



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

# Problema de dominación eterna para grafos de intervalos

Tesis de Licenciatura en Ciencias de la Computación

Martín Rinemberg

Director: Dr. Francisco J. Soullignac  
Buenos Aires, 2019

## PROBLEMA DE DOMINACIÓN ETERNA PARA GRAFOS DE INTERVALOS

Consideremos un juego por turnos sobre un grafo  $G = (V, E)$  jugado por un atacante y un defensor. Inicialmente, el defensor ubica guardias sobre los vértices de  $G$ . En cada turno, el atacante elige un vértice  $v$  para *atacar* y el defensor mueve algunos guardias, a vértices adyacentes a sus posiciones actuales, a fin de ubicar al menos un guardia en  $v$ . Los *problemas de dominación eterna* consisten en determinar la mínima cantidad de guardias con las que se puede defender eternamente a  $G$ . En particular,  $\gamma_{n,1}^\infty(G)$  y  $\gamma_{n,n}^\infty(G)$  denotan la cantidad mínima de guardias necesarios cuando no y sí se pueden posicionar guardias simultáneamente sobre el mismo vértice, respectivamente. En esta tesis demostramos que si  $G$  es un grafo de intervalos, entonces  $\gamma_{n,1}^\infty(G) = \gamma_{n,n}^\infty(G) = \theta_c(G)$ , donde  $\theta_c(G)$  es el número de cubrimiento clique-conexo de  $G$ . Más aún, damos un algoritmo lineal para computar un conjunto dominante eterno de mínima cardinalidad.

**Palabras claves:** problema de dominación eterna, grafos de intervalos, grafos de intervalos propios, número de cubrimiento clique-conexo, estrategia de defensa.

## THE ETERNAL DOMINATING PROBLEM FOR INTERVAL GRAPHS

Consider a game that is played by an attacker and a defender on a graph  $G = (V, E)$ . Initially, the defender places some guards on the vertices of  $G$ . In each turn, the attacker chooses a vertex  $v$  to attack, while the defender moves some of its guards, to neighbors of their current positions, with the aim of positioning at least one guard on  $v$ . The *eternal domination problems* require to find the minimum number of guards needed to eternally defend  $G$ . In particular,  $\gamma_{n,1}^\infty(G)$  and  $\gamma_{n,n}^\infty(G)$  denote the minimum number of guards needed when the defender is forbidden and allowed to move more than one guard to the same vertex, respectively. In this thesis we prove that  $\gamma_{n,1}^\infty(G) = \gamma_{n,n}^\infty(G) = \theta_c(G)$  for every interval graph  $G$ , where  $\theta_c(G)$  is the clique-connected cover number of  $G$ . Furthermore, we design a linear algorithm to compute an eternal dominating set of minimum cardinality.

**Keywords:** eternal domination problems, interval graphs, proper interval graphs, clique-connected cover number, defense strategy.

## AGRADECIMIENTOS

Me gustaría agradecerle en primer lugar a Francisco, mi director de tesis, que me ayudó un montón e invirtió mucho de su tiempo para que yo llegara hoy aquí.

A la FCEN de la UBA y todos los profesores que tuve que me proveyeron de una herramienta esencial, el conocimiento.

A la mejor profesora que tuve en el secundario Clara Freud que me recomendó la carrera.

A mi gran amigo y compañero de cursada Browar, con quien pasé largas jornadas de estudio que siempre incluían mucha comida y una cuota infaltable de diversión.

A los Tutemen que se bancaron que no fuera a varias salidas por tener que estudiar.

A mis padres que con la mejor intención no pararon nunca de insistirme en que terminara la carrera y me brindaron todo su apoyo.

A mi hermana Maga que me acompañó y aconsejó en este largo proceso.

Y por último, a mi prometida Den que me motivó e insistió para que me recibiera y estuvo conmigo tardes enteras sólo para asegurarse que me concentrara en hacer mi tesis.

## Índice general

1.. Introducción . . . . .	1
1.1. Preliminares y cotas elementales . . . . .	5
2.. Dominación eterna en grafos de intervalos . . . . .	9
2.1. Dominación eterna en grafos de intervalos propios . . . . .	9
2.2. Dominación eterna en grafos de intervalos generales . . . . .	11
2.3. Algoritmo para calcular un conjunto dominante $(n, y)$ -eterno . . . . .	13
3.. Conclusiones . . . . .	17

## 1. INTRODUCCIÓN

En esta tesis estudiamos un problema de dominación eterna sobre grafos de intervalos. Para definir el problema, consideremos un juego por turnos que se juega sobre un grafo  $G = (V, E)$  y consta de dos jugadores: el *atacante* y el *defensor*. Inicialmente, el defensor ubica un conjunto de *guardias* sobre los vértices de  $G$ . En cada turno, el atacante comienza eligiendo un vértice  $v$  para *atacar*. Posteriormente, el defensor mueve los guardias a fin de posicionar al menos un guardia sobre  $v$ . Si el defensor logra su objetivo, entonces decimos que el ataque es *repelido*. Los guardias se mueven de un vértice a otro a través de las aristas de  $G$ . Específicamente, si un guardia se encuentra sobre un vértice  $u$ , entonces el defensor o bien deja al guardia en  $u$  o bien lo mueve hacia un vecino de  $u$ . El atacante gana el juego si y sólo si el defensor no puede repeler alguno de sus ataques. Equivalentemente, el defensor gana si logra defender eternamente a  $G$  de cualquier ataque. Los *problemas de dominación eterna* consisten en determinar, para un grafo  $G$ , la mínima cantidad de guardias con las que se puede defender eternamente a  $G$ . Es fácil ver que este valor está bien definido, porque alcanza con ubicar  $n = |V|$  guardias inamovibles, uno en cada vértice.

La razón por la que nos referimos a los problemas de dominación eterna en plural es debido a que podemos definir distintas variantes considerando distintas restricciones para los movimientos de los guardias. Por ejemplo, si se permite que se mueva uno o todos los guardias por turno, o si se restringe la cantidad de guardias que pueden ocupar un vértice al mismo tiempo. Sea  $x$  la cantidad de guardias que pueden moverse en cada turno e  $y$  a la cantidad de guardias que pueden ocupar un vértice simultáneamente. Los valores  $x$  e  $y$  son los *parámetros* del problema; llamamos  $(x, y)$ -juego al juego cuyos parámetros son  $x$  e  $y$ . Denotamos entonces  $\gamma_{x,y}^\infty(G)$  a la menor cantidad de guardias necesarios para defender eternamente a  $G$  en un  $(x, y)$ -juego. A cada multiconjunto  $D$  de vértices tales que  $G$  se puede defender eternamente con  $\gamma_{x,y}^\infty(G)$  guardias, poniendo un guardia en cada  $v \in D$  con la multiplicidad correspondiente, se lo llama *conjunto dominante  $(x, y)$ -eterno*.<sup>1</sup> Vale notar que tanto  $x$  como  $y$  pueden ser mayores a  $\gamma_{x,y}^\infty(G)$ ; en general,  $x$  e  $y$  pueden ser mayores a la cantidad de guardias con las que el defensor decida jugar. Sin embargo, dado que  $\gamma_{x,y}^\infty(G) \leq n$  para todo grafo  $G$  con  $n$  vértices, vamos a restringirnos a considerar que  $x \leq n$  e  $y \leq n$ . Bajo esta hipótesis, el defensor puede mover todos los guardias en cada turno si  $x = n$  y no hay restricción en la cantidad de guardias que pueden ocupar el mismo vértice simultáneamente si  $y = n$ . La Figura 1.1 muestra tres turnos de un  $(3, 2)$ -juego.

El valor  $\gamma_{x,y}^\infty(G)$  de un grafo  $G = (V, E)$  varía de acuerdo a qué valores de  $x$  e  $y$  se consideran. Tomemos, por ejemplo, el grafo  $C_6$  formado por un ciclo  $v_0, \dots, v_5$  de 6 vértices (Figura 1.2). Notemos que  $\gamma_{1,6}^\infty(C_6) > 2 = \gamma_{2,1}^\infty(C_6)$ , i.e., necesitamos al menos tres guardias para defender a  $C_6$  cuando sólo se puede mover un guardia por turno, mientras que alcanza con dos guardias cuando ambos se pueden mover a la vez. En efecto, cuando hay dos guardias y estos ocupan vértices  $v_i$  y  $v_j$  con  $j \in \{i, i+1, i+2\}$  (Figura 1.2 izquierda), entonces el atacante gana eligiendo  $v_{i+4}$  cualquiera sean los parámetros  $x$  e  $y$  del juego. (Todas las cuentas en este ejemplo son módulo 6.) En el caso del  $(1, 6)$ -juego, el atacante gana en dos turnos cuando los guardias ocupan los vértices  $v_i$  y  $v_{i+3}$ : primero ataca  $v_{i+2}$  y luego a  $v_{i+4}$ . Si, en cambio, el defensor puede mover los dos guardias, entonces cuando el atacante ataca  $v_{i\pm 1}$  alcanza con que el defensor mueva los guardias a  $v_{i\pm 1}$  y  $v_{i+3\pm 1}$ . De esta

<sup>1</sup> Usamos el término multiconjunto en lugar de conjunto por simplicidad.

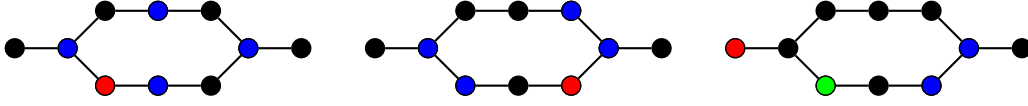


Figura 1.1: Ejemplo de tres turnos de un  $(3, 2)$ -juego sobre un grafo  $G$  que se muestran de izquierda a derecha. Los vértices azules tienen un guardia cada uno, mientras que el vértice verde tiene dos guardias. En el primer turno, el defensor mueve dos de los tres guardias que puede mover, consiguiendo repeler el ataque al vértice rojo. En el segundo turno, mueve tres, dejando dos guardias en la misma posición. Finalmente, en el tercer turno el atacante gana el juego. La estrategia de defensa usada no es óptima; ¿será posible defender el grafo eternamente con 4 guardias? ¿Y si fuera un  $(10, 10)$ -juego?

forma mantiene a los dos guardias en vértices a distancia 3 (Figura 1.2). En consecuencia, los tres conjuntos  $\{v_i, v_{i+3}\}$  de  $C_6$  son conjuntos dominantes  $(2, 1)$ -eternos.

