se elimina la restricción de que cada movimiento debe ser una captura, pero se agrega la restricción de que todas las piezas (excepto una) deben ser capturadas a lo largo de sólo k movimientos.

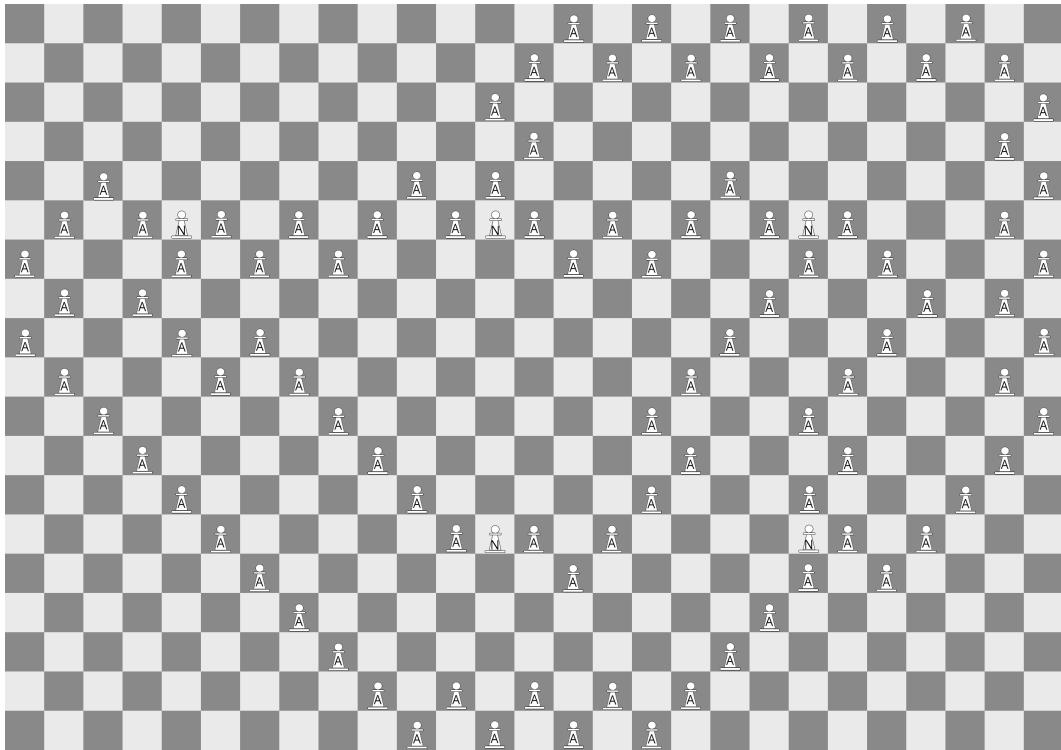


Figura 3.8: Posición generada utilizando el rey como PCN para el grafo de la Fig. 3.7

Considerando las dificultades que existen a la hora de utilizar otras piezas como la dama, el caballo o la alferza, se podrían crear métodos que permiten generar una posición en base a un grafo para dicha variante. Estas posiciones no necesitarán del uso de PL. Sin embargo, hay que tener en cuenta que una solución encontrada para esta variante no implica que tomando el orden en el que se capturen los PN en dicha solución sea un HP o un HC. Además, el valor de k tiene que ser mayor o igual que la cantidad de piezas existentes en la posición, caso contrario no existirá una solución posible.

Para demostrar que el ajedrez solitario en k movimientos es NP-Completo, utilizaremos el teorema demostrado anteriormente:

Teorema 3.6.1. El problema del ajedrez solitario en k movimientos o menos donde $k \geq 0$ en un tablero de $N \times M$ para $N, M \in \mathbb{N}$ es NP-Completo.

*Demuestra*ción. Reducimos el problema del ajedrez solitario al problema enunciado. Para esto, definiremos un valor de $k = \#\text{piezas} - 1$. Si existe una secuencia de movimientos tal que al hacer k movimientos quede en la posición sólo una pieza entonces esa misma serie de movimientos también es solución para el problema del ajedrez solitario en la misma posición. Además, esta serie de movimientos también es solución cuando $k \geq \#\text{piezas} - 1$. Esto nos da una reducción polinomial del problema del ajedrez solitario al problema mencionado.

