# MODELOS SOBRE ÁRBOLES DE UNIÓN, QUE OTROS LLAMAN DE EXPANSIÓN

**Francisco R. Fernández García**
Departamento de Estadística e Investigación Operativa
Universidad de Sevilla

## 1. Árboles de unión

Esta breve exposición sobre árboles de unión pretende poner de manifiesto la importancia que los mismos tienen tanto en la literatura matemática, como en sus aplicaciones. Somos conscientes que cada sección que desarrollamos se corresponde con colecciones de excelentes libros, por lo que la bibliografía que acompaña tampoco pretende ser completa. Escribimos, pues, el artículo sólo con la intención de sacar a la luz, especialmente para los no estudiosos del tema, esta parcela tan importante de la optimización combinatoria.

La teoría de grafos permite modelar muchos problemas cotidianos. En especial, los árboles de unión asociados a un grafo desempeñan un papel crucial en problemas de conexiones en circuitos electrónicos, no es difícil usar la imaginación para ver aplicaciones de los mismos en problemas relacionados con las redes de ordenadores e incluso en las conexiones interiores de un mismo ordenador.

Empleados originalmente en esquemas de clasificación y en búsqueda de cuestionarios óptimos, hoy día son usados como base en los protocolos para el intercambio de mensajes en las redes de comunicación. La historia de los árboles de unión y sus aplicaciones es muy dilatada y fructífera, una excelente exposición puede verse en el artículo de Graham y Hell [6].

Recordemos que un árbol de unión, asociado a un grafo conexo dado (con $n$ nodos), es cualquier subgrafo conexo de él que contiene todos los nodos del mismo, y no tiene ciclos.

Estas estructuras se caracterizan de muy diversas formas:

- Un subgrafo conexo de $n-1$ aristas.
- Un subgrafo sin circuitos de $n-1$ aristas.
- Un subgrafo con un único camino entre cualquier par de nodos del grafo.

Si suponemos que partimos de un grafo completo con $n$ vértices, conocido como $K_n$, una cuestión planteada ya en el siglo XIX fue, ¿cuantos árboles hay asociados a $K_n$?. Sabemos que son $n^{n-2}$ los posibles árboles de unión que tiene asociados el grafo, expresión que se conoce como fórmula de Cayley. Recordemos que fue él quien introdujo los árboles cuando estaba intentando contar el número de isómeros de hidrocarburos saturados. Existen varias pruebas de esta fórmula, véase [10]. La primera prueba detallada la dio Prüfer, y la basó en encontrar una biyección entre el conjunto de árboles de unión de $K_n$ y el conjunto de vectores de dimensión $n-2$, en el que cada una de sus componentes son números enteros entre 1 y $n$.

Escoger entre todos los árboles de unión asociados a un grafo, algunos con determinadas propiedades, debería ser sencillo, pues bastaría con enumerarlos todos y escoger los que verifiquen las propiedades deseadas. Este tipo de problemas son estudiados por la matemática discreta, que desde fechas recientes algunos matemáticos han empezado a desarrollar dada su complejidad computacional y la importancia de sus aplicaciones, aunque no aparece en las célebres listas de problemas matemáticos con proyección futura.

Vamos a considerar algunas de las propiedades que suelen exigirse a los mismos, y cómo estudiando las propiedades estructurales que los árboles tienen en relación a ellas, podemos o encontrar los árboles deseados en tiempo razonable, o en caso contrario debemos contentarnos con árboles que están próximos en algún sentido. Generalmente las aproximaciones nos darán árboles cuyo valor asociado será como mucho una proporción del valor del árbol solución óptima. Debemos indicar que sólo nos referiremos a los llamados grafos no dirigidos, para el caso de grafos dirigidos muchos de los problemas aun están por desarrollar.

## 2. Árboles de unión de mínimo coste

Una de las propiedades que se imponen es la de tener mínimo coste (longitud), entendido por coste del árbol la suma de los costes asociados a las aristas que lo definen. Los árboles asociados a un grafo que verifican esta propiedad se denominan árboles de unión de mínimo coste.