El problema de dominación eterna surge como una variante del problema de dominación romana, cuya base histórica se remonta al S. IV d.C. [1, 12]. En aquel entonces, el emperador Constantino tenía como objetivo la defensa del vasto imperio romano. Desafortunadamente, no estando Roma en su apogeo, las fuerzas armadas para esta tarea escaseaban. Por ello, la decisión fue organizar a las tropas en cuatro *ejércitos de campo* (FAs por sus iniciales en inglés), cada uno de los cuales se consideraba suficiente no sólo para defender una región, sino también para repeler un ataque que ya había comenzado. En otras palabras, se consideraba que una región estaba *asegurada* si al menos una FA se ubicaba en la región. El problema es que Constantino debía defender ocho regiones en total, que se muestran en la Figura 1.3. Obviamente no era posible mantener asegurada todas las regiones simultáneamente, pero sí era posible trasladar una FA de una región a otra. El problema de trasladar una FA de una región  $v$  a otra  $w$  es que al hacerlo se desprotege a  $v$ . Por este motivo, Constantino decidió que una FA se puede mover de una región  $v$  a otra  $w$  sólo si en  $v$  hay por lo menos dos FAs. El problema de la defensa romana consiste en determinar cómo distribuir las cuatro FAs de forma tal que todo ataque pueda ser repelido en la menor cantidad de movimientos de acuerdo a la restricción de la existencia de al menos dos FAs en una región previo al movimiento. La solución tomada por Constantino fue ubicar dos FAs en Roma y dos en Constantinopla. Si bien esta solución no es óptima para el problema, es posible que la decisión se debiera a la defensa de ataques sucesivos.

En 2004, Cockayne et al. [4] definieron el problema de *dominación romana* como aquel que consiste en determinar la mínima cantidad de guardias (FAs) necesarios para defender un grafo  $G$ , de forma tal que cada vértice sin guardias sea adyacente a un vértice con dos

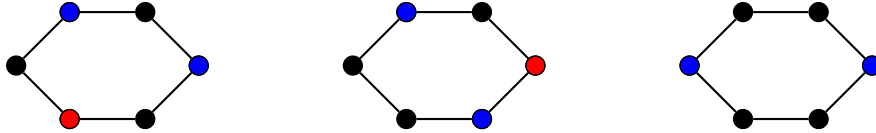


Figura 1.2: Variación de  $\gamma_{x,y}^\infty(C_6)$  para distintos valores de  $x$ . A la izquierda se muestra un grafo  $C_6$  mal defendido por dos guardias azules; el atacante gana jugando en el vértice rojo. En el centro se muestra una defensa correcta para un turno. Hay dos posibilidades cuando el atacante elija el vértice azul: si se puede mover un único guardia, entonces la única defensa posible nos deja en la configuración de la izquierda; caso contrario, podemos pasar a la configuración de la derecha.

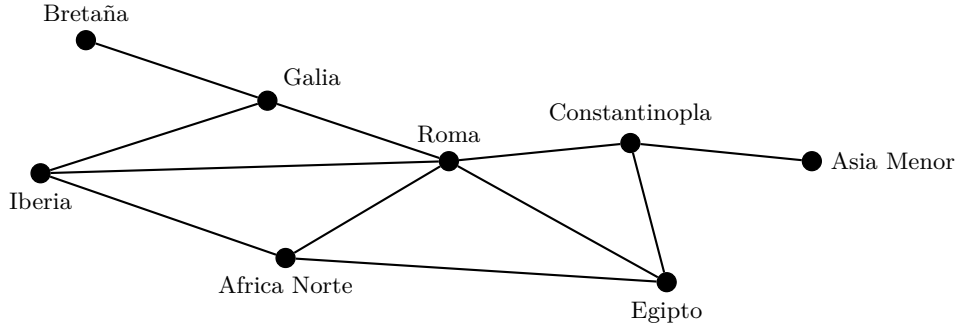


Figura 1.3: Imperio romano en el siglo IV d.C.

guardias. La defensa romana es una variante del problema clásico de dominación en grafos, donde el objetivo es que cada vértice que no tiene un guardia sea adyacente a otro que sí lo posea. En el mismo año 2004, Burger et al. [3] definen el problema de defensa romana en un escenario dinámico donde el atacante ataca sucesivas veces. El objetivo es determinar la mínima cantidad de guardias necesarios para la defensa eterna. Adicionalmente, interesa encontrar un conjunto inicial con la mínima cantidad de guardias que permita generar una estrategia de defensa. Burger et al. plantean este problema no sólo para el problema de dominación romana, sino también para otros problemas de dominación, entre el que se encuentra la versión clásica. En esta última variante el objetivo es determinar el valor  $\gamma_{1,1}^\infty(G)$  definido previamente. Posteriormente, Goddard et al. [6] proponen estudiar la versión  $(n, 1)$  del juego. Es fácil ver que  $\gamma_{1,n}^\infty(G) = \gamma_{1,1}^\infty(G)$  para cualquier grafo  $G$ . En otras palabras, si en cada turno el defensor puede mover un único guardia, entonces hay una estrategia de defensa óptima en la que cada vértice contiene a lo sumo un guardia en cada momento. Goddard et al. conjeturaron también que  $\gamma_{n,1}^\infty(G) = \gamma_{n,n}^\infty(G)$  para todo grafo  $G$ . Recientemente, Finbow et al. [5] mostraron que esta conjetura es falsa; existen grafos en donde mantener más de un guardia en el mismo vértice permite reducir la cantidad de guardias (ver Figura 1.4). En [9] se resumen varios de los resultados sobre distintas variantes de dominación eterna.

El problema de determinar el valor  $\gamma_{x,y}^\infty(G)$  de un grafo  $G$  es un problema difícil, para el cual sólo se conocen algoritmos de tiempo exponencial [5]. Braga et al. [2] estudiaron el problema de dominación eterna para la clase de los grafos de intervalos propios. En su trabajo demuestran que  $\gamma_{n,n}^\infty(G) = \gamma_{1,1}^\infty(G) = \alpha(G)$  para todo grafo de intervalos propios  $G$ , donde  $\alpha(G)$  es el tamaño del conjunto independiente máximo de  $G$ . Más aún, cualquier conjunto independiente máximo es un conjunto dominante  $(x, y)$ -eterno para todo  $1 \leq x, y \leq n$ . Es sabido que tanto  $\alpha(G)$  como un conjunto independiente máximo pueden computarse en tiempo lineal cuando  $G$  es de intervalos propios (ver e.g. [7]). La idea central del trabajo de Braga et al. consiste en separar a  $G$  en tres partes  $A$ ,  $S$  y  $B$  donde  $S$  es una clique que separa a  $A$  de  $B$ . En los grafos de intervalos propios no hay grandes “saltos”, lo que significa que los vértices de  $A$  (o  $B$ ) adyacentes a vértices de  $S$  también forman una clique. Esto permite resolver el problema de la defensa en forma recursiva, separando la defensa de  $A$  de la de  $S \cup B$ . Esta propiedad no es válida para grafos de intervalos generales y es la razón por la que el resultado de Braga et al. no es fácilmente generalizable a ésta clase de grafos.

En esta tesis estudiamos el problema de dominación eterna para los grafos de intervalos. En particular, demostramos que  $\gamma_{n,1}^\infty(G) = \gamma_{n,n}^\infty(G) = \theta_c(G)$  para todo grafo de intervalos



