Topics in Algorithms 2005

Constructing Well-Connected Networks
via Linear Programming and Primal Dual Algorithms

Ramesh Hariharan

The Steiner Tree Problem: A Primal Dual Approach

Given a subset S of vertices which need to be connected together, find the least cost connection.

min Σ c_e x_e

for each cut splitting S:

Σ x_e >= 1

for each edge e:

0<=x_e

max_{a_{S}} Σ_{S} a_{S}

Σ_{S,e crosses S} a_{S} <= c_e

a_{S}>= 0

The Primal Dual Approach

- Find feasible primal and dual solutions
- Dual Solution serves as a lower bound
- Primal Solution is integral and serves as the final answer

$\min \Sigma c_e x_e$

for each cut splitting S:

$\Sigma x_e >= 1$

for each edge e:

$0 <= x_e$

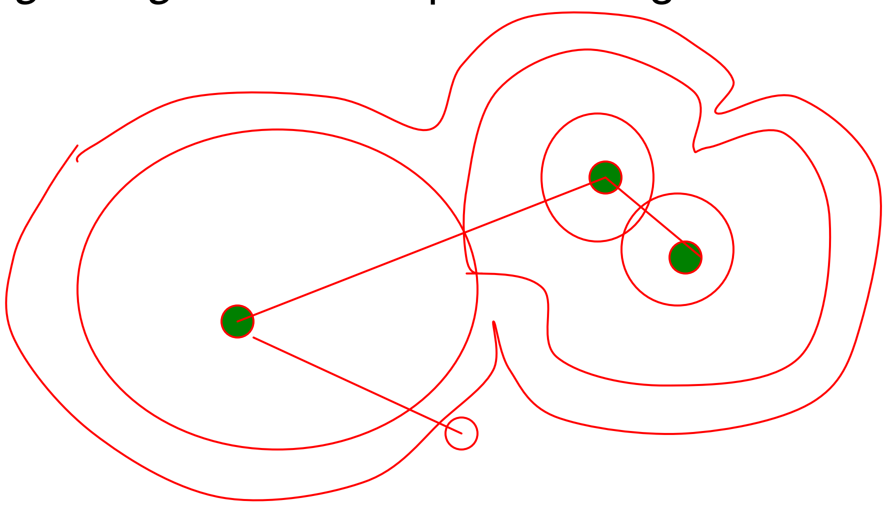
$\max_{a_{S}} \Sigma_{S} a_{S}$

$\Sigma_{S, e \text{ crosses } S} a_{S} <= c_e$

$a_{S} >= 0$

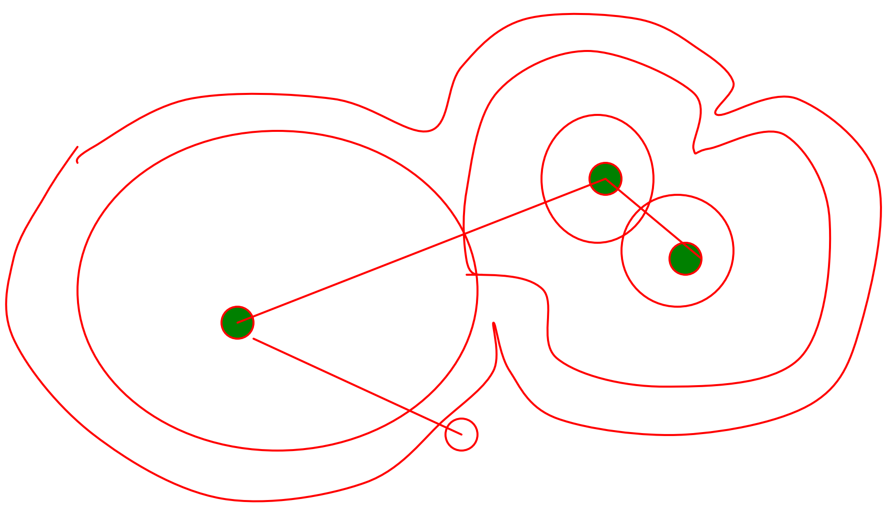
Goemans-Williamson Ball Growing

- Grow balls around each steiner vertex
- All balls grow at the same rate
- If two balls collide, they fuse and grow together after that
- The annulus width of a ball is the a_S value
- The total annuli sizes intersecting an edge do not exceed c_e
- *Tight edges* form the primal integer solution (with some pruning)



Goemans-Williamson Ball Growing

- Sum of annuli sizes is the dual solution, serves as a lower bound
- The primal solution has each annulus multiplied by the degree of that annulus
- Show that this is at most 2 times the sum of annuli sizes.