Por último, demostremos que las posibles soluciones del problema son verificables en un tiempo polinomial. Dado un movimiento, se puede comprobar en tiempo lineal si dicho movimiento es válido y si captura una pieza. Luego, la secuencia de movimientos tiene que tener una longitud menor o igual que k y mayor que la cantidad de piezas en la posición inicial. Por lo tanto, hay que revisar si son válidos k movimientos y si capturan una pieza,

lo cual se puede realizar en tiempo polinomial. Por lo tanto, el problema mencionado es NP-Completo. \square

3.6.2. Ajedrez solitario “Captura-Prevención”

Proponemos aquí una variante que consideramos una formulación natural razonable para dos jugadores. Un jugador debe capturar todas las piezas mientras que su contrincante debe evitarlo. El primero moverá las piezas de color blanco mientras que el segundo moverá las de color negro.

Por un lado, aquel que mueva las piezas de color blanco deberá capturar todas las piezas con el objetivo de que en el tablero sólo quede una pieza sin importar el color de esta. Por otro lado, aquel que mueva las piezas de color negro deberá evitarlo. Cada jugador podrá capturar únicamente las piezas del color de su contrincante. El primer movimiento lo hará el jugador blanco.

Teorema 3.6.2. El problema del ajedrez solitario “Captura-Prevención” en un tablero de $N \times M$ para $N, M \in \mathbb{N}$ dados es NP-Hard.

*Demuestra*ción. Reduciremos el problema de la existencia de un HC en un grafo casi 3-regular al problema mencionado. Para esto, obtenemos una posición de ajedrez solitario utilizando el método Triángulos, realizamos las modificaciones pertinentes para ubicar a la PCN y efectuamos los siguientes cambios:

- Todos los PA, PN y PLP cambiarán a caballos de color negro.
- Para todo caballo negro en la ubicación (x, y) que reemplazará a una PLP se lo ubicará en el casillero $(x - 2, y)$ o $(x + 2, y)$, según si está a la izquierda o a la derecha de los caballos que representan los nodos respectivamente.
- Se agregará una torre de color negro alineada a una cantidad de peones blancos (PB) donde $\#PB = \#\text{piezas} - 2$ por fuera de la posición generada por dicho método y en la misma fila que los nodos. Observar que para todo PB ubicado en la casilla (x, y) no debe haber una pieza en los casilleros $(x + 1, y + 2)$, $(x - 1, y + 2)$, $(x + 1, y - 2)$, $(x - 1, y - 2)$, $(x + 2, y + 1)$, $(x - 2, y + 1)$, $(x + 2, y - 1)$ y $(x + 2, y + 1)$ ya que permitiríamos a un peón ser capturado y bloquear el paso de la torre negra.
- La PLT inicialmente estará ubicada en la fila superior y en la misma columna derecha de los caballos que han sustituido a las PLP .
- En la última fila de caballos que sustituyeron a las PLP existirá un peón negro a 3 casilleros a la derecha del último caballo. Es decir, si C_1 y C_2 son los caballos que reemplazaron a las PLP ubicadas en la fila inferior y $x_{C_1} < x_{C_2}$, la nueva pieza estará en la casilla $(x_{C_2} + 3, y_{C_2})$ y la llamaremos P_{negro} .
- En la fila de los caballos nodos se agregará un PB a la derecha de estos. Si C_i es un caballo que representa un nodo, el PB estará en $(x_{P_{negro}}, y_i)$,

Inicialmente el jugador blanco capturará las piezas mientras que su contrincante estará forzado a capturar los PB alineados a la torre negra. El jugador blanco no podrá utilizar la PLT para capturar su PCN debido a que serán del mismo color, por lo que