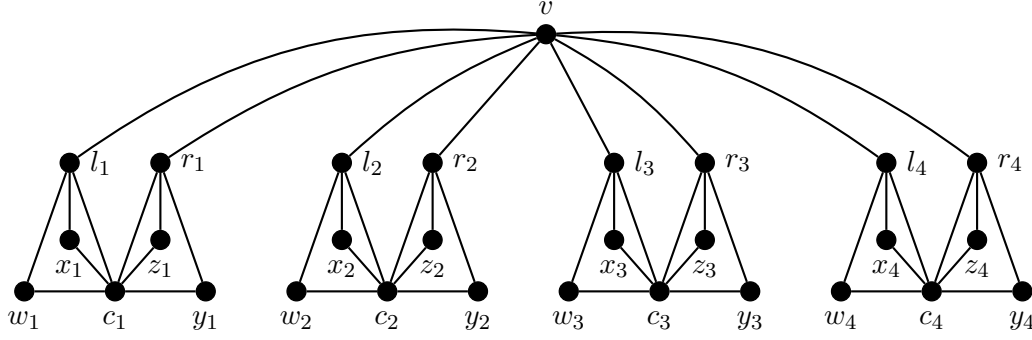


Figura 1.4: Grafo  $G$  con  $\gamma_{n,n}^\infty(G) < \gamma_{n,1}^\infty(G)$ . Sea  $V_i$  el conjunto de los vértices con subíndice  $i$ ,  $1 \leq i \leq 4$ . Si en algún turno del  $(n, 1)$ -juego hay un único guardia en  $G[V_i]$ , entonces el atacante elige  $w_i$ . Como en  $v$  hay a lo sumo un guardia, luego de repeler el ataque quedarán a lo sumo dos guardias en  $G[V_i]$ : uno en  $w_i$  y el otro eventualmente en uno de  $l_i$  o  $r_i$ . Pero entonces el atacante gana el juego eligiendo a  $x_i$  (si no hay guardia en  $l_i$ ) o a  $z_i$  (si hay guardia en  $l_i$  y, por lo tanto, no hay guardia en  $r_i$ ). En consecuencia, luego de cada turno tienen que quedar al menos dos guardias en  $G[V_i]$ . Como esto también en cierto después que el atacante ataca  $v$ , obtenemos que hacen falta al menos 9 guardias para defender a  $G$  en un  $(n, 1)$ -juego. Por otra parte, en un  $(n, 2)$ -juego, podemos disponer dos guardias en  $v$ , un guardia en  $l_i$ , otro en  $r_i$  y otro en un vértice  $m$  de  $V_i \setminus \{l_i, r_i\}$  para algún  $1 \leq i \leq 4$  y, finalmente, un guardia en  $c_j$  para  $1 \leq j \leq 4$ ,  $j \neq i$ . En caso que el atacante elija un vértice  $a \in V_i$ , si  $a = x_i$  y  $m = w_i$  o  $a = w_i$  y  $m = x_i$  lo defendemos moviendo el guardia de  $l_i$  a  $a$  y el guardia de  $m$  a  $l_i$ , si  $a = z_i$  y  $m = y_i$  o  $a = y_i$  y  $m = z_i$  lo defendemos moviendo el guardia de  $r_i$  a  $a$  y el guardia de  $m$  a  $r_i$ , sino lo defendemos moviendo uno de los guardias de  $b \in \{l_i, r_i\}$  hacia  $a$ , uno de los guardias de  $v$  hacia  $b$ , el guardia de  $d \in \{l_i, r_i\} \setminus \{b\}$  hacia  $v$  y el otro guardia de  $V_i$  hacia  $d$ , obteniendo una configuración análoga a la inicial. En caso que el atacante elija a  $a \in V_j$ ,  $j \neq i$ , se mueve el guardia de  $c_j$  hacia  $a$ , los dos guardias de  $v$  hacia  $l_j$  y  $r_j$ , los dos guardias de  $l_i$  y  $r_i$  hacia  $v$  y el guardia sobrante de  $V_i$  hacia  $c_i$ . Esta configuración es nuevamente análoga a la original, reemplazando  $i$  por  $j$ . En consecuencia, podemos defender  $G$  en un  $(n, 2)$ -juego con 8 guardias. Resumiendo,  $\gamma_{n,n}^\infty(G) \leq \gamma_{n,2}^\infty(G) \leq 8 < 9 \leq \gamma_{n,1}^\infty(G)$ . Finbow et al. [5] demuestran que  $\gamma_{n,n}^\infty(G) = 8$  y  $\gamma_{n,1}^\infty(G) = 9$ .

$G$ , donde  $\theta_c(G)$  es el peso máximo de entre todas las neocolonizaciones de  $G$ . Más aún, demostramos que cualquier neocolonización de peso máximo es un conjunto dominante  $(n, y)$ -eterno para cualquier  $1 \leq y \leq n$ . Como discutimos más adelante, nuestro resultado es, en cierta forma, una generalización del resultado de Braga et al. Como corolario, obtenemos un algoritmo lineal para calcular tanto  $\theta_c(G)$  como un conjunto dominante  $(n, y)$ -eterno. Desde el punto de vista técnico, nuestra demostración es considerablemente más simple que la desarrollada por Braga et al. y, en el caso particular de grafos de intervalos propios, la misma se reduce a unas pocas líneas. El resultado principal de esta tesis se encuentra publicado en [11].

La tesis está estructurada de la siguiente forma. En la Sección 2.1 damos una nueva demostración, simple y concisa, de que  $\gamma_{n,n}^\infty(G) = \gamma_{n,1}^\infty(G) = \alpha(G)$  para todo grafo de intervalos propios  $G$ . En Sección 2.2 generalizamos nuestra demostración para todos los grafos de intervalos. El algoritmo resultante se muestra en la Sección 2.3. Cerraremos la tesis con el Capítulo 3 que incluye las conclusiones generales y algunos problemas relacionados que quedan abiertos. Finalmente, en el resto de este capítulo incluimos las definiciones

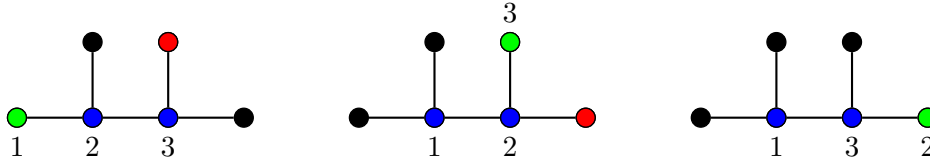


Figura 1.5: Dos turnos mostrando el movimiento virtual de un guardia móvil a través de un conjunto dominante y conexo. Los vértices azules y verde contienen a los guardias “fijos” y “móvil”, respectivamente, mientras que el vértice rojo es atacado. Los números identifican unívocamente a cada guardia. Notar que se mueven todos los guardias en el camino mínimo desde el guardia móvil al vértice atacado, consiguiendo el efecto final de mover al guardia móvil virtualmente; el rol de guardia móvil cambia en cada turno.

necesarias para comprender el resto del trabajo mientras que mostramos distintos tipos de cotas *elementales* sobre las distintas variantes de  $\gamma_{x,y}^\infty$ .

### 1.1. Preliminares y cotas elementales

En esta sección presentamos las definiciones relevantes en el marco de la tesis, a la vez que relacionamos distintos parámetros de los grafos a fin de establecer cotas elementales para los distintos problemas de dominación considerados. Vamos a suponer que el lector tiene algunas nociones básicas de teoría de grafos y de su terminología.<sup>2</sup> Antes de empezar, remarcamos que  $\gamma_{n,n}^\infty(G) \leq \gamma_{n,1}^\infty(G) \leq \gamma_{1,1}^\infty(G)$ , ya que cualquier estrategia de defensa óptima en un  $(x, y)$ -juego es también una estrategia de defensa válida en un  $(x + a, y + b)$ -juego para todo  $a, b \geq 0$ .

En esta tesis escribimos  $V(G)$  para denotar al conjunto de vértices de  $G$  y  $G[V]$  para denotar al subgrafo de  $G$  inducido por  $V \subseteq V(G)$ . Un conjunto  $D \subseteq V(G)$  es *conexo* cuando  $G[D]$  es conexo y *dominante* si cada vértice en  $V(G) \setminus D$  tiene un vecino en  $D$ . El *número de dominación*  $\gamma(G)$  es el mínimo  $k$  tal que  $G$  tiene un conjunto dominante con  $k$  vértices. Análogamente, el *número de dominación conexo*  $\gamma_c(G)$  es el mínimo  $k$  tal que  $G$  tiene un conjunto con  $k$  vértices que es a la vez dominante y conexo. Claramente, el atacante gana un  $(n, n)$ -juego cuando el conjunto inicial de vértices ocupados no es dominante: alcanza con atacar un vértice cuyo vecindario no tenga vértices ocupados (ver Figura 1.2 izquierda). A su vez, un grafo conexo no trivial  $G$  puede ser defendido en un  $(n, 1)$ -juego ubicando un guardia “fijo” en cada vértice de un conjunto dominante y conexo  $D$  y un guardia “móvil” en un vértice  $w \notin D$ . En cada turno, el defensor traslada virtualmente el guardia móvil al vértice atacado  $v$ , moviendo todos los guardias en el camino más corto desde  $w$  hasta  $v$  cuyo interior pertenece a  $D$  (Figura 1.5). Por lo tanto,  $\gamma(G) \leq \gamma_{n,n}^\infty(G) \leq \gamma_{n,1}^\infty(G) \leq 1 + \gamma_c(G)$ .

Cotas similares se pueden obtener para  $(1, 1)$ -juegos considerando los números de independencia y de cubrimiento por cliques. Una *clique* de  $G$  es un conjunto de vértices mutuamente adyacentes de  $G$ , mientras que un *conjunto independiente* es un conjunto de vértices mutuamente no adyacentes de  $G$ . Un *cubrimiento por cliques* de  $G$  es una partición de  $V(G)$  en cliques. El *número de independencia*  $\alpha(G)$  de  $G$  es el máximo tal que  $G$  tiene un conjunto independiente de tamaño  $\alpha(G)$ , mientras que el *número de cubrimiento por cliques*  $\theta(G)$  de  $G$  es el mínimo tal que  $G$  tiene un cubrimiento por cliques con  $\theta(G)$

<sup>2</sup> La misma se introduce en Algoritmos y Estructuras de Datos III, materia obligatoria de la carrera.

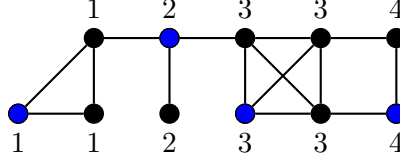


Figura 1.6: Los vértices azules forman un conjunto independiente (máximo). Si el atacante elige un vértice azul distinto en cada paso, entonces se requiere un guardia distinto para repeler cada ataque cuando el defensor puede mover un único guardia por turno. Por otra parte, un cubrimiento por cliques (mínimo)  $V_1, V_2, V_3, V_4$  se obtiene particionando los vértices de acuerdo a la etiqueta. El grafo se puede defender poniendo un guardia para defender a los vértices de  $V_i$  para cada  $1 \leq i \leq 4$ .

partes. Si un vértice diferente de un conjunto independiente es atacado en cada turno, entonces guardias diferentes son requeridos para repeler los ataques (Figura 1.6). A su vez, cada ataque puede ser repelido si un guardia defiende cada clique de un cubrimiento por cliques (Figura 1.6). En otras palabras,  $\alpha(G) \leq \gamma_{1,1}^\infty(G) \leq \theta(G)$ .

Como señaló Goddard et al. [6], ambas cotas superiores pueden ser mejoradas al considerar  $(n, 1)$ -juegos. Definamos el *peso* de un conjunto conexo  $V$  como  $\omega(V) = 1$  si  $V$  es una clique y como  $\omega(V) = 1 + \gamma_c(G[V])$  en caso contrario. Una *neocolonización* de  $G$  es una partición  $\mathcal{V}$  de  $V(G)$  en conjuntos conexos; su *peso* es  $\omega(\mathcal{V}) = \sum_{V \in \mathcal{V}} \omega(V)$  (Figura 1.7). El *número de cubrimiento clique-conexo*  $\theta_c(G)$  de  $G$  es el mínimo tal que  $G$  admite una neocolonización de peso  $\theta_c(G)$ . Por la discusión previa,  $\gamma_{n,1}^\infty(G) \leq \theta_c(G) \leq \min\{\theta(G), \gamma_c(G) + 1\}$  (Figura 1.7).

