Plan de tesis doctoral del Lic. Eric Brandwein: Parámetros de ancho y algoritmos parametrizados en grafos

Directores: Dra. Flavia Bonomo y Dr. Ignasi Sau

2 de mayo de 2023

Resumen

Este plan se enfoca en complejidad parametrizada, en relación a los parámetros de ancho conocidos como *thinness* y *treewidth*. En pocas palabras, *complejidad parametrizada* es un marco para analizar cómo una medida secundaria de una instancia, llamada el *parámetro*, afecta el tiempo de ejecución de un algoritmo, por encima de la medida convencional definida por el tamaño de una instancia.

Los parámetros de ancho desempeñan un rol fundamental en la teoría estructural y algorítmica de grafos. La *thinness* es uno de los pocos parámetros que resulta acotado para grafos de intervalos, y por lo tanto permite generalizar sus propiedades algorítmicas. Se conocen algoritmos XP parametrizados por *thinness* para distintos problemas de optimización combinatoria, incluyendo clique máxima, conjunto independiente máximo, conjunto dominante mínimo.

El problema de decidir si la *thinness* de un grafo es a lo sumo k es NP-completo, pero la complejidad no se conoce para k constante al menos 2. Como parte de este plan, se estudiará la existencia de algoritmos polinomiales restringidos a una cierta clase, o parametrizados ya sea por la misma *thinness* o por otros invariantes, como el *treewidth* o el *vertex cover*. También se busca obtener cotas inferiores de complejidad bajo supuestos como la ETH.

1. Introducción

Desde el punto de vista teórico que adoptaremos en esta tesis, el diseño de algoritmos eficientes debe ir de la mano con un análisis riguroso basado en un marco matemático-computacional. Desafortunadamente, la mayoría de los problemas de optimización combinatoria que surgen tanto en la teoría como en la práctica resultan ser NP-completos. Al manejar tales problemas, a menos que P=NP, no podemos esperar un algoritmo de tiempo *polinomial* que *siempre* produce una solución *óptima* para *cualquier* instancia. En la actualidad existen varios paradigmas para eludir ese límite intrínseco de computabilidad, basados en relajar esos requisitos (eficiencia, determinismo, optimalidad, generalidad), tratando de optimizar el rendimiento en otros aspectos. Por ejemplo, el desarrollo de marcos para el diseño y análisis de algoritmos de aproximación, aleatorizados, moderadamente exponenciales, polinomiales para conjuntos de instancias particulares [9], o el enfoque de *complejidad parametrizada* [11, 12, 14], que es el enfoque principal de este plan de beca. En pocas palabras, complejidad parametrizada es un marco para analizar cómo una medida secundaria de una instancia, llamada el *parámetro*, afecta el tiempo de ejecución de un algoritmo, por encima de la medida convencional definida por el tamaño de una instancia. La idea central de la complejidad parametrizada es refinar la visión de complejidad agregando una o más dimensiones para medir la "dureza" de un problema.

Más precisamente, la complejidad parametrizada se ocupa de problemas y algoritmos cuya complejidad es medida no solamente basándose en el tamaño de la instancia, sino también en un valor entero k asociado a otra característica de la instancia, llamado el *parámetro*. Así, algoritmos que resuelven tal problema con complejidad temporal polinomial en el tamaño de la entrada al fijar como constante el parámetro, se dice que pertenecen a la clase de complejidad **XP**, o *slicewise polynomial* (complejidad $n^{f(k)}$, donde f es alguna función computable, n el tamaño de la instancia, y k el parámetro). Otra clase de complejidad más restrictiva, a la que apuntaremos con nuestros resultados algorítmicos, es la clase **FPT**, o *fixed parameter tractable*, en la que la complejidad requerida para los algoritmos es $f(k)poli(n)$, donde f es alguna función computable, *poli* un polinomio, n el tamaño de la instancia, y k el parámetro.

Los parámetros de ancho desempeñan un rol fundamental en la teoría estructural y algorítmica de grafos, y la jerarquía de parámetros definidos crece año a año. En términos generales, tales parámetros son invariantes que cuantifican qué tan "bien" se puede descomponer un grafo en una estructura de tipo árbol o camino. Entre ellos, el *treewidth* es probablemente el más relevante. Fue introducido por Robertson y Seymour en 1984 con fines estructurales en su proyecto seminal *Graph Minors* y desde entonces se ha convertido en una de las herramientas más útiles en el diseño de algoritmos parametrizados [18]. La *thinness* de un grafo es otro de los tantos parámetros de ancho de un grafo, definido por Mannino, Oriolo, Ricci y Chandran en 2007 [17], que está cobrando cada vez más relevancia. Es uno de los pocos parámetros que resulta acotado para grafos de intervalos, y por lo tanto permite generalizar las tantas propiedades algorítmicas de esa clase de grafos.