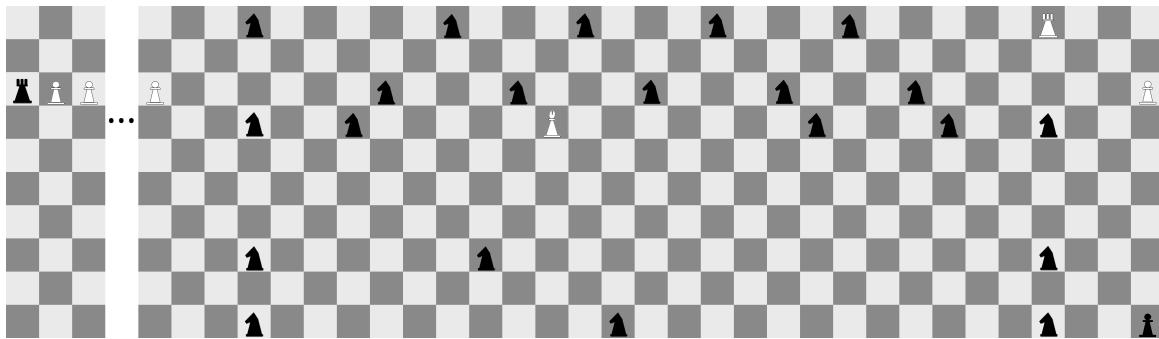


Figura 3.9: Posición generada de la variante “Captura-Prevención” para el grafo de la Fig. 2.10

lo hará forzadamente el jugador negro, dejando esta en la ubicación de un caballo que representa un nodo. Si todos los caballos que representan a algún nodo fueron capturados, los movimientos restantes estarán vinculados con las nuevas piezas agregadas:

1. El jugador blanco capturará con la torre al nuevo peón negro en la casilla $(x_{C_2} + 3, y_{C_2})$.
 2. El jugador negro capturará el PB en la casilla $(x_{P_{negro}}, y_n)$ con la torre negra, siendo y_n la coordenada y de los caballos nodos.
 3. El jugador blanco capturará la torre negra con la PLT ya que ambas están en la misma columna, dejando sólo una pieza en el tablero.

Caso contrario, la torre negra no alcanzará al PB ubicado en la casilla $(x_{P_{negro}}, y_n)$ y, por lo tanto, no existirá una solución.

Con las herramientas vistas y otras con que contamos, no podemos asegurar que la variante mencionada sea un problema NP-Completo, debido a que al haber un segundo jugador se debe comprobar que la solución sea una estrategia ganadora, esto es, los movimientos del blanco son tales que este gane sin importar la defensa del negro y esto último no es previsible que sea polinomial. Al reducir el problema de ajedrez solitario a esta variante se construyó una posición donde el jugador negro siempre tiene un único movimiento válido por turno.

3.6.3. Otras posibilidades

Omitiendo los detalles, se puede probar la NP-Complejidad del problema principal con piezas de un mismo color y otras restricciones, a saber:

- Usando sólo 2 torres, peones (alternativamente, reyes) y 2 alfiles.
 - Usando sólo 2 torres, peones (alternativamente, reyes) y 2 alfiles, con eventualmente más piezas y requiriendo que la pieza final sea una en particular (alfil, torre, peón, rey, caballo o dama).

- Usando sólo 2 torres, peones (alternativamente, reyes) y 2 alfiles, requiriendo que en lugar de 1 pieza final queden en el tablero k piezas y sin posibilidad de más capturas, para $k \geq 1$.

Análogamente se podría reformular la versión con dos colores acoplada a estas restricciones.

Capítulo 4

Complejidad P en el ajedrez solitario

En este capítulo exploramos parte de la frontera P [12] correspondiente al problema general tratado. En este sentido, buscamos diversas restricciones en las posiciones con tal de asegurar que el problema básico de existencia de solución en ajedrez solitario esté en la clase P.

Dado que abordaremos problemas con un sólo tipo de pieza, para las demostraciones siguientes identificamos el grafo asociado a la posición, de la manera en que nos hemos referido anteriormente, en el cual interesarán propiedades básicas de conexidad.

El primero de los casos es el más simple, siguiendo luego otros progresivamente más complejos.

Proposición 4.0.1. El ajedrez solitario con sólo peones en un tablero de $N \times M$ es P.