Junto con  $\gamma_{n,1}^\infty(G) \leq \alpha(G)$  [6, Teorema 13], las de arriba son algunas de las cotas *elementales* que fueron descubiertas desde que los problemas de dominación eterna fueron introducidos en [3] y [6]; ver [9] para una revisión actualizada. Una característica interesante de estas desigualdades es que son fáciles de demostrar: las cotas inferiores surgen a partir de secuencias de ataques golosas simples, mientras que las cotas superiores se obtienen mediante la partición de  $G$  en subgrafos fáciles de defender. Ninguna de las desigualdades en las cadenas  $\gamma(G) \leq \gamma_{n,n}^\infty(G) \leq \gamma_{n,1}^\infty(G) \leq \alpha(G) \leq \gamma_{1,1}^\infty(G) \leq \theta(G)$  y  $\gamma_{n,1}^\infty(G) \leq \theta_c(G) \leq \theta(G)$  vale por igualdad para todos los grafos [9]. En efecto:

- $\gamma(P_3) = 1$  (Figura 1.8 (a)) mientras que  $\gamma_{n,n}^\infty(P_3) = 2$  (Figura 1.8 (b)) ya que si un guardia se encuentra en una hoja, entonces no puede defender la otra hoja en el

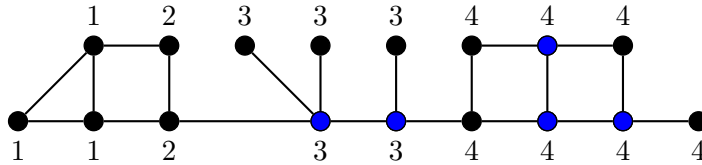


Figura 1.7: El grafo  $G$  mostrado admite la neocolonización  $V_1, V_2, V_3, V_4$ , donde  $V_i$  está formado por los vértices etiquetados con  $i$ . Como  $V_1$  y  $V_2$  son cliques (de peso 1), alcanza con poner un guardia para repeler los ataques a cada clique  $V_1$  y  $V_2$ . Por otra parte, los vértices azules forman conjuntos dominantes y conexos mínimos de  $V_3$  y  $V_4$ . Luego,  $V_3$  y  $V_4$  tienen peso 3 y 4, respectivamente, y podemos defender cada conjunto con un guardia en cada vértice azul y un guardia móvil adicional que se mueve virtualmente como se discute en la Figura 1.5.

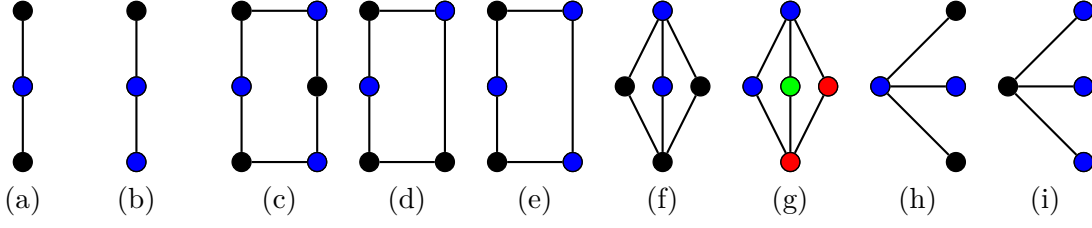


Figura 1.8: Grafos que muestran que las desigualdades de las cadenas  $\gamma \leq \gamma_{n,n}^\infty \leq \gamma_{n,1}^\infty \leq \alpha \leq \gamma_{1,1}^\infty \leq \theta$  y  $\gamma_{n,1}^\infty \leq \theta_c \leq \theta$  son estrictas en algunos casos. Los vértices azules muestran el conjunto correspondiente que se describe en el texto principal.

siguiente ataque.

- $\gamma_{n,n}^\infty(G) = 8 < 9 = \gamma_{n,1}^\infty(G)$  para el grafo  $G$  de la Figura 1.4.
- $\gamma_{n,1}^\infty(C_6) = 2$  como se discute en la Figura 1.2 mientras que  $\alpha(G) = 3$  (Figura 1.8 (c)).
- $\alpha(C_5) = 2$  (Figura 1.8 (d)) mientras que  $\gamma_{1,1}^\infty(C_5) = 3$  ya que si el defensor usa dos guardias, sus vértices correspondientes  $v$  y  $w$  tienen que formar un conjunto dominante como se muestra en la Figura 1.8 (d). Sin embargo, eligiendo el vecino común de  $v$  y  $w$ , los guardias quedan en vértices que no forman un conjunto dominante. En cambio, el grafo se puede defender con tres guardias ubicados en forma arbitraria (Figura 1.8 (e)).
- Si  $G$  es un grafo con  $\alpha(G) = 2$ , entonces  $\gamma_{1,1}^\infty(G) = 3$  [6], ya que alcanza con mantener dos guardias en vértices  $x_1$  y  $x_2$  no adyacentes y otro guardia  $y$  en un vértice arbitrario. Si el vértice  $v$  elegido por el atacante es vecino de  $y$ , entonces se mueve el guardia de  $y$  a  $v$ ; caso contrario, al menos uno de  $x_1$  o  $x_2$  es adyacente a  $v$  y se mueve el guardia del vértice correspondiente. En cualquier caso, dos de los guardias siempre están en vértices no adyacentes y el grafo se puede defender con la estrategia anterior. Por otra parte, existen grafos con  $\alpha(G) = 2$  y  $\theta(G) > 3$  (e.g., los complementos de los grafos de Mycielski<sup>3</sup>). Para dichos grafos,  $\gamma_{1,1}^\infty(G) \leq 3 < \theta(G)$ .
- $\gamma_{n,1}^\infty(K_{2,3}) = 2$  ya que alcanza con mantener dos guardias, cada uno en una parte distinta de la bipartición (Figura 1.8 (f)). En cambio,  $\theta_c(K_{2,3}) = 3$  ya que cualquier cubrimiento por cliques tiene al menos tres cliques (Figura 1.8 (g)), cualquier conjunto dominante y conexo tiene al menos dos vértices y peso 3 (Figura 1.8 (f)), y cualquier partición que contenga al menos una clique (de peso 1) y un conjunto dominante y conexo (de peso al menos 2) tiene peso al menos 3.
- $\theta_c(K_{1,3}) = 2$  porque el vértice no hoja forma un conjunto dominante y conexo de peso 2 (Figura 1.8 (h)), mientras que  $\theta(K_{1,3}) = 3$  porque cada hoja está en una clique maximal distinta (Figura 1.8 (i)).

Aunque ninguna cota vale por igualdad para todos los grafos, la igualdad vale para ciertas clases de grafos. Por ejemplo,  $\alpha(G) = \gamma_{1,1}^\infty(G) = \theta(G)$  cuando  $G$  es perfecto<sup>4</sup> (cf. arriba) y

<sup>3</sup> Partiendo desde el grafo trivial  $M_1$ , el grafo  $M_{i+1}$  se obtiene desde  $M_i$  agregando un vértice  $w(v)$  con vecindario  $N(v) \cup \{u\}$  por cada  $v \in V(M_i)$ , donde  $u$  es un vértice nuevo. El grafo de Mycielski  $M_i$  no tiene cliques de tamaño 3 y cualquier partición de  $V(M_i)$  en conjuntos independientes tiene al menos  $i$  conjuntos [10]. En otras palabras,  $\alpha(\overline{M_i}) = 2$  y  $\theta(\overline{M_i}) = i$ .

<sup>4</sup> Un grafo  $G$  es perfecto cuando  $\alpha(G[V]) = \theta(G[V])$  para todo  $V \subseteq V(G)$ .

$\gamma_{n,1}^\infty(G) = \theta_c(G)$  cuando  $G$  es un árbol [8]. Como se indica en la sección anterior, Braga et al. [2] demostraron que  $\gamma_{n,n}^\infty(G) = \alpha(G)$  cuando  $G$  es un grafo de intervalos propios. Como los grafos de intervalos propios son perfectos [7], también vale que  $\gamma_{n,n}^\infty(G) = \theta_c(G)$ . En esta tesis generalizamos este resultado probando que  $\gamma_{n,n}^\infty(G) = \theta_c(G)$  para todos los grafos de intervalos. Mientras que Braga et al. derivan cotas inferiores no triviales de  $\gamma_{n,n}^\infty(G)$  para grafos generales, nosotros obtenemos una demostración simple y sucinta, similar en espíritu a las de las cotas elementales, restringiendo nuestra atención exclusivamente a los grafos de intervalos.

## 2. DOMINACIÓN ETERNA EN GRAFOS DE INTERVALOS

Un grafo  $G$  es *de intervalos* cuando es posible asignar un intervalo cerrado  $I(v)$  a cada vértice  $v \in V(G)$  de forma tal que  $v$  y  $w$  son adyacentes en  $G$  si y sólo si  $I(v) \cap I(w) \neq \emptyset$ . La familia de intervalos  $\mathcal{I} = \{I(v) \mid v \in V(G)\}$  es un *modelo de intervalos* de  $G$ . La Figura 2.4 muestra un grafo de intervalos  $G$  junto con uno de sus modelos de intervalos. Si ningún intervalo de  $\mathcal{I}$  está propiamente contenido en otro intervalo, entonces  $\mathcal{I}$  es un *modelo de intervalos propios*. Aquellos grafos que admiten un modelo de intervalos propios son llamados *grafos de intervalos propios*. Es sabido que un grafo  $G$  es de intervalos propios si y sólo si es de intervalos y no contiene a  $K_{1,3}$  como subgrafo inducido [7]. En consecuencia, el grafo de la Figura 2.4 no es de intervalos propios. En la Figura 2.1 (izquierda) se muestra un grafo de intervalos propios: uno de sus modelos se muestra en la Figura 2.2.

Dado un modelo de intervalos  $\mathcal{I}$  de un grafo  $G$ , vamos a escribir  $v(I)$  para denotar al vértice de  $G$  correspondiente a  $I \in \mathcal{I}$ . Análogamente, escribimos  $v(\mathcal{I}') = \{v(I) \mid I \in \mathcal{I}'\}$  para denotar al conjunto de vértices correspondientes al submodelo  $\mathcal{I}' \subseteq \mathcal{I}$ . Finalmente, escribimos  $s(I) = s$  y  $t(I) = t$  para denotar al *extremo inicial* y al *extremo final* del intervalo  $I = [s, t]$  cuyos *extremos* son  $s$  y  $t$ , respectivamente.