Dada la naturaleza matemática que tienen estos árboles de unión, los algoritmos miopes (o voraces), que en cada etapa escogen la mejor decisión, aunque con ello no exploren globalmente todas las decisiones posibles, son óptimos para la resolución de estos problemas. Esto se debe a que se verifica que si tememos construido un subárbol de un árbol de unión óptimo, al unirle una de las aristas de menor coste, tendremos un nuevo subárbol de un árbol de unión óptimo.

No es frecuente que estos procedimientos den buen resultado, más bien casi nunca lo dan, pero cuando funcionan son muy rápidos para buscar la solución del problema. Así todos los algoritmos para encontrar los árboles de unión de coste mínimo, basados en el anterior principio, Bor*ü*vka, Prim, Kruskal, son algoritmos polinomiales, en el sentido del número de operaciones matemáticas que debemos realizar para obtener la solución óptima. Para una descripción de estos algoritmos, su complejidad y los problemas abiertos entorno de ellos, véase [11].

Dadas las propiedades que presentan estos árboles de unión de mínimo coste, no es extraño que sean utilizados para ayudar a buscar soluciones para otros problemas. Consideremos los problemas, en los que se busca un circuito con cierta propiedad sobre el grafo, conocidos como problema de cubrimiento de aristas (problema del cartero chino) y problema de cubrimiento de nodos (problema del viajante).

### 2.1. El problema del cartero chino

Dado un grafo, el problema de cubrimiento de aristas consiste en buscar un circuito que contenga todas las aristas del grafo y que sea de coste mínimo. Es decir, el cartero debe recorrer todas las calles, al menos una vez, minimizando la distancia.

Si el grafo es euleriano, es decir, si existe un circuito que contenga una sola vez a cada arista, éste será la solución del problema. Euler estableció las condiciones necesarias y suficientes para que un grafo sea euleriano: *Todos los nodos tienen un número par de aristas incidentes.* El número de aristas incidentes en un nodo se conoce como grado del mismo.

Para los grafos eulerianos es sencillo usar algoritmos polinomiales para la búsqueda de la solución del problema, véase [8]. Es interesante comprobar que los algoritmos rápidos para dar solución manual a estos problemas, así como para resolver los pasatiempos del tipo "dibuje la siguiente figura sin levantar el lápiz del papel", no son los mejores para ser usados en un ordenador, lo que es de gran interés para la investigación matemática. Los algoritmos implementables en ordenador se deducen directamente de la prueba de Euler del teorema citado.

Si el grafo no es euleriano, para buscar la solución óptima nos basamos en la propiedad que nos dice que *el número de nodos con grado impar es siempre par*. Usando esta propiedad convertimos el grafo dado en un grafo euleriano al añadir al mismo las aristas que nos proporciona cualquier emparejamiento mínimo, sobre el grafo, de los nodos de grado impar. Dado que existen algoritmos polinomiales parar encontrar emparejamientos mínimos, véase [3], cuando apliquemos los algoritmos para grafos eulerianos al nuevo grafo transformado, tendremos la solución deseada, en tiempo total polinomico.

### 2.2. Los árboles de unión y el problema del viajante

Dado un grafo, el problema de cubrimiento de nodos consiste en buscar un circuito que contenga todos los nodos del grafo, y que sea de coste mínimo. Es decir, el viajante de comercio debe visitar todas las ciudades con un menor coste de su recorrido.

Si el grafo es hamiltoniano, es decir, si existe al menos un circuito que contenga una sola vez a cada nodo, éstos podrían ser candidatos a ser solución del problema. Pero desconocemos condiciones necesarias y suficientes para que un grafo sea hamiltoniano, que puedan ser usadas para su implementación, como en el caso de los circuitos eulerianos.

Para su resolución podemos recordar que cualquier árbol de unión de mínimo coste, asociado al grafo, contiene todos los nodos. Por lo que podemos usar dicho árbol como solución del problema, pues por duplicación de las aristas del mismo, tendríamos un circuito, que como mucho tiene un valor doble del verdadero circuito solución del problema.