Demostración. Debido al patrón de movimiento del peón existirá una solución si el grafo asociado a la posición es conexo y “confluente hacia arriba”. Es decir, dado el peón P_{max_y} en el casillero (x_{max_y}, y_{max_y}) no debe haber otro peón con el mismo valor en la coordenada y . Al mismo tiempo para todo peón en el casillero (x, y) con $y < y_{max_y}$ debe existir una serie de movimientos compuestos por capturas tal que la última pieza a capturar es P_{max_y} . Si existen dos peones $P1$ y $P2$ en los casilleros (x_1, y_1) y (x_2, y_2) respectivamente tal que no existe un peón en los casilleros $(x_1 + 1, y_1 + 1)$, $(x_1 - 1, y_1 + 1)$, $(x_2 + 1, y_2 + 1)$ y $(x_2 - 1, y_2 + 1)$, no existirá una solución debido a que toda serie de movimientos terminará con un peón en los casilleros donde se ubican $P1$ y $P2$.

El grafo dirigido asociado a la posición consistirá en un nodo por cada peón y un nodo A tendrá una arista hacia el nodo B si y sólo si el peón B puede capturar a A sin peones intermedios. Luego aplicaremos el algoritmo de BFS iniciando por el nodo que represente a P_{max_y} . Si al aplicar este algoritmo encontramos todos los nodos entonces el grafo asociado es débilmente conexo [17], caso contrario no lo es. Esto se debe a que si encontramos todos los nodos aplicando dicho algoritmo es porque existen caminos del nodo inicial a cada uno de aquellos.

Debido a como se construyó el grafo, si este es conexo y no existen 2 peones en 2 casilleros cuyo valores en la coordenada y es y_{max_y} implica que la posición tiene solución y para todo P en el casillero (x, y) con $y < y_{max_y}$ existen una secuencia de capturas cuya última pieza capturada es la que se ubica en la casilla (x_{max_y}, y_{max_y}) .

Tanto la construcción del grafo como la ejecución del algoritmo BFS sobre este es realizable en tiempo polinomial por lo que el problema enunciado pertenece a la clase P. \square

En esta proposición se puede observar que si se reemplazara los peones por piezas que pueden capturar en diagonal tanto hacia arriba como hacia abajo, entonces interesaría sólo la conexidad y no la confluencia.

Proposición 4.0.2. El ajedrez solitario con sólo torres o sólo caballos en un tablero de $N \times M$ es P. Más precisamente, hay solución si y sólo si el grafo asociado es conexo¹.

Demostración. Primero se demostrará la proposición para el ajedrez solitario con sólo torres y luego para sólo caballos.

El grafo asociado a la posición consistirá en nodos que representarán las torres y un par de nodos estarán conectados mediante una arista si y sólo si las torres que representan pueden ser capturadas la una a la otra².

Primero veremos que si el grafo asociado es conexo si y sólo si existe una solución. Utilizaremos el algoritmo de BFS para determinar si dicho grafo es conexo, luego:

- Si no es conexo entonces la posición no tiene solución debido a que habrá dos conjuntos de torres para los cuales una torre de un conjunto no podrá llegar a los casilleros donde se ubican las torres del otro conjunto.
- Si es conexo podemos tomar un árbol generador correspondiente al grafo [9, 10] y eliminar los nodos de este tal que en ningún momento se desconecte el árbol y hasta que quede sólo uno. Por lo tanto, si el grafo es conexo entonces existe una solución.

Por último para demostrar que dada una posición con solución implica que el grafo asociado es conexo, sabemos que cualquier casillero que contenga una torre puede ser el casillero de la pieza que sobrevive al final de todas las capturas. Como las torres representan los nodos y por la propiedad transitiva, si la torre A puede capturar a la torre B y B puede capturar a C entonces A puede capturar a C luego de capturar a B . Por lo tanto el grafo asociado debe ser conexo.

Tanto la construcción del grafo como la decisión si dicho grafo es conexo o no son calculables en tiempo polinomial por lo que el problema enunciado pertenece a la clase P.