Formalmente, un grafo es **k-thin** si existe un orden de sus vértices y una partición de los mismos en k clases tal que para todo $v < w < z$ vértices, si v y w pertenecen a la misma clase y vz es una arista del grafo, entonces también lo es wz. La ***thinness*** de un grafo es el menor k tal que es k-thin.

Tanto para el *treewidth* como para la *thinness* se conocen algoritmos parametrizados para problemas de optimización combinatoria, incluyendo algunos de los más famosos, como clique máxima, conjunto independiente máximo, conjunto dominante mínimo. En general, los algoritmos parametrizados por *treewidth* son FPT [18], mientras que en el caso de la *thinness*, los algoritmos que se conocen son XP [1]. Más aún, el problema de coloreo es NP-completo para grafos de *thinness* 2 [3].

El problema de decidir si la *thinness* de un grafo es a lo sumo k es NP-completo [19]. Es sabido que los grafos de intervalos (equivalentes a los grafos de *thinness* 1) se pueden reconocer en tiempo polinomial [15], pero la complejidad del problema de reconocimiento de grafos de *thinness* k permanece abierto para k constante mayor o igual a 2.

Resulta relevante estudiar si el problema admite algoritmos polinomiales cuando se restringe el grafo de entrada a una cierta clase, y si el problema admite algoritmos polinomiales parametrizados ya sea por la misma *thinness* o por otros invariantes, como ser el *treewidth* o el *vertex cover*, que también es frecuentemente utilizado como parámetro para el cálculo de parámetros de ancho, entre otros problemas [10, 13]. Como parte de la tesis de licenciatura del candidato (publicada en [6]), se presentó un algoritmo polinomial para calcular la *thinness* de un bosque, y en [1] se computa la *thinness* de los cografos en tiempo polinomial. Intuimos que el problema parametrizado por *treewidth* resultará más desafiante que por *vertex cover*, ya que los árboles (*treewidth* 1) tienen thinness no acotada y su cómputo en árboles es no trivial, mientras que el *vertex cover* es una cota superior para la *thinness* de un grafo [5]. Algo intermedio puede llegar a ser parametrizar por *cutwidth*, *pathwidth* o *bandwidth*, que son parámetros lineales y cotas superiores de la *thinness* [17, 7] (más precisamente, la *thinness* está acotada superiormente por una función de ellos).

También se utilizará el concepto de *fine grained complexity* para obtener cotas inferiores de complejidad bajo supuestos razonables y ampliamente aceptados, como la ETH (Exponential Time Hypothesis) [16].

2. Objetivos

Yaroslav Shitov demostró en 2021 que decidir si un grafo tiene *thinness* a lo sumo k es NP-completo [19]. Sin embargo, la complejidad está abierta cuando k es constante (salvo en el caso $k = 1$ [8, 15]), lo cual lleva a los primeros objetivos:

- Encontrar un algoritmo eficiente para decidir si un grafo tiene *thinness* a lo sumo k, donde k es constante. Alternativamente, demostrar que dicho problema es NP-completo.

- De no ser posible encontrar el algoritmo del objetivo anterior, estudiar la existencia de un algoritmo eficiente parametrizado por k tal que, dado un grafo, si el grafo tiene *thinness* k, obtenga una representación de ancho a lo sumo $f(k)$ (f función fija).

- De no ser posible encontrar el algoritmo del primer objetivo, estudiar la existencia de un algoritmo eficiente parametrizado por k tal que, dada una representación de ancho a lo sumo k de un grafo, calcule la *thinness* exacta del grafo.

Un *bosque* es un grafo sin ciclos, o en otras palabras, un grafo tal que cada una de sus componentes conexas es un árbol. En [6], se presentó un algoritmo para encontrar la *thinness* de un bosque en tiempo polinomial. Los bosques no triviales son exactamente los grafos de *treewidth* 1, y por lo tanto de aquí surge el siguiente objetivo del plan:

- Encontrar un algoritmo eficiente para calcular la *thinness* de un grafo, parametrizado por su *treewidth*, o determinar que no existe tal algoritmo bajo suposiciones de complejidad razonables (ej. [16]). De no ser posible, encontrar algoritmos eficientes para valores fijos de *treewidth*, comenzando por los grafos de *treewidth* 2.