No obstante existen mejores aproximaciones. Una que emplea los árboles de unión de coste mínimo se debe a Christofides, véase [3]. Ésta, parte de un árbol de unión de coste mínimo, y en lugar de duplicar todas las aristas, realiza un emparejamiento entre los nodos del árbol que tienen grado impar, añadiendo las aristas correspondientes al árbol de partida. Este nuevo grafo tiene un circuito euleriano, fácil de determinar, que es un circuito que pasa por todos los nodos del grafo primitivo, aunque recorre varias veces un mismo nodo, pero tiene como mucho un valor 3/2 del verdadero circuito solución del problema.

Cuando existen millones de puntos que conectar en un único circuito, cualquier aproximación es bien recibida, y muy en especial cuando no podemos encontrar para este problema su solución óptima. No es extraño, que dada la importancia que el problema tiene en sus aplicaciones reales, se hayan desarrollado muchos algoritmos heurísticos para la resolución del mismo, véase [9].

### 2.3. Árboles de Steiner

Consideremos un grafo plano completo definido por un conjunto de $n$ nodos y sus posibles conexiones rectilíneas.

Cuando el árbol de mínimo coste, pueda usar, además de los nodos dados otros nodos de conexión (conocidos como puntos de Steiner) y aristas que no pertenezcan al grafo original, decimos que buscamos un árbol de Steiner de coste mínimo.

Lógicamente, ello será interesante si dada una estructura de nodos en un grafo, los árboles de Steiner óptimos tienen un coste inferior a los árboles de unión clásicos. Esto es así, trivialmente, como contestación al celebre problema de Fermat:

*Dados tres puntos en el plano, encontrar un cuarto punto , tal que la suma de las distancias a los otros tres puntos sea mínima.*

Este cuarto punto es un punto de Steiner, y en este caso solo hay uno, ya que se conoce que el máximo número de puntos de Steiner, dados $n$ puntos, es de en $n-2$. Y sólo en el caso de que el triángulo que determinan sea obtusángulo, ambos problemas de árboles de unión tendrán igual solución, y por ello valor.

Este problema es de más difícil solución que el clásico, ya que pertenece a la clase NP. La dificultad radica en la determinación del número y posición de los puntos de Steiner asociados al problema. Existen algoritmos para encontrar estos árboles de Steiner de mínimo coste, pero a medida que el número de nodos aumenta se hacen inoperantes. Ello puede ser importante en casos reales, pensemos en una red telefónica que debe unir todos los puntos de una gran región, la conexión mínima la proporciona el árbol de Steiner asociado.

#### 2.3.1. Relación con los árboles de unión

Una pregunta interesante en estos casos es ¿si resolvemos el problema de conexión usando la solución clásica en lugar del árbol de Steiner en cuanto nos podemos desviar como mucho? La contestación a este tipo de preguntas no es generalmente fácil, ya que desconocemos el valor de la solución del árbol Steiner óptimo para los problemas. No obstante, en este caso la pregunta ha sido respondida, usando argumentos de Teoría de Juegos, indicando que el cociente entre ambos valores óptimos es como mucho $\sqrt{3}/2$, valor alcanzado en el problema de Fermat, antes citado, cuando los puntos definan un triángulo equilátero.

#### 2.3.2. Árboles de Steiner rectangulares

Para problemas específicos, las respuestas pueden ser más concretas. Si el grafo de partida tiene sus aristas definidas por líneas horizontales y verticales, como ocurre en muchas situaciones que describen circuitos electrónicos, se conoce un conjunto finito de candidatos a posibles puntos de Steiner. Estos puntos son los puntos de intersección de la red definida por las líneas horizontales y verticales que pasan por todos los nodos del grafo, que estén dentro de la envolvente convexa de los nodos originales. Partiendo de esta información, podemos buscar soluciones mejores que la que nos proporciona el árbol de unión clásico. No obstante el problema sigue siendo de complejidad no polinomial, aunque su aproximación por el árbol de unión clásico es solo de 2/3, véase [4].