En el caso del ajedrez solitario con sólo caballos la demostración es similar a la anterior. La única diferencia es que para la construcción del grafo dada una pieza se verá el alcance de esta y los nodos representarán a los caballos. Es decir, dada una pieza P_1 en el casillero (x_1, y_1) existirá en el grafo una arista que conecte con el nodo que representa a la pieza P_2 si esta está en el casillero $(x_1 + 1, y_1 + 2), (x_1 + 2, y_1 + 1), (x_1 + 1, y_1 - 2), (x_1 + 2, y_1 - 1), (x_1 - 1, y_1 - 2), (x_1 - 2, y_1 - 1), (x_1 - 1, y_1 + 2)$ o $(x_1 - 2, y_1 + 1)$.³ \square

¹Obsérvese que los caminos de las torres podrían cruzarse entre sí, ya sea con o sin acceso a casillas comunes.

²Esto es, sin torres intermedias que bloqueen el movimiento de la torre que mueve.

³Es claro que no ha de confundirse este último caso con el clásico problema del recorrido del caballo que visita una vez cada casilla del tablero y que consiste esencialmente en hallar un camino hamiltoniano en el grafo asociado usando un único caballo. En nuestro caso, los caballos presentes se capturan los unos a los otros en un grafo donde si el grafo es conexo determina que hay soluciones en el problema de ajedrez solitario.

A la hora de demostrar si una variante de ajedrez solitario con sólo un tipo de pieza tiene solución se puede utilizar el concepto de árbol generador pero únicamente cuando la pieza permite movimientos reversibles. En el caso del ajedrez solitario con sólo peones, podríamos generar un grafo de la misma manera que se hizo en la proposición 4.0.2 el cual, si bien existe un árbol generador, no implicaría que la posición tenga solución. Por ejemplo si en una posición existen 3 peones en los casilleros (x, y_{max}) , $(x+2, y_{max})$ y $(x+1, y_{max}-1)$ el grafo asociado tendrá un árbol generador pero la posición no tiene solución.

Diremos que una pieza es reversible si cada vez que pueda moverse de un casillero a otro, podrá también volver del segundo al primero. Es decir, sea P una pieza en el casillero (x_1, y_1) , se dice que es reversible si cuando puede mover de (x_1, y_1) a (x_2, y_2) también puede mover de (x_2, y_2) a (x_1, y_1) .

Lema 4.0.3. Sea P un tipo de pieza, y sea R un tipo de pieza reversible cuyo patrón de movimiento incluye el de captura de P (por ejemplo $R =$ rey, alfil o dama, y $P =$ peón). Consideremos posiciones con sólo piezas de tipo R de modo que su grafo asociado es conexo, y sea C un casillero ocupado por una de estas R . Entonces existe solución del problema de ajedrez solitario en un tablero de $N \times M$ para la cual la posición final es una R en C , y esta solución puede obtenerse en tiempo polinomial en el número de piezas.

Demostración. El grafo asociado tendrá una cantidad de nodos igual a cantidad de piezas existentes en la posición. Al mismo tiempo para todo par de nodos existirá una arista si y sólo si las piezas que representan pueden capturarse la una a la otra.

Consideremos un árbol generador del grafo asociado, que es calculable en tiempo polinomial, e ignoremos todo movimiento que involucren aristas que no estén en dicho árbol. Sea p la cantidad de piezas en la posición:

- Si $p = 1$ es trivial.
- Asumiendo que la proposición es válida para $p = n$ se demostrará que es verdadera para $p = n + 1$. Para esto se debe considerar como primera jugada una captura realizada por una pieza que no esté ubicada en C y sea una hoja en el árbol generador. Este movimiento existe ya que para todo árbol no trivial existe al menos 2 hojas. Luego de realizar esta captura podemos aplicar la hipótesis inductiva sobre la posición.

Se puede obtener un árbol generador en tiempo polinomial [9, 10] sobre la cantidad de piezas. Al mismo tiempo, se puede realizar la captura de una pieza que represente una hoja en tiempo lineal en función de la cantidad de piezas por lo que se puede obtener la solución en tiempo polinomial en función de la cantidad de piezas.

□

Proposición 4.0.4. Sea P un tipo de pieza, y R un tipo de pieza reversible cuyo patrón de movimiento incluye el de captura de P . Consideremos posiciones con sólo piezas de tipo R , con grafo asociado conexo, más una pieza P en alguna casilla adicional con respecto al grafo. La existencia de solución al problema de ajedrez solitario en un tablero de $N \times M$ en tales posiciones puede determinarse en tiempo polinomial.

