|  |
| --- |
| The Open University |
| **MAXIMUM ACYCLIC SUB GRAPH PROBLEM** |
| **Advantage over Random** |
|  |
| **Alexander Stekolchik** |
| **‏2009** |

|  |
| --- |
| New approximation algorithm for the MAXIMUM ACYCLIC GRAPH PROBLEM based on work "On the advantage over Random from Maximum Acyclic Subgraph by Moses Charikar Konstantin and Yury Makarychev 2007" |

Content

[1. Introduction 3](#_Toc219700518)

[1.1. Abstract 3](#_Toc219700519)

[1.2. Applications 3](#_Toc219700520)

[1.3. Current status 3](#_Toc219700521)

[1.4. Approximation with status 1/2 3](#_Toc219700522)

[1.5. Good Cases 4](#_Toc219700523)

[1.6. Definitions 4](#_Toc219700524)

[1.7. Examples 5](#_Toc219700525)

[1.8. Main result 5](#_Toc219700526)

[2. Approximation Algorithm 6](#_Toc219700527)

[2.1. Adjacency matrix 6](#_Toc219700528)

[2.2. Proof interpretation 6](#_Toc219700529)

[2.3. Efficient Implementation 7](#_Toc219700530)

[3. Cut Norm of Skew Symmetric Matrices 8](#_Toc219700531)

[4. References 9](#_Toc219700532)

## Introduction

## Abstract

Given a directed graph G = (V,A) the maximum acyclic subgraph problem is to compute a subset A’ of arcs of maximum size or total weight so that G’ = (V,A’) is acyclic. Equivalently find an ordering of the vertices so as to maximize the number of edges going forward. The problem is complement to minimum feedback arc set.

## Applications

* ordering alternatives by group voting
* determining of a hierarchy of the sectors of an economy
* determining ancestry relationships
* analysis of systems with feedback
* scheduling problems

## Current status

Exact solution is NP hard ( R.M. Karp, Reducibility among combinatorial problems )

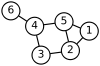
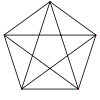
Approximation with factor ½ is simple, see paragraph [Approximation with status 1/2](#_Approximation_with_status)

The question is whether it is possible to beat this ½ approximation

## Approximation with status 1/2

Apply arbitrary sort of vertices   
Find out group the edges connecting vertices with small values to vertices with big values A1 = {(u,v) A | u < v }  
Find out group of the edges connecting vertices with big values to vertices with small values A2 = {(u,v) A | u > v }  
Max {w (A1) , w (A2) } >= 0.5 w (A)

## Good Cases

* Planar graph (polynomial solvable 1981)  
  (a planar graph is a graph which can be embedded in the plane, i.e., it can be drawn on the plane in such a way that its edges intersect only at their endpoints.)  
  
* If measure OPT as fraction of edges, then if OPT is 1 – ε for very small ε possible use approximation algorithm for Min FeedBack Arc Set . Using O( log n log log n) algorithm of Seymour instances where OPT is 1 – ε and   
  ε = O(1/(log n log logn) See Lior Kamma lecture from 26/11/2008 Approximation ratio is   
  ½ + Ω(1/ O(1/(log n log logn) )

## Definitions

Let G = (V,E) be a directed graph on n vertices and let π :{1..n} be a linear arrangement then the advantage or gain over random of the arrangement π is equal to the fraction of edges going forward minus fraction of the edges going backward gain(G, π )

If a linear arrangement has value 0.5 + δ then the gain of this arrangement is 2 δ

## Examples

* Random ordering  
   gain(G, π ) = 3/5 – 2/5 = 0.2
* OPT  
   gain(G, π ) = 4/5 – 1/5 = 0.6

## Main result

Given an instance with optimal gain δ   
can we produce a solution with gain   
f(δ) for all values δ Є ( 0,1 )

**Theorem 1.3** There exists a randomized polynomial time algorithm that given a directed graph G finds a linear arrangement π of its vertices with gain over random at least   
Ω ( δ / log n )

Main ideas are

* There is a connection between the advantage over random and the cut norm of the adjacency matrix of the graph G.
* The result shows how ordering information from one distance scale in the optimum solution can be exploited algorithmically