## 3. Árboles de unión con conexión mínima

Otra propiedad que suele pedirse para los árboles de unión es que sus aristas sean individualmente lo menor posible. Dado que si escogemos las $n-1$ menores aristas, éstas no formaran un árbol de unión, se pone como objetivo el de buscar el árbol cuya arista de mayor coste sea lo menor posible. Por razones obvias, llamamos a estas soluciones árboles de unión con conexión mínima. Suelen ser de gran interés al construir conexiones para transportar líquidos, que necesitan cierta presión entre los nodos directamente conectados.

El lector puede buscar algoritmos polinomiales sencillos para resolver este problema, basándose en lo anteriormente expuesto.

## 4. Árboles de caminos mínimos

Si fijamos un nodo cualquiera en el grafo, el árbol de union de coste mínimo tiene la propiedad de que la suma de las aristas que lo componen es mínima. No obstante, puede ocurrir que un nodo del grafo más próximo que otro nodo al nodo destacado, tenga sobre el árbol de unión óptimo un camino de conexión con dicho nodo de mayor coste.

Por tanto, otra propiedad que le podemos pedir al árbol es, dado un nodo origen, la de tener mínimo coste desde cualquier nodo a dicho nodo origen, entendido, nuevamente, como coste la suma de los costes de las aristas que forma el camino para ir de un nodo al origen. Esta propiedad es equivalente a desear que la suma de los costes desde el origen a cada uno de los restantes nodos sea mínima. Los árboles asociados a un grafo que verifican esta propiedad se denominan árboles de unión de caminos mínimos. En los problemas asociados a las redes de alta velocidad, es fácil ver las aplicaciones de estos árboles a la distribución la información lo más rápidamente posible desde el centro de producción de la misma a cada uno de sus usuarios.

Este problema tiene como caso particular al de encontrar el camino de mínimo coste entre dos nodos dados. Situación que se nos presenta en un navegador cuando le pedimos que nos de la ruta óptima entre dos ciudades. Lógicamente, en este caso el algoritmo miope no funciona, como todos hemos experimentado al buscar las mejores rutas.

Pero, aunque estos árboles de caminos mínimos no verifican la anterior propiedad de los árboles de unión, conservan parte de ella, pues un subárbol de un árbol de caminos óptimos es un subárbol óptimo. Sin embargo debemos de tener presente más información para conocer las aristas que podemos añadir en cada paso.

Una contestación a este problema del árbol de caminos mínimos es dada con el algoritmo de Dijkstra. Si buscamos la distancia del nodo destacado y el nodo más distante de él, nos proporciona, con complejidad polinomial, el árbol deseado de caminos mínimos.

Hemos supuesto que todas las aristas tienen asociados costes positivos, pero si alguno fuese negativo, dicho algoritmo puede que no nos proporcione la solución deseada. En este caso, la propiedad que se verifica, y que antes hemos indicado, nos permite formular este problema como un problema de programación dinámica, lo que conduce a su solución usando el algoritmo de Bellman-Ford [8].

### 4.1. Puntos mediana de un grafo

Si el nodo a destacar debe ser determinado por el criterio del tipo que Fermat impuso, de que la suma de los costes a los otros restantes puntos sea mínima, la búsqueda del mismo es sencilla, ya que una modificación del algoritmo de Dijkstra, usualmente atribuido a Floyd, nos permite encontrar los costes entre todos los nodos del grafo [8], por lo que encontrar el nodo buscado es trivial. Estos nodos son conocidos como nodos mediana del grafo.

Si el punto a destacar puede ser cualquiera de los nodos o de los puntos sobre las aristas, estos puntos se conocen como puntos mediana general, y entonces el problema comienza a complicarse, aunque Hakimi caracterizó un conjunto finito de puntos que son candidatos para solución del problema, véase [8], al ser la función objetivo cóncava. Dicho conjunto está determinado sólo por los nodos del grafo.

Con ello el problema del árbol de unión de caminos mínimos es resoluble en tiempo polinomial, tanto si el nodo origen está dado, como si deseamos usar el nodo que dé menor suma de longitudes.