### 2.1. Dominación eterna en grafos de intervalos propios

Braga et al. [2] demostraron que  $\gamma_{n,n}^\infty(G) = \gamma_{1,1}^\infty(G) = \alpha(G)$  para todo grafo de intervalos propios  $G$ . La demostración de Braga et al. está basada en la siguiente construcción. Para  $A, S \subseteq V(G)$ , definamos  $N(A, S)$  como el subconjunto de los vértices de  $A$  que tienen algún vecino en  $S$ . El grafo  $G^+[A, S]$  se obtiene de  $G[A]$  agregando dos vértices  $a(v)$  y  $b(v)$  por cada  $v \in N(A, S)$  de forma tal que  $v, a(v), b(v)$  formen un camino (Figura 2.1). Desde un punto de vista técnico, el siguiente es el resultado central en [2].

**Teorema 1** ([2, Teorema 2]). *Sea  $G$  un grafo y  $A, S, B$  una partición de  $V(G)$  tal que  $S$  es una clique y ningún vértice de  $A$  es adyacente a un vértice de  $B$ . Si  $\gamma(G^+[A, S]) \geq |N(A, S)| + p$  para algún  $p \geq 0$ , entonces  $\gamma_{n,n}^\infty(G) \geq \gamma_{n,n}^\infty(G[S \cup B]) + p$ .*

Para aplicar el Teorema 1, consideremos un modelo de intervalos propios  $\mathcal{I}$  de un grafo  $G$  cuyos intervalos  $I_1, \dots, I_n$  son tales que  $s(I_i) < s(I_{i+1})$  para  $1 \leq i < n$ . Definamos  $p$  y  $q$  como los máximos índices tales que  $I_1 \cap I_p \neq \emptyset$  e  $I_{p+1} \cap I_q \neq \emptyset$  (Figura 2.2). Por construcción:

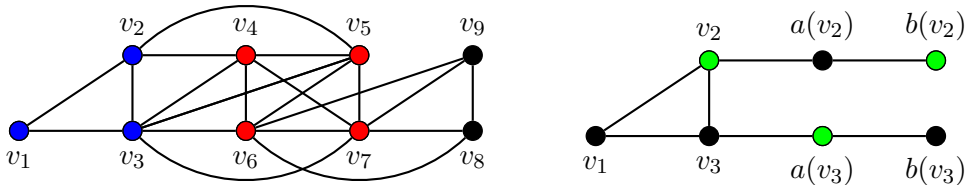


Figura 2.1: A la derecha se observa el grafo  $G^+[A, S]$  correspondiente al grafo  $G$  de la izquierda, donde  $A$  está formado por los vértices azules y  $S$  por los vértices rojos. Los vértices verdes de  $G^+[A, S]$  forman un conjunto dominante mínimo con  $|N(A, S)| + 1$  vértices.

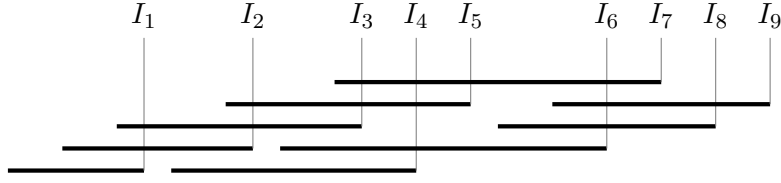


Figura 2.2: Modelo de intervalos propios correspondiente al grafo de la Figura 2.1 izquierda donde  $I_i = I(v_i)$  para  $1 \leq i \leq 9$ ; en este caso,  $p = 3$  y  $q = 7$ .

1.  $A = \{v(I_1), \dots, v(I_p)\}$  y  $S = \{v(I_{p+1}), \dots, v(I_q)\}$  son cliques de  $G$ ,
2. todos los vecinos de  $v(I_1)$  pertenecen a  $A$ , y
3. ningún vértice de  $A$  es adyacente a un vértice de  $B$ .

Al considerar un conjunto dominante  $D$  del grafo  $G^+[A, S]$ , observamos que  $D$  tiene que contener al menos uno de los vértices  $a(v(I_i))$  o  $b(v(I_i))$  para cada  $v(I_i) \in A$  tal que  $v(I_i) \in N(A, S)$  tiene un vecino en  $S$  (i.e., que  $t(I_i) > s(I_p)$ ). Más aún, como  $v(I_1)$  no es adyacente a ninguno de estos vértices, obtenemos que  $|D| > |N(A, S)|$  (Figura 2.1). En otras palabras,  $\gamma(G^+[A, S]) \geq |N(A, S)| + 1$ . Luego  $\gamma_{n,n}^\infty(G) \geq \gamma_{n,n}^\infty(G[S \cup B]) + 1$  por Teorema 1. Por otra parte, cualquier conjunto independiente de  $G$  contiene a lo sumo un vértice de  $A$  porque  $A$  es una clique. Por lo tanto,  $\alpha(G[S \cup B]) + 1 \geq \alpha(G)$ . Luego, si suponemos que  $\gamma_{n,n}^\infty(G[S \cup B]) = \alpha(G[S \cup B])$  por hipótesis inductiva, obtenemos que:

$$\gamma_{n,n}^\infty(G) \geq \gamma_{n,n}^\infty(G[S \cup B]) + 1 = \alpha(G[S \cup B]) + 1 \geq \alpha(G).$$

Como mencionamos en el Capítulo 1,  $\alpha(G) \geq \gamma_{n,n}^\infty(G)$  para todo grafo  $G$  (cota elemental) mientras que  $\gamma_{1,1}^\infty(G) = \alpha(G)$ . Por lo tanto,  $\gamma_{n,n}^\infty(G) = \gamma_{n,1}^\infty(G) = \gamma_{1,1}^\infty(G) = \alpha(G)$ .

El párrafo anterior resume la idea central en la demostración de Braga et al. A nuestro entender, esta demostración sufre de dos inconvenientes principales. Por un lado, el Teorema 1 es demasiado general, ya que funciona para cualquier grafo, sea o no de intervalos propios. Si bien desde un punto de vista teórico conviene tener un resultado general, su demostración resulta innecesariamente complicada para el propósito de demostrar cómo defender un grafo de intervalos propios. Por otro lado, el Teorema 1 no se puede generalizar a grafos de intervalos con la misma idea. Si bien siempre podemos construir una clique  $A$  que contenga a todos los vecinos del intervalo con mínimo extremo final, no es cierto que  $A$  pueda separarse del resto del grafo por una clique (e.g. Figura 2.3). A continuación subsanamos el primer inconveniente demostrando de forma directa que  $\gamma_{n,n}^\infty(G) = \alpha(G)$  para todo grafo de intervalos propios  $G$ . En la sección siguiente mostramos que esta demostración sí se puede generalizar de forma simple a todos los grafos de intervalos.

**Teorema 2.** Si  $G$  es un grafo de intervalos propios, entonces  $\gamma_{n,n}^\infty(G) = \gamma_{n,1}^\infty(G) = \alpha(G)$ .

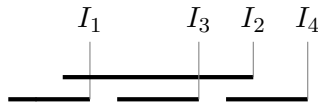


Figura 2.3: Cuando  $v(I_1)$  pertenece a un  $K_{1,3}$ , no se puede separar  $N(v_1)$  del resto del grafo por medio de una clique, ya que alguno entre  $v(I_3)$  o  $v(I_4)$  no pertenece a esa clique y es vecino de  $v(I_2)$ .



*Demostración.* Sea  $\mathcal{I}$  un modelo de intervalos propios de  $G$  y consideremos cualquier conjunto de intervalos  $\mathcal{A} = \{A_1, \dots, A_k\}$  tales que  $v(A_1), \dots, v(A_k)$  es un conjunto independiente de  $G$ . Sin pérdida de generalidad, supongamos que  $s(A_i) < s(A_{i+1})$  para todo  $1 \leq i < k$ .

Veamos, por inducción en  $j$ , que  $v(A_{i+j})$  está a distancia al menos  $j+1$  de  $v(A_i)$  para cada  $1 \leq i < i+j \leq k$ . El caso base  $j = 1$  vale porque  $v(A_i)$  y  $v(A_{i+1})$  no son adyacentes y, por lo tanto, su distancia es al menos 2. Para el paso inductivo  $j > 1$ , consideremos cualquier camino  $v_1, \dots, v_d$  desde  $v_1 = v(A_i)$  hasta  $v_d = v(A_{i+j})$  en  $G$ . Si llamamos  $I_2$  al intervalo correspondiente a  $v_2$ , tenemos que  $s(I_2) \leq t(A_i)$  por ser  $v_1$  adyacente a  $v_2$  y  $t(A_i) < s(A_{i+1})$  por ser  $v(A_i)$  no adyacente a  $v(A_{i+1})$ . Como  $\mathcal{I}$  es propio, esto implica que  $t(I_2) \leq t(A_{i+1})$ . En consecuencia, tenemos dos posibilidades. O bien  $v(A_{i+1}) = v_h$  para algún  $3 \leq h \leq d$  en cuyo caso  $v_h, \dots, v_d$  es un camino de  $v(A_{i+1})$  a  $v_d$  o bien  $v(A_{i+1}), v_3, \dots, v_d$  es un camino de  $v(A_{i+1})$  a  $v_d$ . En cualquier caso,  $d \geq 1 + d'$  donde  $d'$  es la distancia de  $v(A_{i+1})$  a  $v(A_{i+j})$ . Por hipótesis inductiva,  $d' \geq j$  y, por lo tanto,  $d \geq j+1$ .