En esa tesis también, como corolario del algoritmo, se obtuvieron cotas para la *thinness* de un árbol en función de su cantidad de hojas. Un resultado similar vale para los grafos cordales, donde la *thinness* resulta estar acotada por la *leafage* menos 1 [4]. Los grafos de intervalo son los cordales de *leafage* 2. Eso lleva al siguiente objetivo:

- Encontrar un algoritmo eficiente para calcular la *thinness* de un grafo cordal, parametrizado por su *leafage*, o determinar que no existe tal algoritmo bajo suposiciones de complejidad razonables. De no ser posible, encontrar algoritmos eficientes para valores fijos de *leafage*, comenzando por los grafos cordales de *leafage* 4 (para los de *leafage* 3, se sabe que tienen *thinness* a lo sumo 2, y se puede reconocer en tiempo polinomial si tienen *thinness* 1).

También se sabe que en grafos de co-comparabilidad, la *thinness* está acotada por el número cromático [2], que además coincide con el tamaño de una clique máxima. Planteamos dos objetivos relacionados con eso:

- Encontrar un algoritmo eficiente para calcular la *thinness* de un grafo de co-comparabilidad, parametrizado por su número cromático (o clique máxima), o determinar que no existe tal algoritmo bajo suposiciones de complejidad razonables.

- Mejorar la cota conocida para grafos de co-comparabilidad, usando algún otro parámetro o combinación de parámetros, ya que esa cota vale por igualdad en algunos casos, pero puede ser arbitrariamente mala para, por ejemplo, grafos de intervalos.

El *vertex cover* es un parámetro que ha resultado útil a la hora de encontrar algoritmos parametrizados para el cálculo de otros parámetros de ancho [10], y es una cota superior para la *thinness* de un grafo (no vacío) [5]. Situaciones similares se dan para el *pathwidth*, *cutwidth*, *bandwidth*, y el *vertex cover* del complemento. Un objetivo del plan es el siguiente:

- Encontrar un algoritmo eficiente para calcular la *thinness* de un grafo, parametrizado por su *vertex cover*, *modular width*, *pathwidth*, *cutwidth*, *bandwidth*, y *vertex cover* del complemento. En este caso, tenemos indicios de que tal algoritmo existe, al menos para el *vertex cover* del grafo y de su complemento.

En [1], se presentaron algoritmos para resolver una gran cantidad de problemas NP-completos conocidos en tiempo polinomial cuando la *thinness* del grafo de entrada es acotada. Los algoritmos tienen complejidad en XP parametrizados por la *thinness*. El siguiente objetivo se centra en esta familia de problemas:

- Encontrar algoritmos de complejidad FPT para algunos problemas de esa familia, o determinar que no existen tales algoritmos bajo suposiciones de complejidad razonables. En ese caso, tratar de encontrar algoritmos XP más eficientes para resolver los mismos problemas en grafos de *thinness* acotada.

3. Metodología

Se comenzará por cursar materias relacionadas al tema, incluyendo materias de grafos y de algoritmia. Por ser el tema propuesto un tema que no es abarcado por las materias obligatorias de grado en la carrera, la primera parte del doctorado consistirá en ponerse al día con los últimos avances sobre la materia. Se leerá bibliografía adecuada (hay varios libros recientes sobre complejidad parametrizada, y un gran número de papers como para incorporar las técnicas típicamente utilizadas en esta clase de algoritmos) para poder avanzar en los objetivos propuestos, y se propondrán distintas posibles ramas de investigación. El punto de partida serán los objetivos propuestos en este plan y se irán adecuando a lo largo de los años las preguntas a los nuevos resultados obtenidos, ya sea por el candidato o por la comunidad internacional. Siendo un área de investigación muy activa, se intentará participar asiduamente de workshops y congresos en el tema. Para algunos de estos primeros objetivos ya tenemos alguna idea clara de un camino a la solución (por ejemplo, para el algoritmo que calcule la thinness de un grafo parametrizado por su vertex cover number). Para otros, contamos con los últimos avances en problemas relacionados, pero requieren ser pensados y que surjan ideas originales para resolverlos, y en algunos podrían terminar siendo demasiado difíciles. Es por eso que el número de objetivos es grande, y de hecho se espera que la tesis termine teniendo resultados relacionados con algunos de ellos, no necesariamente todos. Se investigará en las oficinas proporcionadas por la UBA, junto a la directora de doctorado Flavia Bonomo. Además, se realizarán reuniones periódicas con el co-director Ignasi Sau, HDR en el LIRMM, Université de Montpellier, Montpellier, Francia. Algunas se realizarán por video llamada, y otras en persona, a través de visitas mutuas financiadas por proyectos de cooperación o por el Laboratorio Asociado Franco-Argentino SINFIN.