#### 4.1.1. Problemas con varias medianas

El problema anterior se extiende de forma natural al caso de destacar un conjunto de $p$ nodos, como puntos de distribución, lo que conduce al problema de la $p$-mediana, en el que interviene un nuevo elemento combinatorio por la elección de los $p$ nodos-mediana entre los $n$ posibles, lo que eleva la dificultad computacional del problema. Estos problemas, en general, pertenecen a la clase NP.

## 5. Nuevos tipos de árboles de unión, no menos interesantes

Debido a sus relaciones con las situaciones planteadas en Internet, no es extraño que hayan aparecido recientemente nuevos objetivos para escoger árboles de unión, asociados especialmente a problemas de comunicación en la red, en función de los usuarios que en cada momento estén compartiendo la información. Así se desean encontrar árboles de unión cuyo tiempo (coste) medio de comunicación entre cualquier par de vértices del mismo sea mínimo. Ello es equivalente a minimizar el tiempo total de comunicación entre todos los pares de vértices.

Notemos que si atravesar cada arista del árbol tiene asociado un coste (tiempo, dinero,dificultad,..), estamos resolviendo el problema de minimizar el coste de envío de mensajes entre cualquier par de vértices. Este problema es conocido, por tanto, como el problema del árbol de unión de ruta mínima.

En un principio podíamos pensar que la solución la daría el árbol de unión de caminos mínimos, pero no es así. Incluso este problema, a diferencia del anterior, pertenece a la clase de los problemas NP. No obstante, existen relaciones entre los problemas:

El árbol de unión de mínima longitud, respecto de la mediana del grafo, tiene un valor como mucho del doble del valor de la solución óptima del problema del árbol de unión de ruta mínima.

Aún mas difícil es estudiar estos problemas cuando existan pesos (importancia, fiabilidad,...) asociados a las aristas del grafo que se atraviesan, e incluso varios nodos como posibles centros de distribución.

### 6. Árboles de unión con varios objetivos

Cada arista puede estar valorada en función de varios criterios, esto produce que cualquier problema anteriormente considerado, deba de ser formulado como un problema multiobjetivo. Por lo que debemos de hablar en este caso de árboles de unión eficientes, respecto de los criterios considerados.

Lamentablemente la buena propiedad estructural que se tiene para los árboles de unión a coste mínimo se pierde para los árboles de unión eficientes. Es tal la pérdida de propiedades estructurales que el problema pertenece a la clase NP, véase[7].

También llegamos al mismo tipo de problemas cuando deseemos considerar simultáneamente varios de los criterios que anteriormente estudiamos, dado que los mismos conducen a diferentes propiedades para la solución. De este modo los árboles de unión a coste mínimo son interesantes cuando deseamos construir una conexión entre los diferentes nodos, pero cuando pensamos en el "tráfico"que debe soportar, los árboles de caminos mínimos son preferibles. Por ello se pueden buscar soluciones de compromiso para estas situaciones, algunas de las cuales poseen algoritmos polinomiales para su solución, [11].

Más complejo,aún, sería el caso de que los costes asociados fuesen aleatorios, siendo especialmente importante el caso de que éstos estén distribuidos uniformemente en un cierto intervalo.

### 7. Repartos y construcción de árboles de unión

Una vez determinado el árbol de unión, cuando los nodos representan diversos agentes (o usuarios de la red), se ocasiona un nuevo problema, al tener que decidir como dividir el coste total asociado al árbol, entre los diferentes usuarios. Esto conduce a un modelo de juego cooperativo, tanto en el caso de existir un solo criterio como en el caso multicriterio, véase [2],[5].

Considérese una red que une varios agentes y un nodo de distribución, que está modelada por un grafo. Cada vértice está asociado a un agente y cada arco del grafo tiene asociado un coste. Asociada a ella se define un juego cooperativo en el que las coaliciones están formadas por subconjuntos de agentes y el nodo de distribución, y cuyo valor está determinado por el coste del árbol mínimo que puedan formar. En este juego se buscan repartos del árbol de unión del grafo que pertenezcan al core del juego, dadas las propiedades de estabilidad que estos elementos tienen.