Como  $v(A_{i+j})$  está a distancia al menos  $j+1$  de  $v(A_i)$  y en cada turno un guardia se mueve entre vértices adyacentes, obtenemos que ningún guardia puede ocupar  $v(A_i)$  en un turno  $z$  y  $v(A_{i+j})$  en el turno  $z+j$ . Esto significa que si el atacante decide atacar  $v(A_i)$  en el  $i$ -ésimo turno para cada  $1 \leq i \leq k$ , entonces ningún guardia utilizado para repeler el  $i$ -ésimo ataque puede ser utilizado para defender  $v(A_{i+j})$  en el  $(i+j)$ -ésimo turno. Consecuentemente, al menos  $k$  guardias son necesarios para repeler estos primeros  $k$  ataques, es decir,  $\gamma_{n,n}^\infty(G) \geq k$ . Ahora, como  $v(A_1), \dots, v(A_k)$  es un conjunto independiente arbitrario, podemos suponer que  $k = \alpha(G)$ . Finalmente, teniendo en cuenta que  $\alpha(G) \geq \gamma_{n,1}^\infty(G) \geq \gamma_{n,n}^\infty(G)$  por las cotas elementales, concluimos que  $\gamma_{n,n}^\infty(G) = \gamma_{n,1}^\infty(G) = \alpha(G)$ .  $\square$

## 2.2. Dominación eterna en grafos de intervalos generales

Recordemos que  $\gamma_{n,1}^\infty(G) \leq \theta_c(G) \leq \theta(G)$  para todo grafo  $G$ . En el caso en que  $G$  es un grafo de intervalos propios, obtenemos que  $\theta(G) = \alpha(G) = \gamma_{n,1}^\infty(G) = \gamma_{n,n}^\infty(G)$ . Dicho de otra forma, el Teorema 2 establece que  $\gamma_{n,n}^\infty(G) = \gamma_{n,1}^\infty(G) = \theta_c(G)$ . En esta sección generalizamos el Teorema 2 a todos los grafos de intervalos. La demostración es análoga a la que usamos para el caso de los grafos de intervalos propios.

Intuitivamente, podemos dividir la demostración del Teorema 2 en dos partes. Por un lado, para demostrar que  $\gamma_{n,n}^\infty(G) \geq \alpha(G)$  la idea es generar una secuencia de ataques, en este caso dada por el conjunto independiente  $v(A_1), \dots, v(A_k)$ . El objetivo es que los intervalos atacados estén lo más pegados y sean lo más cortos posible, a fin de obligar al defensor a usar la mayor cantidad posible de guardias. Por otro lado, para demostrar que  $\theta(G) \geq \gamma_{n,1}^\infty(G)$ , bastó con mostrar una estrategia de defensa; esto lo hicimos en la Sección 1.1, ya que esta cota es elemental. Visto desde la perspectiva de los modelos de intervalos propios, podemos pensar que la idea es posicionar los guardias en intervalos que “defiendan” (i.e., intersequen) la mayor cantidad de otros intervalos. Este mismo esquema se utiliza en la siguiente demostración, donde los intervalos  $A_i$  se usan para atacar y los  $D_i$  para defender. Es por esto que usamos los intervalos  $A_i$  para demostrar que  $\gamma_{n,n}^\infty(G) \geq \theta_c(G)$  y los intervalos  $D_i$  para demostrar que  $\theta_c(G) \geq \gamma_{n,n}^\infty(G)$ .

**Teorema 3.** Si  $G$  es un grafo de intervalos, entonces  $\gamma_{n,n}^\infty(G) = \gamma_{n,1}^\infty(G) = \theta_c(G)$

*Demostración.* Sin pérdida de generalidad,  $G$  tiene un modelo de intervalos  $\mathcal{I}$  cuyos intervalos tienen todos diferentes extremos (ver Figura 2.4). Definimos  $A_1, \dots, A_k$  y  $D_1, \dots, D_k$



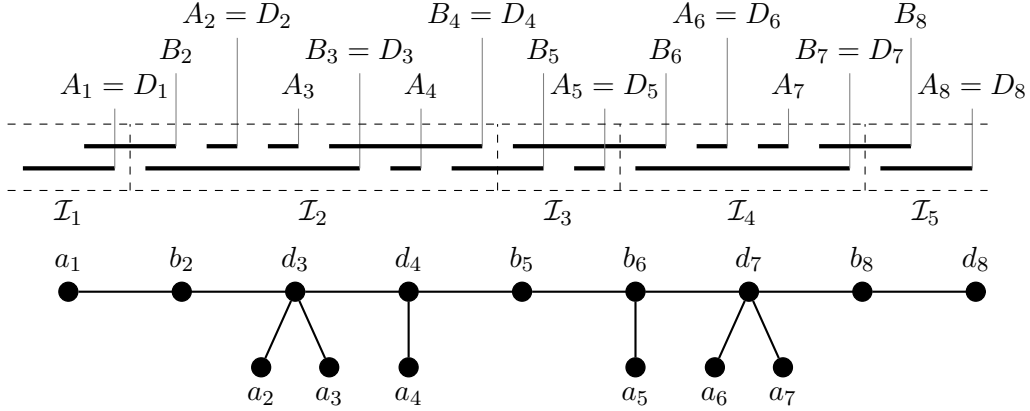


Figura 2.4: Un modelo de intervalos  $\mathcal{I}$  (arriba) de un grafo de intervalos  $G$  (abajo). Las etiquetas para los intervalos se corresponden con las de la demostración del Teorema 3, donde  $a_i = v(A_i)$ ,  $b_i = v(B_i)$ , y  $d_i = v(D_i)$ . Notemos que  $h = 6$ ,  $p_2 = 2$ ,  $p_3 = 5$ ,  $p_4 = 6$ , y  $p_5 = 8$ . Por lo tanto,  $\mathcal{I}_2 = \{I \in \mathcal{I} \mid t(D_1) < s(I) < t(D_4)\} = \{B_3, A_2, A_3, B_4, A_4, B_5\}$ .

como las máximas secuencias tales que  $A_1 = D_1$  es el intervalo de  $\mathcal{I}$  con mínimo extremo final, y:

- (a)  $B_{i+1}$  es el intervalo con extremo final  $\max\{t(I) \mid I \in \mathcal{I} \text{ y } s(I) < t(D_i)\}$ ,
- (b)  $A_{i+1}$  es el intervalo con extremo final  $\min\{t(I) \mid I \in \mathcal{I} \text{ y } s(I) > t(D_i)\}$ , y
- (c)  $D_{i+1} = A_{i+1}$  si  $t(A_{i+1}) > t(B_{i+1})$  y  $D_{i+1} = B_{i+1}$  sino.

Afirmamos que  $v(A_{i+j})$  está a distancia al menos  $j + 1$  de  $v(A_i)$  para cada  $1 \leq i < i + j \leq k$ . Notemos que  $t(D_{i+j-1}) < s(A_{i+j})$  por (b). En consecuencia, basta con probar que si  $v \in V(G)$  está a distancia a lo sumo  $j$  de  $v(A_i)$ , entonces  $s(I(v)) < t(D_{i+j-1})$ . El caso  $j = 1$  sale por (c), porque  $s(I(v)) < t(A_i) \leq t(D_i)$  cuando  $v$  es adyacente a  $v(A_i)$ . Para el caso  $j > 1$ , notemos que  $s(I(v)) < t(I(w))$  para todos los vecinos  $w$  de  $v$ . Si  $w$  está a distancia a lo sumo  $j - 1$  de  $v(A_i)$ , entonces  $s(I(w)) < t(D_{i+j-2})$  por inducción. Así,  $t(I(w)) \leq t(B_{i+j-1})$  por (a), mientras que  $t(B_{i+j-1}) \leq t(D_{i+j-1})$  por (c). Por ende,  $s(I(v)) < t(D_{i+j-1})$  como queríamos demostrar.

Como  $v(A_{i+j})$  está a distancia al menos  $j + 1$  de  $v(A_i)$ , ningún guardia puede ocupar  $v(A_i)$  en un turno  $z$  y  $v(A_{i+j})$  en el turno  $z + j$ . Esto significa que si el atacante decide atacar  $v(A_i)$  en el  $i$ -ésimo turno para cada  $1 \leq i \leq k$ , entonces ningún guardia utilizado para repeler el  $i$ -ésimo ataque puede ser utilizado para defender  $v(A_{i+j})$  en el  $(i + j)$ -ésimo turno. Consecuentemente, son necesarios al menos  $k$  guardias para repeler estos primeros  $k$  ataques, es decir,  $\gamma_{n,n}^\infty(G) \geq k$ .

Ahora vamos a probar que  $k \geq \theta_c(G)$ . Para simplificar la notación, digamos que  $A_0 = D_0$  y  $A_{k+1} = D_{k+1}$  son intervalos que no pertenecen a  $\mathcal{I}$  con  $t(A_0) < \min\{s(I) \mid I \in \mathcal{I}\}$  y  $s(A_{k+1}) > \max\{t(I) \mid I \in \mathcal{I}\}$ . De esta forma,  $p_0 = 0, p_1 = 1$ , y  $p_h = k + 1$  para los índices  $p_0 < \dots < p_h$  tales que  $D_{p_i} = A_{p_i}$ . Fijemos  $1 \leq i \leq h$ , y consideremos  $p = p_i$ ,  $q = p_{i+1}$ , e  $\mathcal{I}_i = \{I \in \mathcal{I} \mid t(D_{p-1}) < s(I) < t(D_{q-1})\}$  (Figura 2.4). Si  $q = p + 1$ , entonces  $A_p = D_p = D_{q-1}$ ; esto implica, por (b), que  $t(A_p) = t(D_{q-1})$  es el mínimo entre los extremos finales en  $\mathcal{I}_i$ . En consecuencia,  $v(\mathcal{I}_i)$  es una clique porque todos sus intervalos contienen a  $t(D_{p-1})$ . Notemos que, por definición,  $v(\mathcal{I}_i)$  tiene peso  $p_{i+1} - p_i = 1$ . Si  $q > p + 1$ ,

entonces  $D_i = B_i$  para todo  $p < i < q$  y, por (a),  $s(D_{i+1}) < t(D_i)$  cuando  $i < q - 1$ . Esto, a su vez, implica que  $D = D_{p+1} \cup \dots \cup D_{q-1}$  es el intervalo  $[s(D_{p+1}), t(D_{q-1})]$ . Más aún, como  $A_p = D_p$ , entonces, por (b),  $t(A_p) = t(D_p) > s(D)$  es el mínimo entre todos los extremos finales en  $\mathcal{I}_i$  y, por lo tanto, cada intervalo en  $\mathcal{I}_i$  tiene al menos un extremo dentro de  $D$ . En otras palabras,  $v(D_{p+1}), \dots, v(D_{q-1})$  es un conjunto dominante conexo de  $G[v(\mathcal{I}_i)]$  que, por definición, tiene peso  $\omega(v(\mathcal{I}_i)) \leq q - p = p_{i+1} - p_i$ . En resumen,  $\mathcal{V} = \{v(\mathcal{I}_1), \dots, v(\mathcal{I}_{h-1})\}$  es una neocolonización de  $G$  con peso

$$\omega(\mathcal{V}) = \sum_{i=1}^{h-1} \omega(v(\mathcal{I}_i)) \leq \sum_{i=1}^{h-1} (p_{i+1} - p_i) = p_h - p_1 = k. \quad (2.1)$$