Demostración. Existen dos posibles casos:

- Si la pieza de tipo P no puede capturar ninguna de las piezas de tipo R, ni puede ser capturada por algunas de estas, entonces la posición no tiene solución ya que finalmente quedará la pieza de tipo P y al menos una de las piezas de tipo R. Esta condición puede ser verificable en tiempo polinomial en N y M .
- Si la pieza de tipo P puede ser capturada por una de las piezas de tipo R, sea C la casilla que contiene la pieza de tipo P y considerar la posición que resulta de reemplazar la pieza de tipo P por una pieza de tipo R. Esta posición se resuelve por el lema anterior concluyendo con una pieza de tipo R en C. Es decir, sin la necesidad de mover la pieza que reemplazó a la de tipo P.
- No existen otros casos ya que si la pieza de tipo P puede capturar una de las piezas de tipo R, esta podría capturar a la primera por ser reversible.

□

Proposición 4.0.5. Sea P un tipo de pieza, y R un tipo de pieza reversible cuyo patrón de movimiento incluye el de captura de P. Consideremos posiciones con sólo piezas de tipo R al menos una, con grafo asociado desconexo, más una pieza P en alguna casilla adicional con respecto al grafo. La existencia de solución al problema de ajedrez solitario en tales posiciones en un tablero de $N \times M$ puede determinarse en tiempo polinomial.

Demostración. Si el reemplazo de una P por una R vuelve al grafo conexo entonces, por el lema anterior, el problema tiene solución que se consigue con una pieza del tipo R final en la casilla en que se consideró el reemplazo. De otro modo, no existe solución ya que quedará la pieza del tipo P y al menos una de las piezas de tipo R. □

Dado que determinar la conexidad de un grafo es polinomial [9, 10], juntando ambas proposiciones se obtiene:

Teorema 4.0.6. Sea P un tipo de pieza y R un tipo de pieza reversible cuyo patrón de movimiento incluye el movimiento de captura de P. Consideremos posiciones con 0 o más piezas de tipo R y no más que 1 pieza de tipo P. El problema de ajedrez solitario en un tablero de $N \times M$ restringido a estas posiciones es polinomial.

Observemos que si el número de P es mayor que 1, como el caso de reyes con peones intercalados, la situación se complica, como muestran tan sólo las distintas posibilidades de un rey adyacente a 2 peones. También observemos que esto incluye tomar R simplemente como cualquier pieza reversible, del ajedrez actual o no (ya que podemos ignorar la hipótesis de P si tomásemos P como una “pieza nula”, i.e. que no mueve).

Capítulo 5

Conclusiones

A lo largo de este trabajo se exploraron distintas maneras de representar cierta familia de grafos mediante posiciones de ajedrez solitario, utilizando diferentes piezas con distintos grados de libertad en el movimiento y se demostró que el problema de determinar la existencia de una solución del ajedrez solitario utilizando peones, alfiles y torres es NP-Completo como también lo es cuando se utiliza peones, reyes y alfiles. La principal dificultad de esta tarea se encontró en cómo representar las aristas y cómo eliminar las piezas que se utilizaban para representarlas sin establecer caminos entre PN que no existan en el grafo.

Los diferentes métodos tienen determinadas ventajas o desventajas con respecto a la complejidad del algoritmo, los tamaños de la posición y la cantidad de piezas utilizadas. Hemos usado especialmente los grafos casi 3-regulares para lograr una transformación adecuada de las instancias positivas y negativas de los problemas. Los métodos explorados si bien a priori permitirían representar grafos 3-regulares, no podrán hacerlo con el objetivo buscado, a la vez que no permiten representar aquellos que tengan nodos de grado mayor que 3 teniendo en cuenta las funciones de representación usadas.