The Generalized Steiner Network Problem

Given a weighted graph G and a set of requirements, find a minimum cost subgraph satisfying these requirements

Requirements: For each subset S of vertices (i.e., a cut), how many edges f(S) must cross this cut

Kamal Jain shows a 2 factor approximation for this problem

The Linear Program

Select the least weight subset of edges satisfying all requirements

min Σ c_e x_e

for each subset S of V:

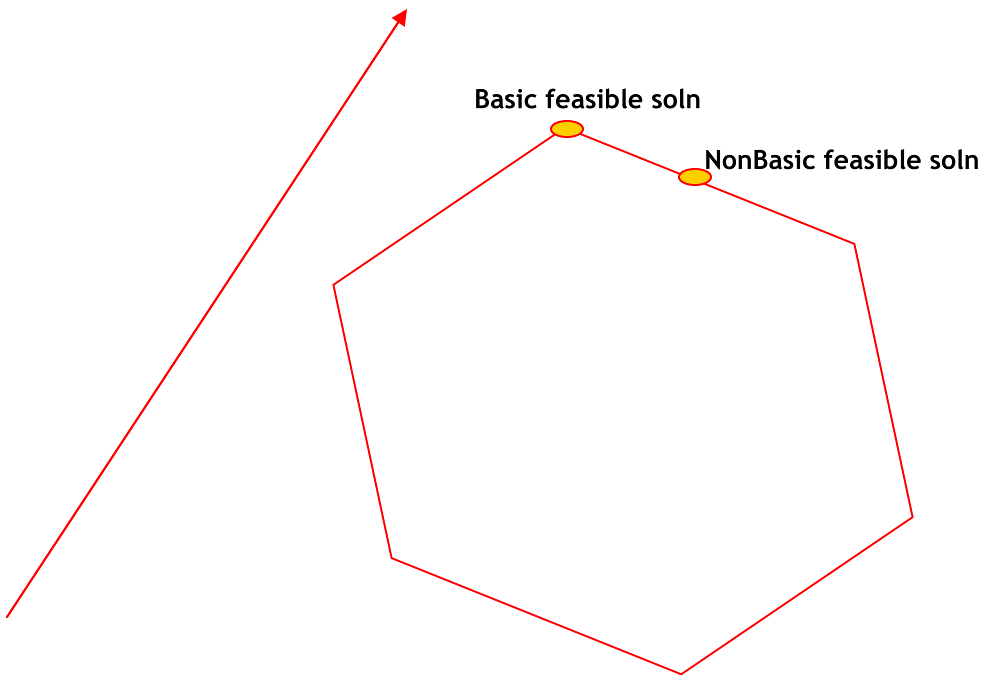
Σ x_e >= f(S)

for each edge e:

$0 \le x_e \le 1$

Linear Programming

How does one solve a linear program? Later…



Interpreting Linear Programming Results

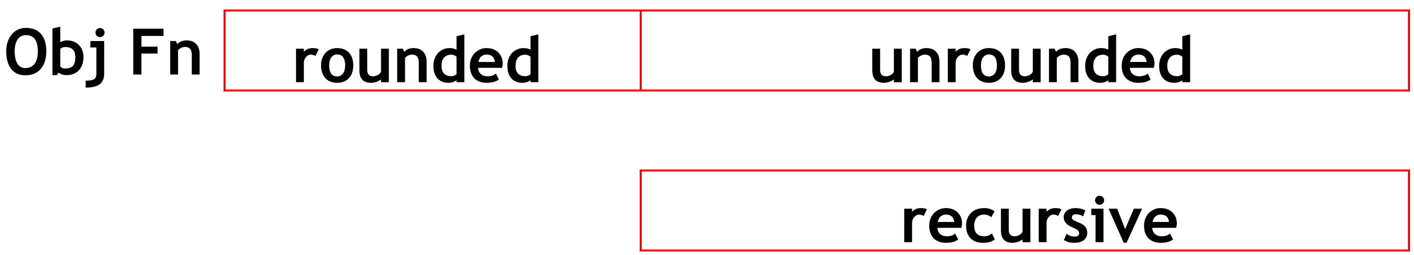
How does one get a solution from the linear program results?

Problem: x_e need not be 0/1, could be fractional

Claim: At least one x_e is at least .5 for basic feasible points

- So What: Set this x_e to 1, remove all requirements satisfied in the process and recurse.
- How do we show that this results in a 2 factor approximation to the optimum?