En consecuencia,  $\theta_c(G) \leq k \leq \gamma_{n,n}^\infty(G) \leq \gamma_{n,1}^\infty(G) \leq \theta_c(G)$  como queríamos demostrar.  $\square$

Repasando la demostración del Teorema 3, podemos observar que  $v(A_1), \dots, v(A_k)$  es un conjunto independiente de  $G$ . En el caso particular en que el modelo  $\mathcal{I}$  es propio, observamos también que  $k = h - 1$  y  $A_i = D_i$  para  $1 \leq i \leq k$ . En consecuencia,  $v(\mathcal{I}_1), \dots, v(\mathcal{I}_k)$  es un cubrimiento por cliques de  $G$  y, por lo tanto,  $\theta(G) \leq \gamma_{n,n}^\infty(G) \leq \alpha(G) \leq \theta(G)$ . Dicho en otras palabras, la demostración del Teorema 3 también implica el resultado de Braga et al. Esto no es una sorpresa, teniendo en cuenta que la demostración del Teorema 2 que dimos en la Sección 2.1 es la simplificación de la demostración del Teorema 3.

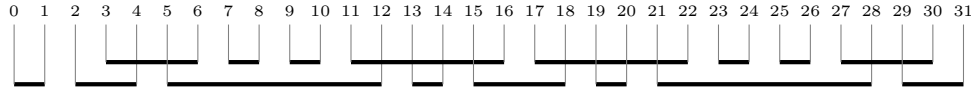
En el caso general,  $\mathcal{V} = \{v(\mathcal{I}_1), \dots, v(\mathcal{I}_{h-1})\}$  es una neocolonización de peso mínimo. Más aún,  $v(\mathcal{I}_i)$  puede ser defendido eternamente si un guardia es ubicado inicialmente en  $v(D_j)$  para  $p \leq j < q$ , porque  $v(D_{p+1}), \dots, v(D_{q-1})$  es un conjunto dominante conexo de  $G[v(\mathcal{I}_i)]$  cuando  $v(\mathcal{I}_i)$  no es una clique. Por ende,  $D = \{v(D_i) \mid 1 \leq i \leq k\}$  es un conjunto dominante  $(n, y)$ -eterno de  $G$  de tamaño mínimo, para cualquier  $1 \leq y \leq n$ .

### 2.3. Algoritmo para calcular un conjunto dominante $(n, y)$ -eterno

El algoritmo para calcular un conjunto dominante  $(n, y)$ -eterno consiste en calcular, para un modelo de intervalos  $\mathcal{I}$ , la secuencia  $D = D_1, \dots, D_k$  definida por (c) en la demostración del Teorema 3. Como parte del algoritmo, también computamos la secuencia  $A = A_1, \dots, A_k$  definido de acuerdo a la regla (b). A continuación describimos el algoritmo a alto nivel; su implementación en C++ se muestra en el Algoritmo 2.1 y se utiliza como referencia.

El modelo de entrada  $\mathcal{I}$  se implementa con una secuencia de intervalos (línea 14). A su vez, cada intervalo  $I = [s, t]$  de  $\mathcal{I}$  se representa con el par de números  $s$  y  $t$  (línea 7). Suponemos que tanto  $s$  como  $t$  son enteros pertenecientes al intervalo  $[2, 2n + 1]$ . Más aún, requerimos que todos los intervalos de  $\mathcal{I}$  tengan extremos distintos, con lo cual hay exactamente un intervalo con un extremo  $i$  para cada  $i \in [2, 2n + 1] \cap \mathbb{Z}$ . Para simplificar la implementación, agregamos el intervalo  $[0, 1]$  a  $\mathcal{I}$  que representa al intervalo  $A_0 = D_0$  de la demostración del Teorema 3 (línea 23). La Figura 2.5 muestra parte del input correspondiente al modelo de la Figura 2.4 y el modelo obtenido luego de agregar el intervalo  $[0, 1]$ . La razón por la que consideramos “corridos” a todos los intervalos es para facilitar la implementación del código, ya que los índices de C++ empiezan desde 0.

Antes de empezar el cálculo del conjunto dominante eterno propiamente dicho, aplicamos un pre-procesamiento sobre  $\mathcal{I}$  que consiste en la creación de un *vector de extremos*  $E = e_0, \dots, e_{2n+1}$  (líneas 26–30). Para cada  $0 \leq i \leq 2n + 1$ , el elemento  $e_i$  es un par que



**Input:**  $\{[2, 4], [3, 6], [5, 12], [7, 8], [9, 10], [11, 16], \dots, [25, 26], [27, 30], [29, 31]\}$

$E = \langle ([0, 1], s), ([0, 1], t), ([2, 4], s), ([3, 6], s), ([2, 4], t), ([5, 12], s), ([3, 6], t), \dots, ([27, 30], t), ([29, 31], t) \rangle$

Figura 2.5: Representación computacional del modelo de intervalos de la Figura 2.4 y su correspondiente vector de extremos  $E$ .

contiene al intervalo  $I$  que corresponde al extremo  $i$  junto con un tipo  $s$  o  $t$  que indica si el extremo de  $I$  correspondiente a  $i$  es un extremo inicial o final de  $I$  (línea 17). La Figura 2.5 muestra parte del vector de extremos  $E$  generado en el pre-procesamiento para el modelo input de ejemplo. Como se observa en el Algoritmo 2.1 (líneas 26–30), podemos calcular  $E$  en tiempo  $O(n)$  recorriendo una vez cada intervalo de  $\mathcal{I}$ .

Una vez computado  $E$ , el algoritmo procede al cómputo de las secuencias  $A$  y  $D$ . Este cómputo se realiza en un ciclo principal que en la  $(i + 1)$ -ésima iteración computa los intervalos  $A_{i+1}$  y  $D_{i+1}$  a partir de los intervalos  $A_i$  y  $D_i$  previamente computados (líneas 36–54). Para simplificar la programación, inicializamos tanto  $A$  como  $D$  con el intervalo *ficticio*  $A_0 = D_0 = [0, 1]$  que, en rigor, no forma parte de la solución (línea 33). Este intervalo es descartado al finalizar el algoritmo (línea 58). Cada iteración del ciclo principal se puede dividir en tres partes. La primera (líneas 38–45) es la encargada de calcular  $A_{i+1}$ , la segunda (líneas 48–50) calcula el intervalo  $B_{i+1}$  definido como en (a) y, finalmente, la tercera (línea 53) calcula  $D_{i+1}$ .

Para el cálculo de  $A_{i+1}$  recordemos que, por (b),  $A_{i+1}$  es el intervalo con  $s(A_{i+1}) > t(D_i)$  cuyo extremo  $t(A_{i+1}) \leq 2n$  es lo mínimo posible. Como el objetivo es minimizar el valor de  $t(A_{i+1})$ , empezamos inicializando  $A_{i+1}$  como  $[2n + 1, 2n + 2]$  (línea 38). Luego, recorremos el vector de extremos  $E$  hacia adelante desde la posición  $t(D_i)$ . Para cada extremo  $e = [I, s]$  en  $E$ , actualizamos  $A_{i+1} := I$  en el caso en que  $t(I) < t(A_{i+1})$ . El recorrido termina en el momento en que llegamos a la posición  $t(A_{i+1})$  de  $e$ , dado que sabemos que todos los extremos subsiguientes corresponden a intervalos con extremo final mayor a  $t(A_{i+1})$ . Remarcamos que podría ocurrir que no existieran extremos de tipo  $s$  en  $E$  después de  $t(D_i)$ ; en este caso el algoritmo finaliza al llegar al último extremo de  $E$  (línea 44). Notar que, en este caso,  $k = i$ , i.e.,  $A = A_1, \dots, A_i$  es la secuencia buscada.

Observemos que en la búsqueda de  $A_{i+1}$  recorremos sólo una parte de  $E$ , que va desde la posición  $t(D_i)$  hasta la posición  $t(A_{i+1})$ . Por (c), sabemos que  $t(D_i) \geq t(A_i)$ . En consecuencia, no recorremos ningún intervalo más de una vez para el cómputo total de  $A$ . Como se puede observar en el Algoritmo 2.1 (líneas 38–45), cada operación del recorrido consiste en accesos a un vector y asignaciones y comparaciones de números acotados. Por lo tanto, cada operación requiere  $O(1)$  tiempo y, en consecuencia, el tiempo total consumido para el cómputo de  $A$  es  $O(n)$ .