4. Grupo de Investigación

Formaré parte del grupo de investigación que trabajará en el proyecto de Teoría Estructural y Algorítmica de Grafos, dirigido por Flavia Bonomo. El grupo cuenta actualmente en su rama en el DC con una becaria postdoctoral, dos tesistas doctorales, una investigadora graduada, y cinco tesistas de licenciatura.

5. Factibilidad

Se espera poder sostener al menos este primer año de investigación con un cargo de auxiliar dedicación exclusiva y obtener lugar de trabajo en el DC. El postulante se está presentando también a una beca UBACyT de doctorado. Se cuenta con buena parte de la bibliografía necesaria, y como se describe en la sección de metodología, tenemos ya ideas preliminares para atacar algunos de los objetivos. El candidato cuenta también con antecedentes en el tema.

Referencias

[1] F. Bonomo and D. De Estrada. On the thinness and proper thinness of a graph. *Discrete Appl. Math.*, 261:78–92, 2019.

[2] F. Bonomo, S. Mattia, and G. Oriolo. Bounded coloring of co-comparability graphs and the pickup and delivery tour combination problem. *Theor. Comput. Sci.*, 412(45):6261–6268, 2011.

[3] F. Bonomo-Braberman, E. Brandwein, F. S. Oliveira, M. Sampaio, A. Sansone, and J. Szwarcfiter. Thinness and its variations on some graph families and coloring graphs of bounded thinness. arXiv:2303.06070 [math.CO], March 2023.

[4] F. Bonomo-Braberman, N. Brettell, A. Munaro, and D. Paulusma. Classifying the thinness of H-graphs. Manuscript.

[5] F. Bonomo-Braberman, C. Gonzalez, F. S. Oliveira, M. Sampaio, and J. Szwarcfiter. Thinness of product graphs. *Discrete Appl. Math.*, 312:52–71, 2022.

[6] F. Bonomo-Braberman, E. Brandwein, C. L. Gonzalez, and A. Sansone. On the thinness of trees. In I. Ljubić, F. Barahona, S. Dey, and A. R. Mahjoub, editors, *Proc. ISCO 2022*, volume 13526 of *Lect. Notes Comput. Sci.*, pages 189–200, 2022.

[7] F. Bonomo-Braberman and G. A. Brito. Intersection models for 2-thin and proper 2-thin graphs. In *Proc. XI LAGOS 2021*, volume 195 of *Proc. Comput. Sci.*, pages 221–229, 2021.

[8] K. Booth and G. Lueker. Linear algorithms to recognize interval graphs and test for the consecutive ones property. In *Proc. 7th STOC*, pages 255–265, Las Vegas, 1975.

[9] A. Brandstädt, V. Le, and J. Spinrad. *Graph Classes: A Survey*, volume 3 of *SIAM Monographs on Discrete Mathematics*. SIAM, Philadelphia, 1999.

[10] M. Chapelle, M. Liedloff, I. Todinca, and Y. Villanger. Treewidth and pathwidth parameterized by the vertex cover number. *Discrete Appl. Math.*, 216(1):114–129, 2017.

[11] M. Cygan, F. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.

[12] R. G. Downey and M. R. Fellows. *Fundamentals of parameterized complexity*, volume 4 of *Texts in Computer Science*. Springer, 2013.

[13] F. Fomin, M. Liedloff, P. Montealegre, and I. Todinca. Algorithms parameterized by vertex cover and modular width, through potential maximal cliques. *Algorithmica*, 80:1146–1169, 2018.

[14] V. Garnero, C. Paul, I. Sau, and D. M. Thilikos. Explicit linear kernels via dynamic programming. *SIAM J. Discrete Math.*, 29(4):1864–1894, 2015.

[15] M. Habib, R. McConnell, C. Paul, and L. Viennot. Lex-BFS and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing. *Theor. Comput. Sci.*, 234(1-2):59–84, 2000.

[16] D. Lokshtanov, D. Marx, and S. Saurabh. Lower bounds based on the exponential time hypothesis. *Bull. EATCS*, 105:41–72, 2011.

[17] C. Mannino, G. Oriolo, F. Ricci, and S. Chandran. The stable set problem and the thinness of a graph. *Oper. Res. Lett.*, 35:1–9, 2007.

[18] N. Robertson and P. Seymour. Graph minors. II. Algorithmic aspects of tree-width. *J. Algorithms*, 7(3):309–322, 1986.

[19] Y. Shitov. Graph thinness is NP-complete. Manuscript, 2021.