Rank-2 Relaxation Heuristics for MAXCUT and Others

Yin Zhang
Rice University

Joint work with:
Sam Burer and Renato Monteiro
Georgia Institute of Technology

Novel Approaches to Hard Discrete Optimization

Waterloo, Canada, April 27, 2001

Outline

o Maxcut formulation

o Semidefinite Relaxation

o Rank-2 relaxation: Properties and Heuristic

o Numerical comparison

o Concluding thoughts

Maxcut as an quadratic program

Given graph $G(V, E)$ and edge weights $\{w_{ij}\}$, find a cut of V that maximizes the sum of weights on in-between edges:

$$\begin{array}{ll} \max & \frac{1}{2}\sum_{i>j} w_{ij}(1 - x_i x_j), \\ \text{s.t.} & |x_i| = 1, \;\; i = 1, \cdots, n. \end{array}$$

Maxcut is NP-hard.

Applications:

o Ising model of spin glasses in physics
o Circuit board design
o Network design
o Equivalence to other problems
o Simple and interesting

Semidefinite (SDP) Relaxation:

$$\begin{aligned} \text{max} \quad & \tfrac{1}{2}\sum w_{ij}(1 - x_i x_j), \\ \text{s.t.} \quad & |x_i| = 1, x_i \in \Re \end{aligned}$$

Replace scalars x_i by vectors $v_i \in \Re^n$:

$$\begin{aligned} \text{max} \quad & \tfrac{1}{2}\sum w_{ij}(1 - v_i^T v_j), \\ \text{s.t.} \quad & \|v_i\|_2 = 1, v_i \in \Re^n \end{aligned}$$

A solution is n points on the unit sphere.

Let $X = V^T V$, where $V = [v_1\, v_2\, \cdots\, v_n]$,

$$\begin{aligned} \text{min} \quad & W \bullet X, \\ \text{s.t.} \quad & \text{diag}(X) = e \\ & X \succeq 0. \end{aligned}$$

This is an SDP, solvable in polynomial time.

SDP Relaxation $\rightarrow$ Upper bound

Randomized Algorithm $\rightarrow$ Lower bound

Each unit vector (point on the unit sphere) represents a node. Any random halving of the unit sphere gives a random cut.

<u>Performance Guarantee:</u>

Goemans and Williamson (1994):

$$E(\texttt{random cuts}) \geq 0.878 * (\texttt{Optimal Cut})$$

A ground-breaking theoretical result, followed by many works in this direction.

Semidefinite relaxation has become a powerful theoretical tool for combinatorial optimizaton.

Difficulties in practice:

SDPs are expensive to solve:
— $O(n^2)$ variables or constraints
— $O(n^2)$ storage

Interior-point methods:
— $O(n^6)$ flops/iter in the worst case

Several less expensive alternatives have been developed, but still nono is cheap.

Our Motivation:

To find a technique:
— scalable to large-scale problems;
— based on a similar but "fast" relaxation;
— still uses GW randomization approach.

This leads to rank-2 relaxation.

Max-Cut:

$$\begin{array}{ll} \max & \frac{1}{2}\sum w_{ij}(1 - x_i x_j), \\ \text{s.t.} & |x_i| = 1, x_i \in \Re \end{array}$$

A rank-2 relaxation:

Replace scalars x_i by vectors $v_i \in \Re^2$:

$$\begin{array}{ll} \max & \frac{1}{2}\sum w_{ij}(1 - v_i^T v_j), \\ \text{s.t.} & \|v_i\|_2 = 1, v_i \in \Re^2 \end{array}$$

Polar Coordinates:

$$v_i = [\cos(\theta_i)\ \sin(\theta_i)]^T,\ v_i^T v_j = \cos(\theta_i - \theta_j).$$

Then,

$$\begin{array}{ll} \max & \frac{1}{2}\sum w_{ij}(1 - v_i^T v_j), \\ \text{s.t.} & \|v_i\| = 1, v_i \in \Re^2 \end{array}$$

becomes

$$\max\ \frac{1}{2}\sum_{ij\in E} w_{ij}[1 - \cos(\theta_i - \theta_j)],$$

n variables, unconstrained, but **nonconvex**. (There is no free lunch.)

Or equivalently

$$\min f(\theta) \equiv \sum_{ij\in E} w_{ij}\cos(\theta_i - \theta_j).$$

A solution gives n points on the unit circle.

Properties of

$$f(\theta) \equiv \sum w_{ij} \cos(\theta_i - \theta_j)$$

o Every cut is a stationary point of f.

o All cuts are saddle points with 1 exception.

o Only max cut can possibly be a local min.