El método 2 requiere una menor cantidad de piezas y genera un tablero de menor tamaño a comparación del método 1. Las dimensiones del tablero del método 1 crecen polinomialmente a la hora de ubicar las piezas limpiadoras debido a que como la PL principal es el alfil se debe agrandar el tablero tanto horizontal como verticalmente y debe evitar que alguna de estas piezas agregadas puedan ser capturadas por la PCN y establecer conexiones inválidas. En el método 2 también se agranda en ambos sentidos cuando se ubica las piezas mencionadas, pero su crecimiento no sólo es polinomial sino constante debido a que por un lado no aumenta la cantidad de columnas y por otro lado sólo se agrega una fila para ubicar 2 PLP. Esto se debe a que el movimiento de la torre no incluye moverse a través de las filas y columnas al mismo tiempo debido a su patrón de movimiento y por la dificultad de establecer una conexión inválida cuando la PCN es un alfil. Esto mismo ocurre comparando el método 3 y el método 4 donde el primero utiliza una torre como PCN y el segundo un alfil. Por lo tanto, podríamos concluir que la elección de la PL principal y la PCN influirá en las dimensiones del tablero a la hora de posicionar las PL.

Al mismo tiempo, la PL principal (es decir, PLT y PLA) varía según la PCN, debido a que es necesario que la PCN no pueda llegar a las PLP ya que estas son utilizadas por la PL principal para capturar a todos los PA. Si ambos tuvieran el mismo patrón de movimiento entonces tanto el PCN como la PL principal podrían llegar a las PL y los PN. En la

búsqueda de una solución de la posición, no sólo agregaría más opciones de movimientos y posibles posiciones sino que además podríamos encontrar soluciones inválidas donde el orden en el que se capturan los PN no es válido como un recorrido en el grafo original. Por lo tanto, la elección de una PCN o PL principal influirá fuertemente en el otro.

Si bien se necesita que los patrones de movimiento entre ambos tipos de piezas sean diferentes, también se requiere que el patrón de movimiento de uno no contenga al otro. Elegir a una dama como PCN tendría como consecuencia que no podamos utilizar una torre o un alfil como PL principal ya que su patrón de movimiento es la unión de ambas. Esta libertad en el movimiento dejaría como único candidato para ser una PL principal al caballo, el cual se consideró como inviable por la cantidad de PLP a agregar y la posible formación de conexiones indeseadas. Aunque el patrón de movimiento de uno no está contenido en el otro, la libertad de la dama para moverse complicaría la ubicación de las PL.

A pesar de esto, una baja libertad de movimiento en las PCN también implica complicaciones al generar una posición. Al elegir piezas que no tengan un alcance ilimitado en el movimiento, se necesitaría ubicar más PA para representar una arista entre 2 nodos en la posición. Esto produciría que haya PN que no puedan representar una arista nueva ya que, al estar encerradas entre PA, establecerían conexiones no presentes en el grafo original. Los únicos grafos que podríamos representar serán los grafos planares donde las aristas no se cruzan como se demostró al utilizar el rey como PCN.

Como trabajo futuro se podría buscar una manera de representar grafos con nodos de grado mayor que 3 o grafos 3-regulares y buscar mejoras para los métodos con el fin de reducir tanto las dimensiones del tablero como la cantidad de piezas a utilizar. Además, se podría buscar métodos para crear posiciones en base a grafos que usen como PCN damas o caballos para las variantes de ajedrez solitario mencionadas.

También se puede buscar variantes que utilicen piezas de color blanco y piezas de color negro pero que, a diferencia de la variante “Captura-Prevención”, sólo haya un único jugador y aplicando la regla de que cada pieza pueda capturar sólo piezas del otro color, o variantes en las que haya más de una PCN o donde se permita a una sola pieza capturar otras.

Otra línea de investigación posible consiste en explorar con mayor profundidad la frontera entre los problemas polinomiales y NP-completos para las variantes de ajedrez solitario, teniendo en cuenta las piezas usadas y su movilidad. Por ejemplo, una posición donde las únicas piezas existentes son reyes es resoluble en tiempo polinomial debido a que basta con determinar si el grafo es conexo. Esta frontera puede abordarse desde distintas perspectivas, como por ejemplo:

- Las piezas usadas.
- Las piezas que tienen permitido capturar y las que no lo tienen.
- Los colores de las piezas.

A lo largo de los métodos de representación usados, puede ser interesante la búsqueda de cotas superiores más bajas para el ancho y el alto del tablero con piezas limpiadoras, en particular para el método 3. Finalmente, también interesa el estudio de posibles demostraciones o refutaciones de que la variante “Captura-Prevención” es un problema NP-Completo, y una posible vinculación con la situación de otros juegos para dos jugadores de características similares.