El cálculo de  $B_{i+1}$  sigue una estrategia similar al utilizado para computar  $A_{i+1}$ . Recordemos que, por (a),  $B_{i+1}$  es el intervalo con  $s(B_{i+1}) < t(D_i)$  cuyo extremo final es máximo. Sea  $j = 0$  si  $i = 0$  y  $j = t(D_{i-1})$  si  $i > 0$ . Cuando  $i \geq 1$ , podemos observar que  $t(D_{i-1}) < s(A_i) < t(A_i) \leq t(D_i)$  por (b) y (c) y, por lo tanto,  $s(B_{i+1}) \in [j, t(D_i)]$ . Obviamente, si  $i = 0$ , entonces  $s(B_1) \in [j, t(D_0)]$ . En cualquier caso,  $s(B_{i+1})$  pertenece al intervalo  $[j, t(D_i)]$  (línea 49). En consecuencia, alcanza con inicializar  $B_{i+1} = [-2, -1]$  para luego recorrer todos los extremos  $e = [I, s] \in E$  entre las posiciones  $j$  y  $t(D_i)$ , ac-

tualizando  $B_{i+1} := I$  si  $t(I) > t(B_{i+1})$  (líneas 48–50). Como antes, este recorrido cuesta  $O(t(D_i) - t(D_{i-1}))$  cuando  $i > 0$  y  $O(t(D_1))$  cuando  $i = 0$ . Por lo tanto, el costo total para computar todos los  $B_i$  es  $O(n)$ . Finalmente, por (c),  $D_{i+1}$  es el intervalo en  $\{A_{i+1}, B_{i+1}\}$  con máximo extremo final. Ciertamente,  $D_{i+1}$  se puede calcular en  $O(1)$  tiempo con una mera comparación (línea 53). Podemos concluir, pues, que el Algoritmo 2.1 requiere  $O(n)$  tiempo.

### Algoritmo 2.1: cómputo de un conjunto dominante eterno en C++

Para la implementación del algoritmo, suponemos que el modelo de entrada  $\mathcal{I}$  se ingresa por la consola estándar. Se espera un primer entero  $n = |\mathcal{I}|$ , seguido de  $n$  pares de enteros  $s$  y  $t$  con  $2 \leq s < t \leq 2n + 1$  que representan cada intervalo  $[s, t]$  de  $\mathcal{I}$ . Cada valor de la entrada se encuentra separado del siguiente por al menos un espacio. Asimismo, el conjunto dominante eterno  $D$  computado por el algoritmo es mostrado en la consola estándar.

```

1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5
6  //Par representando un intervalo I = [s,t]; beginning = s, ending = t
7  struct interval{int beginning, ending;};
8
9  ostream& operator<<(ostream& out, interval I) {
10     return out << "(" << I.beginning << "," << I.ending << ") ";
11 }
12
13 //Modelo = secuencia de intervalos
14 using model = vector<interval>;
15
16 //Un elemento del vector de extremos; kind == true sii tipo = t
17 struct extreme{interval I; bool kind;};
18
19 int main()
20 {
21     //Lectura y pre-procesamiento del modelo M = I; notar que M[0] = [0,1]
22     int n; cin >> n;
23     model M(n+1, {0,1});
24     for(int i = 1; i <= n; i++) cin >> M[i].beginning >> M[i].ending;
25
26     vector<extreme> E(2*n+2);
27     for(auto I : M){
28         E[I.beginning] = {I, false};
29         E[I.ending] = {I, true};
30     }
31
32     //Inicialización de las secuencias A y D
33     vector<interval> A{M[0]}, D{M[0]};
34
35     //Ciclo principal
36     while(true){
37         //Parte 1: computo de I = A_{i+1} cuando A = A_0, ..., A_i
38         interval I{2*n+2, 2*n+3};
39
40         int j = D.back().ending;
41         for(; j < 2*n and (!E[j].kind or E[j].I.ending != I.ending); ++j)
42             if(!E[j].kind and E[j].I.ending < I.ending) I = E[j].I;
43
44         if(j == 2*n+1) break;
45         else A.push_back(I);
46     }

```

```
47 //Parte 2: computo de  $B = B_{i+1}$ 
48 interval B = {-2, -1};
49 for(j = D.size() > 1 ? D[D.size()-2].ending : 0; j < D.back().ending; ++j)
50     if(!E[j].kind and E[j].I.ending > B.ending) B = E[j].I;
51
52 //Parte 3: computo de  $D_{i+1}$ 
53 D.push_back(I.ending > B.ending ? I : B);
54 }
55
56 //Output del algoritmo ignorando el primer intervalo ficticio de D
57 cout << "D: ";
58 for(int i = 1; i < D.size(); i++) cout << D[i] << " ";
59 cout << endl;
60
61 return 0;
62 }
```

### 3. CONCLUSIONES

El problema de dominación eterna es un problema que, desde su reciente introducción por Burger et al. [3] y Goddard et al. [6], está atrayendo cada vez más atención desde la comunidad de teoría de grafos. En esta tesis, en particular, nos centramos en el estudio de este problema para grafos de intervalos, generalizando un resultado previo de Braga et al. [2]. A diferencia de lo que ocurre en los modelos de intervalos propios, los modelos de intervalos generales tienen intervalos largos que, de cierta forma, permiten que un guardia “salte” a una posición alejada en la recta real. Esta condición hace que no se pueda generalizar fácilmente la demostración de Braga et al. para este caso. Más aún, la demostración de Braga et al. es relativamente compleja, dado que utiliza algunas propiedades sobre grafos generales. El aporte principal de esta tesis es la determinación de la cantidad mínima de guardias (y la estrategia correspondiente) necesarios para defender un grafo de intervalos general, utilizando una demostración simple, directa y sucinta.

Repasando la demostración del Teorema 3, vemos que la misma es constructiva y brinda las estrategias óptimas tanto para atacar como para defender el grafo. El algoritmo resultante no sólo es óptimo en cuanto a su complejidad asintótica, también es simple de implementar y entender. En pocas palabras, los ataques se realizan sobre el intervalo que termine lo antes posible y no es alcanzable por las guardias que defendieron ataques previos. En cambio, los guardias se posicionan en los intervalos que lleguen a cubrir lo máximo posible dejando una brecha corta con el guardia anterior a fin de cubrir los eventuales ataques. Remarcamos que si bien el algoritmo y su demostración son simples, obtener este tipo de resultados suele ser un proceso creativo no trivial. Esto se observa en muchos trabajos en donde la obtención de métodos cada vez más simples conlleva un trabajo mayor.

Una buena dirección para seguir después de esta tesis es el estudio del problema de dominación eterna sobre grafos arco-circulares y sus subfamilias. En pocas palabras, los modelos arco circulares se obtienen reemplazando la recta real por un círculo. Específicamente, un modelo arco-circular es un par  $(C, \mathcal{A})$  donde  $\mathcal{A}$  es un conjunto de arcos sobre un círculo  $C$ . El grafo arco-circular representado por  $(C, \mathcal{A})$  tiene un vértice por cada  $A \in \mathcal{A}$  y dos vértices son adyacentes cuando sus respectivos arcos tienen intersección no vacía. Es fácil ver que todo grafo de intervalos es arco-circular. Más aún, considerando aquellos modelos arco-circulares en los que ningún arco contiene a otro, obtenemos la familia de los grafos arco-circulares propios que incluyen a los grafos de intervalos propios. Analizando la estrategia de ataque óptima para grafos de intervalos, observamos que, en cierta forma, cada ataque se aleja del ataque anterior a través de la recta real. Esto obliga a agregar guardias nuevos porque los usados para defender los primeros ataques no pueden llegar a defender los posteriores. La dificultad para estudiar los modelos arco-circulares es que dicha estrategia ya no funciona, porque al movernos suficientemente en una de las dos direcciones del círculo volvemos en algún momento al punto de partida. En consecuencia, el guardia que defiende el primer ataque puede empezar a moverse “preventivamente” en dirección contraria a la de los ataques a fin de repeler un ataque posterior.

## Bibliografía

- [1] J. Arquilla and H. Fredricksen. “Graphing” an optimal grand strategy. *Military Operations Research*, 1(3):3–17, 1995.
- [2] A. Braga, C. C. de Souza, and O. Lee. The eternal dominating set problem for proper interval graphs. *Inform. Process. Lett.*, 115(6-8):582–587, 2015. doi:[10.1016/j.ipl.2015.02.004](https://doi.org/10.1016/j.ipl.2015.02.004).
- [3] A. P. Burger, E. J. Cockayne, W. R. Gründlingh, C. M. Mynhardt, J. H. van Vuuren, and W. Winterbach. Infinite order domination in graphs. *J. Combin. Math. Combin. Comput.*, 50:179–194, 2004.
- [4] E. J. Cockayne, P. A. Dreyer, Jr., S. M. Hedetniemi, and S. T. Hedetniemi. Roman domination in graphs. *Discrete Math.*, 278(1-3):11–22, 2004. doi:[10.1016/j.disc.2003.06.004](https://doi.org/10.1016/j.disc.2003.06.004).
- [5] S. Finbow, S. Gaspers, M.-E. Messinger, and P. Ottaway. A note on the eternal dominating set problem. *Internat. J. Game Theory*, 47(2):543–555, 2018. doi:[10.1007/s00182-018-0623-0](https://doi.org/10.1007/s00182-018-0623-0).
- [6] W. Goddard, S. M. Hedetniemi, and S. T. Hedetniemi. Eternal security in graphs. *J. Combin. Math. Combin. Comput.*, 52:169–180, 2005.
- [7] M. C. Golumbic. *Algorithmic graph theory and perfect graphs*, vol. 57 of *Annals of Discrete Mathematics*. Elsevier Science B.V., Amsterdam, second edition, 2004.
- [8] W. F. Klostermeyer and G. MacGillivray. Eternal dominating sets in graphs. *J. Combin. Math. Combin. Comput.*, 68:97–111, 2009.
- [9] W. F. Klostermeyer and C. M. Mynhardt. Protecting a graph with mobile guards. *Appl. Anal. Discrete Math.*, 10(1):1–29, 2016. doi:[10.2298/AADM151109021K](https://doi.org/10.2298/AADM151109021K).
- [10] J. Mycielski. Sur le coloriage des graphs. *Colloq. Math.*, 3:161–162, 1955. doi:[10.4064/cm-3-2-161-162](https://doi.org/10.4064/cm-3-2-161-162).
- [11] M. Rinemberg and F. J. Souignac. The eternal dominating set problem for interval graphs. *Inform. Process. Lett.*, 146:27–29, 2019. doi:[10.1016/j.ipl.2019.01.013](https://doi.org/10.1016/j.ipl.2019.01.013).
- [12] I. Stewart. Defend the Roman Empire! *Sci. Am.*, 281:136–138, 1999. doi:[10.1038/scientificamerican1299-136](https://doi.org/10.1038/scientificamerican1299-136).