Uno de estos repartos viene dado por la regla de Bird [2], en la que cada agente debe de abonar el coste de la arista unida a su nodo, en el camino que en el árbol óptimo lo une el nodo de distribución. En este caso encontramos repartos pertenecientes al core del juego en tiempo polinomial.

El análisis anterior parte de que se tiene un acuerdo a priori para construir un árbol de unión a coste mínimo, por lo que el coste está dado, y se desea realizar su reparto *ex post.* Sin embargo, si se desea permitir a los agentes que participen en la construcción del árbol de unión *ex ante*, se debe plantear el problema como un juego no cooperativo cuya solución de equilibrio lleve al reparto deseado, véase [1].

Para establecer el proceso en el que los agentes irán uniéndose para conectarse al nodo de distribución, de forma óptima, se define un juego no cooperativo multietápico, asociado al grafo y a su matriz de costes asociada a todos los pares de nodos.

Inicialmente todos los agentes están desconectados. En la primera etapa cada agente decide si conectarse o no al origen. Si ningún agente se conecta, o todos deciden conectarse, el juego termina. En las sucesivas etapas los agentes no conectados deciden conectarse a los ya conectados, o permanecer desconectados. El juego termina cuando todos los agentes están conectados a la fuente, o ningún agente se ha conectado en dicha etapa.

En cada etapa el jugador que se conecta debe pagar el coste de la conexión efectuada. Cuando el juego termina, los agentes que no estén conectados tienen una penalización muy elevada, ya que no podrán unirse al árbol formado.

En este juego, una estrategia es tipo Prim, si el jugador escoge su arista de unión cuando el proceso secuencial le indique que debe tomar la decisión, en base al orden inicial y a las decisiones tomadas por los restantes jugadores en etapas anteriores, de unirse al árbol formado según indica el algoritmo de Prim.

Es de fácil demostración que estas estrategias definen un equilibrio fuerte de Nash. Por lo que la obtención de estos equilibrios se hace también en tiempo polinomial, y sus pagos coinciden, para cada jugador,con los proporcionados por la regla de Bird, para el juego cooperativo.

Notése que estos protocolos pueden emplearse para la construcción de redes de distribución en las que los individuos pueden hacer su elección sin tener acuerdos previos con los restantes agentes, como ocurría en el caso cooperativo. Estos protocolos son de gran aplicación en teoría económica, y están relacionados con lo que se conoce como *programa de Nash*.

### Referencias


[1] Bergantiños G., Lorenzo L. (2004). A non-cooperative approach to the cost spanning tree problem. *Mathematical Methods of Operations Research*, **59**,393-403.

[2] Bird C.G. (1976). On Cost Allocation for a Spanning Tree: A Game Theoretic Approach. *Network*, **6**, 335-350.

[3] Christofides N. (1975). *Graph Theory. An Algorithmic Approach*, Academic Press, London.

[4] Du D-Z., Pardalos P.M. (Eds.) (2005). *Handbook of Combinatorial Optimization. Vol B*, Springer.

[5] Fernández F.R., Hinojosa M.A., Puerto J. (2004). Multi-criteria Minimum Cost Spanning Tree Games. *European Journal of Operations Research*, **158**(2), 399-408.

[6] Graham, R.L., Hell P. (1985). On the history of the minimum spanning tree problem. *Ann. Hist. Comput.*, **7**, 43-57.

[7] Hamacher H.W., Ruhe G. (1994). On Spanning tree Problems with Multiple Objectives. *Annals of Operations Research*, **52**, 209-230.

[8] Larson R.C., Odoni A.R. (1981). *URBAN Operations Research*, Prentice-Hall.

[9] Lawler E., LenstraJ.K., Rinnoy Kan A., Shmoys D. (1985). *The Traveling Salesman Problem*, Wiley.

[10] Moon J.W. (1967). *Various Proofs of Cayley´s Formula for Counting Trees*, Chapter II in *A Seminar on Graph Theory* (F. Harary, ed.). Holt,Rinehart and Winston Inc.

[11] Wu, B.Y., Kao K-M. (2004). *Spanning Trees and Optimization Problems.* Chapman Hall/CRC.