## Approximation Algorithm

## Adjacency matrix

Define WG for unweighted matrix as follows



The gain over random is equal to



|E| denotes total number of all edges

* Partition vertices of the graph to into three sets A,B,C in special way
* Permute values in each set
* With probability of ½ output vertices in order A,B,C and C,A,B.
* All edges from A to B going forward, from B to A backward all other with probability ½ or forward or backward

Expected gain is



If π is an optimal ordering then always exists disjoint sets A,B for which



## Proof interpretation

The goal of the analysis is to show if there is an ordering π with gain δ then there are subsets of vertices A and B of such that placing all vertices of A before vertices of B gives gain at least δ/logn.

Define the length of an edge to be a distance between its end points in the ordering π.

It possible to group edges by length into O(logn) groups. If the gain of the ordering is δ then atleast one of these groups must have gain δ/logn

The sets A, B constructed as result of random sampling the positions in the ordering π. The selection probabilities vary periodically with position  
This sampling is incorporated to certain bilinear form. The expression involves terms xk(r) yk(r) that corresponds to selected sets A and B.  
For appropriate choice of period constructed bilinear form must have value at least δ/logn.

## Efficient Implementation

Cut norm of a matrix W(u,v) is



It is not required A and B to be disjoint, but given arbitrary sets A and B it possible always find disjoint sets A’ and B’



Partition vertices into two disjoint sets X and Y



Lemma 2.2 for every skew symmetric matrix G, the sets A’ and B’ are disjoint and satisfy



* A’ and B’ are disjoint since X and Y are disjoint
* For every u A and v B probability that   
   u A’ and v B’ is ¼

**Theorem 2.3** (Alon and Naor) There exists a randomized polynomial time algorithm that given a matrix W(u,v) finds two subsets of indices A and B such that  
  
  
  
  
αAN ~ 0.56



## Cut Norm of Skew Symmetric Matrices

**Theorem 3.1** let W be n x n skew symmetric matrix.  
Define   
S+ = S+(W) = Σ wkl ; 1<= k < l <= n

Then



Remark gain (G,π ) = S+(WG) / |E|

**Lemma 3.2** let Ŝt be discrete Fourier sine transform of a sequence S1…Sn-1 defined as follows



then



**Lemma 3.3** Let W be an n x n skew symmetric matrix  
Define



For 1 <=k<=n-1 and let Ŝt bethe discret Fourier sine transform of Sk

Then



Where   
Let t0 = argmaxt |Ŝt| for every k,l from 1 to n and r from 0 to n-1 define xk(r) = sin (π ( k + r ) t0 / n ) , yk(r) = -cos (π ( k + r ) t0 / n )

**Corollary 3.4**



The maximum of bilinear form is attained at a vertex of the cube

The lemma 3.3 allows find reverse function between gain |Ŝ| and ordering wklxkyl since valuesxk,yl  depend on ordering in our case optimal ordering.

* Lemma 3.5



**Proof of theorem 3.1**

Let



Then S+ = Σ Sk

By lemma 3.2 and Corollary 3.4



By lemma 3.5



Missing proves see here.  
Moses Charikar, Konstantin Makarychev, Yury Makarychev   
On the Advantage over Random for Maximum Acyclic Subgraph (2007**)**

## References

* Moses Charikar, Konstantin Makarychev, Yury Makarychev   
  On the Advantage over Random for Maximum Acyclic Subgraph (2007)
* Refael Hassin, Shlomi Rubinstein  
  Approximations for the maximum acyclic subgraph problem (1994)
* Prasad Raghavendra, Venkatesan Guruswami , Rajsekar Manokaran   
  Beating the Random Ordering is Hard: Inapproximability of Maximum Acyclic Subgraph (2008)
* R.M. Karp, Reducibility among combinatorial problems, in: R.E. Miller and J.W. Thatcher, eds.,Complexity of Computer Computations (Plenum, New York, 1972) 85-10
* Noga Alon, Assaf Naor Approximating the cut-norm via Grothendieck's inequality (2004)
* Harold N. Gabow. A representation for crossing set families with applications to submodular flow problems (1993)
* Moses Charikar Anthony Wirth Maximizing Quadratic Programs: Extending Grothendieck's Inequality(2004)