Apéndice A

Información de los métodos

Cuadro A.1: Dimensiones del tablero sin PL

Método utilizado	Ancho del tablero	Alto del tablero
Arcos	$O(3n - 1) = O(n)$	$O(15n - 9/2) = O(n)$
Caminos diagonales	$O(6n + 2) = O(n)$	$O(9n + 3) = O(n)$
Caminos rectos	$O(10n) = O(n)$	$O(10n) = O(n)$
Triángulos	$O(8(n - 1) + 3) = O(n)$	$O(4(n - 1) + \lceil(8(n - 1) + 3)/2\rceil) = O(n)$

Cuadro A.2: Dimensiones del tablero con PL

Método utilizado	Ancho del tablero	Alto del tablero
Arcos	$O(33/2n - 11/2) = O(n)$	$O(33/2n + 19/2) = O(n)$
Caminos diagonales	$O(6n + 3) = O(n)$	$O(9n + 4) = O(n)$
Caminos rectos	$O(28n - 9) = O(n)$	$O(28n - 9) = O(n)$
Triángulos	$O(8(n - 1) + 5) = O(n)$	$O(4(n - 1) + \lceil(8(n - 1) + 3)/2\rceil) = O(n)$

Cuadro A.3: Cantidad de piezas a utilizar

Método utilizado	Cantidad de piezas sin PL	Cantidad de piezas con PL
Arcos	$O(6n - 2) = O(n)$	$O(16n - 2) = O(n)$
Caminos diagonales	$O(6n - 2) = O(n)$	$O(9n) = O(n)$
Caminos rectos	$O(11/2n - 3/2) = O(n)$	$O(29/2n - 7/2) = O(n)$
Triángulos	$O(11/2n - 3/2) = O(n)$	$O(17/2n + 5/2) = O(n)$

Cuadro A.4: Piezas utilizadas

Método utilizado	PCN	PL
Arcos	Torre	Alfiles, torres y peones
Caminos diagonales	Alfil	Torres y peones
Caminos rectos	Torre	Alfiles y peones
Triángulos	Alfil	Torres y peones

Bibliografía

- [1] <https://www.casi.com.ar/sites/default/files/Reglamento%20Ajedrez.pdf>
- [2] <https://archive.org/details/AHistoryOfChess>
- [3] <https://users.dcc.uchile.cl/~jegger/memoria/node12.html>
- [4] [https://en.wikipedia.org/wiki/Deep_Blue_\(chess_computer\)](https://en.wikipedia.org/wiki/Deep_Blue_(chess_computer))
- [5] [https://en.wikipedia.org/wiki/Deep Thought_\(chess_computer\)](https://en.wikipedia.org/wiki/Deep Thought_(chess_computer))
- [6] <http://c3ar.blogspot.com/search?q=ajedrez+solitario>
- [7] <https://www.chess.com/es/solo-chess>
- [8] <https://www.thinkfun.com/products/brain-fitness-solitaire-chess>
- [9] R. Diestel, *Graph Theory*, Springer-Verlag, 1997.
- [10] F. Harary, *Graph Theory*, Addison-Wesley, 1969.
- [11] J.A. Bondy, U.S.R. Murty, *Graph Theory with Applications*, Macmillan, 1978.
- [12] M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, 1979.
- [13] C. Picouleau, *Complexity of the hamiltonian cycle in regular graph problem*, *Theoretical Computer Science*, 463-473, 1994.
- [14] N.R. Aravind, N. Misra, H. Mittal, *Chess is Hard even for a Single Player*, 2022. <https://arxiv.org/pdf/2203.14864>
- [15] J. Brunner, L. Chung, M. Coulombe, E.D. Demaine, T. Gomez, J. Lynch, *Complexity of Solo Chess with Unlimited Moves*, 2023. <https://arxiv.org/pdf/2302.01405>
- [16] J. Hopcroft and R. E. Tarjan, *Efficient planarity testing*. ACM, 549–568, 1974.
- [17] https://es.wikipedia.org/wiki/Grafo_conexo