Rounding Effectiveness Proof



- The objective function value coming from the LP is a lower bound. Why?
- The recursive objective function value is at most the unrounded objective function value. Why??
- 2 factor follows. Why??

Proving the claim

- Based on tight set structure
- Tight Cuts: Cuts which have requirements exactly met

1. A,B are tight then

    - either A-B and B-A are tight

      A-B and B-A together have the same linear span as A,B

    - or A cap B and BUA are tight

      A cap B and BUA together have the same linear span as A,B

- Prove this??

Proving the claim

Corollary:

- Consider any family of tight cuts
- This can be decomposed into a laminar family with the same linear span. Prove this??
- The laminar family can also be chosen to make the cuts in the family linearly independent
- The dimension of this linear span equals E', the number of edges which get a fractional weight (i.e., non-0 weight, assuming no edge gets weight 1) ; this assumes that the solution is a basic feasible solution (bfs)
- It follows that the number of cuts in the laminar family equals E'

Proving the claim

Show a 1/3 bound rather than 1/2 : show ½ as an exercise

- Consider only the E’ edges with fractional non-1 weights in the analysis below
- If some cut in the family has at most 3 edges then done
- Otherwise, each cut has at least 4 edges
- Then show that the total count of edges is strictly greater than the number of cuts in the laminar family.

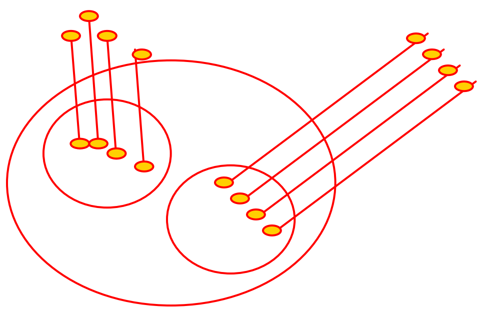
Proving the claim

Show a 1/3 bound rather than 1/2 : show ½ as an exercise

- Then show that the total count of edges is strictly greater than the number of cuts in the laminar family.

- Show that the endpoints of the E' edges can be distributed such
that each cut gets 2 endpoints and every maximal cut in the family gets 4 endpoints; why is this sufficient?

- Go bottom up on the cuts; leaves have 4 edges so they use two of the endpoints and send two upwards. Each cut with at least two child cuts gets 2 endpoints from each child, it uses two and send two upwards

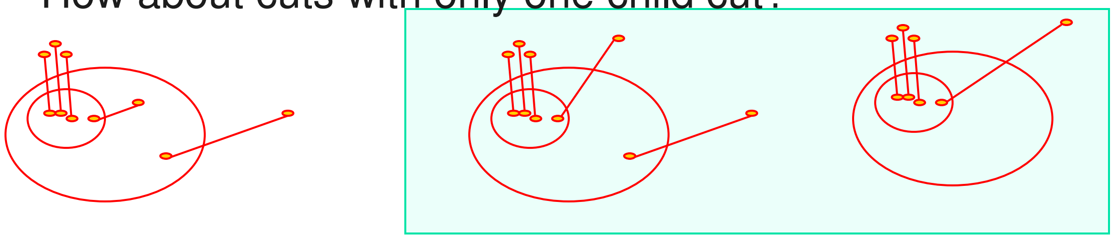


- How about cuts with only one child cut?

Proving the claim

Show a 1/3 bound rather than 1/2 : show ½ as an exercise

- How about cuts with only one child cut?



Solving linear programs

- Can be solved in polynomial time via Ellipsoid (expensive in practice) and Karmarkar’s methods. Later..

Solving large linear programs

- What if the number of constraints is very large: for instance, each vertex v wishes to be connected to some specified set of vertices with a certain number of paths
- Can still do polynomial time if we have the following procedure:

check if a given solution is feasible or not; if not show a violated constraint

how does one do this?

Open Problem

- Linear programming is expensive, can we do without that to get a 2 factor approximation. Best known is goemans, williamson, vazirani, mihail, a not constant factor

The Linear Program for the Generalized Steiner Problem

Select the least weight subset of edges satisfying all requirements

min Σ c_e x_e

for each subset S of V:

Σ x_e >= f(S)

for each edge e:

0<=x_e<=1

Exercise: Write the dual; Why does the ball growing approach not work. Where does it get stuck?

References

- Kamal Jain’s paper on the generalized steiner network problem
- Goemans and Williamson’s paper on the Steiner Tree problem
- Goemans,Williamson,Vazirani,Mahail on the Steiner Network Problem

All available on the web