Un algoritmo exacto para el problema del árbol generador con el número máximo de hojas

Sergio Alonso, Miguel Ángel Domínguez-Ríos, Marcos Colebrook y Antonio Sedeño-Noda

Departamento de Estadística e Investigación Operativa,
Universidad de La Laguna

Resumen

Desarrollamos en este trabajo un método exacto tipo ramificación y acotación, para la construcción del árbol generador con el número máximo de nodos-hoja, basado en estrategias enumerativas sobre el conjunto de árboles generadores de un grafo conexo.

Palabras Clave: Maximum-Leaf Spanning Tree.

AMS: 90B10, 05C05.

1. Introducción

Dado un grafo no dirigido y conexo, *G=(V,A)*, llamaremos a todo subgrafo que conecte los nodos de V sin ciclos, (conexo y acíclico), árbol generador de G. El conjunto de árboles generadores de un grafo conexo G, al que denotaremos por $\sigma(G)$, tiene un cardinal que, en el peor caso, el de los grafos completos, llega a ser n^{n-2}. De ahí la dificultad de trabajar con técnicas enumerativas sobre este conjunto, como hacemos en este trabajo.

Dado un grafo, decimos que un nodo es hoja (*leaf*), si su grado es uno. El problema del árbol generador con el máximo número de hojas, trata de encontrar entre todas las conexiones posibles de los nodos, aquel árbol generador con un mayor número de nodos terminales o hojas. Es por tanto, un problema de diseño de redes, en el que se optimiza indirectamente el coste y la fiabilidad de la interconexión. Esto es debido a que, si un nodo terminal deja de funcionar, no afecta al resto de la red (fiabilidad). Además, un nodo no terminal implica una inversión superior al tener, además, responsabilidades de transmisión y comunicaciones, que no posee uno simplemente terminal (coste). Este problema es NP-completo [2].

En [1] se desarrolla un algoritmo, tipo ramificación y acotación, sobre una modelización adecuada del problema para programación entera. Además se afirma no haber encontrado otro método exacto para el problema planteado en la bibliografía. En este trabajo, diseñamos un algoritmo exacto para la construcción del *maximum leaf spanning tree* (*MLST*), basado en las propiedades de los árboles dentro de la teoría de grafos.

2. Desarrollo

La dificultad de diseñar un algoritmo exacto para este problema, es encontrar un adecuado método enumerativo de árboles generadores. En nuestro caso, nos interesan las secuencias de subárboles de la forma,

$$T_0 \subset T_1 \subset T_2 \subset \ldots \subset T_{p-1} \subset T_p,$$

donde $T_0 = \emptyset$ y $T_p \in \sigma(G)$.

El paso de un elemento de la secuencia al siguiente se realiza de forma sencilla. Consideramos las aristas del grafo original que conectan un nodo de T_{k-1}, con otro no conectado aún. Los subconjuntos que mantengan la no existencia de ciclos de las aristas seleccionadas podrán ser añadidas al subárbol para obtener, T_k.

Al ser subárboles, se mantiene la propiedad de conectividad y no ciclos sobre un subconjunto de nodos, por lo que podemos considerar como de *tipo Prim* el diseño de construcción elegido, (véase [3]). Además, si por nh denotáramos la función que nos devuelve el número de hojas de un grafo, se verificará que $nh(T_{k-1}) \leq nh(T_k)$.

Sobre esta estructura se simplifica la toma de decisiones sobre la continuación o no, del desarrollo de secuencias que *a priori* no van a mejorar el árbol con mayor número de hojas obtenido hasta ese momento.

3. Contactar

e-mail: salonso@ull.es

4. Agradecimientos

Este trabajo ha sido desarrollado parcialmente bajo la convocatoria 2003 de proyectos para grupos precompetitivos de la Universidad de La Laguna.

5. Bibliografía

[1] Fujie, T. (2003). *An exact algorithm for the maximum leaf spanning tree problem.* Computers & Operations Research, Vol 30, No 13, Pags 1931-1944.

[2] Garey M. R.;Johnson, D. S. (1979). *Computers and Intractability: A guide to the theory of NP-completeness.* W. H. Freeman, San Francisco.

[3] Prim RC. (1957). *Shortest connection networks and some generalizations.* Bell Systems Technical Journal, Vol 36, Pages 1389-401.