Spectral Approximation Algorithms for Graph Cut Problems

Giuseppe Ottaviano

Il problema del massimo taglio in un grafo (MaxCut) consiste nel trovare una partizione dei nodi in due parti tali che gli archi con un estremo nella partizione e un estremo nell'altra siano massimizzati. È uno dei problemi combinatoriali su grafi più naturali e interessanti. Sfortunatamente come molti dei problemi interessanti è NP-completo: è uno dei 21 problemi di cui Karp ha dimostrato la NP-completezza nell'articolo del 1972 che ha gettato le basi della teoria delle riduzioni.

Esclusa quindi l'esistenza di algoritmi efficienti ed esatti (assumendo che P ≠ NP), la ricerca sui problemi di ottimizzazione combinatoria si è spostata su due fronti duali:

- Lo sviluppo di *algoritmi approssimati*, cioè che trovano una soluzione subottimale al problema, ma dimostrabilmente vicina alla soluzione ottima.
- La teoria dell'inapprossimabilità, che studia la massima precisione che possono raggiungere gli algoritmi polinomiali.

Qualora si dimostri che un problema non è approssimabile oltre un certo limite teorico (solitamente una funzione dell'ottimo), e che tale limite è raggiunto da un algoritmo di approssimazione, la complessità approssimata del problema è essenzialmente risolta, in quanto sia l'algoritmo che il risultato di inapprossimabilità sono *ottimali*.

In questa tesi descriviamo un algoritmo presentato in un recente lavoro di Trevisan che sfrutta tecniche spettrali per l'approssimazione di MaxCut. L'obiettivo di questa tesi è fornire una sua analisi sperimentale che dimostra che l'algoritmo è applicabile a problemi di grandi dimensioni.

Per poter sviluppare la tesi, richiameremo il concetto di Laplaciano di un grafo e alcuni risultati elementari di teoria spettrale dei grafi. Definiremo i problemi di SparsestCut e *edge expansion*, le disuguaglianze alla Cheeger e l'algoritmo di partizionamento spettrale, che sono le prime applicazioni storiche della teoria spettrale e che hanno ispirato l'algoritmo spettrale per MaxCut.

Nel seguito concentreremo l'attenzione su MaxCut. Descriveremo l'algoritmo di Goemans e Williamson, che in un articolo del 1995 hanno presentato la prima approssimazione non banale, con precisione $\alpha_{GW} = \min_{0 \le \theta \le \pi} \frac{2}{\pi} \frac{\theta}{1 - \cos\theta} = 0.878\ldots$, cioè che restituisce una soluzione di valore almeno α_{GW} volte l'ottimo. L'algoritmo si basa su un rilassamento geometrico di un problema di programmazione quadratica intera equivalente a MaxCut. Tale problema è formulabile come programma semidefinito, quindi risolubile in tempo polinomiale. La soluzione del rilassamento viene quindi *arrotondata* così da ottenere una soluzione ammissibile del programma quadratico intero.

Non è stato scoperto alcun algoritmo che garantisca un rapporto di approssimazione al caso pessimo migliore di quello di Goemans-Williamson. Vedremo come la *Unique Games Conjecture*, una recente congettura di Khot et al., implicherebbe l'ottimalità dell'algoritmo semidefinito. Tale congettura è adoperata per la costruzione di un verificatore PCP (*Probabilistically Checkable Proof*) riducibile a MaxCut. L'analisi del verificatore si basa su una congettura recentemente dimostrata: il teorema *Majority is Stablest*, sulle proprietà estremali delle funzioni monotone $\{-1,1\}^n \longrightarrow \mathbb{R}$. Altro ingrediente fondamentale è l'analisi di Fourier delle funzioni finite, un campo recentemente in fermento per le sue applicazioni in combinatoria e informatica teorica.

Infine presenteremo l'algoritmo spettrale per MaxCut. A differenza degli approcci tipo Goemans-Williamson, basati su programmazione semidefinita, esso fa uso dell'autovettore principale del Laplaciano per trovare un'immersione del grafo nella retta. Tramite un algoritmo lineare di arrotondamento è quindi possibile estrarre dall'immersione sottografi con buoni tagli, e ripetere ricorsivamente il procedimento. Il rapporto di approssimazione raggiunto è almeno $0.507\ldots$.

Rispetto all'algoritmo semidefinito, l'algortimo spettrale ha un'analisi elementare ed è facilmente implementabile, visto che l'unico passo complesso consiste nel calcolo di un autovettore di una matrice sparsa quanto il grafo. Sfruttando librerie esistenti per il calcolo numerico di autovettori di matrici sparse, è possibile applicare l'algoritmo a grafi di dimensioni consistenti. Presenteremo quindi un'implementazione e un'analisi sperimentale, che mostra che l'algortimo è molto efficiente in pratica, e che pone nuovi problemi aperti sulla sua analisi teorica.

Relatori:
Prof. Roberto Grossi, Università di Pisa
Prof. Luca Trevisan, University of California - Berkeley