(For some graphs, the max cut can indeed be a minimum of f.)

Algorithmic Idea:

Local min $\rightarrow$ Saddle point $\rightarrow$ New local min

min f $\rightarrow$ Get a cut $\rightarrow$ min f near the cut

Cut Generation:

$\theta \in \Re^n$ corresponds to n point on the unit circle. Any halving of the circle gives a cut (GW).

No Randomness Now: *For any given n points on the unit circle, it only takes $\mathcal{O}(n)$ work to examine all possible cuts and find the best.*

If $\frac{1}{2}(\sum w_{ij} - f(\theta))$ is an upper bound for the optimal cut value, then the 0.878 performance guarantee will hold, but ......

Algorithm: (input N, θ^0, output a cut)
Let $k = 0$.
While $k \leq N$

1. From θ^0, minimize f to get θ.
2. Find the best cut associated with θ.
3. Set θ^0 to a perturbation of the cut.
4. If the best cut so far is improved, set $k = 0$; otherwise set $k = k + 1$.

End

Geometric interpretation:

From a local minimum, find the lowest saddle point nearby; restart from the saddle point to find another, hopefully lower, minimum; continue until no progress seen for N consecutive times.

Algorithm can be restarted from different θ^0.

State of Affair: SDP + GW randomization approach has been reported to generate good-quality cuts, often better than many heuristics (but it is slow).

If our approach can get better cuts than SDP at a much lower cost, then it should be competitive with many heuristics.

Comparisons with SDP codes:

Our code, named **CirCut**, is written in F90 and run on an SGI Origin2000 (with 300 MHZ R12000 processors) at Rice University.

SBmethod: Spectral Bundle method
by Helmberg, ZIB in Germany; run on the SGI Origin2000 at Rice.

DSDP: Dual interior-point method
by Benson, Choi, Ye and Zhang, University of Iowa; run on a HP9000/785/C360 (367 MHZ processor) at Iowa.

Concluding Remarks

Is SDP useful for finding good-quality feasible solutions (lower bounds) in practice? — It does not appear promising for now.

We are curious to know a class of graphs on which SDP relaxation approach does better in terms of quality.

Rank-2 relaxation heurestics seem promising, and can be applied to other problems.

Comparisons with some heurestics are planned.

Statistics for the Torus Set of Max-Cut Problems

Graph	Size	Lbound	Ubound
pm3-8-50	(512,1536)	456*	461*
pm3-15-50	(3375,10125)	2988	3069*
g3-8	(512,1536)	41684814	41684814*
g3-15	(3375,10125)	2.842e+8	2.877e+8*

(* – From Michael Juenger and Frauke Liers)

Comparison with SBmethod on the Torus Set

Graph	SBmethod		CirCut ($N = 4$, $M = 100$)		
No.	Value	Time	Avg V	Avg T	Best V
1	434	29	443	0.2	452
2	2728	2132	2888	2.3	2936
3	4.05e+7	36	4.09e+7	0.3	4.14e+7
4	2.73e+8	3605	2.74e+8	2.8	2.78e+8

Time for Max-Cut Problems (I)

Graph	Size	DSDP	CirCut
G11	(800, 1600)	16.61	0.055
G12	(800, 1600)	17.66	0.063
G13	(800, 1600)	18.19	0.055
G14	(800, 4694)	35.23	0.090
G15	(800, 4661)	32.11	0.090
G20	(800, 4672)	32.03	0.113
G21	(800, 4667)	37.56	0.078
G22	(2000, 19990)	4123.31	0.363
G23	(2000, 19990)	3233.52	0.371
G24	(2000, 19990)	3250.70	0.297
G30	(2000, 19990)	3718.93	0.316
G31	(2000, 19990)	3835.70	0.332
G32	(2000, 4000)	142.61	0.176
G33	(2000, 4000)	132.48	0.141
G34	(2000, 4000)	156.66	0.117
G48	(3000, 6000)	343.64	0.270
G49	(3000, 6000)	303.30	0.270
G50	(3000, 6000)	264.59	0.172
G55	(5000, 12498)	1474.77	0.539
G56	(5000, 12498)	15618.60	0.457
G57	(5000, 10000)	1819.76	0.480
G60	(7000, 17148)	58535.13	0.730
G61	(7000, 17148)	52719.63	0.508
G62	(7000, 14000)	5187.23	0.473
G64	(7000, 41459)	102163.92	0.941
G70	(10000, 9999)	33116.22	0.367
G72	(10000, 20000)	12838.07	0.715
G77	(14000, 28000)	32643.36	0.949
G81	(20000, 40000)	131778.21	1.492
Total		**467589.73**	**11.516**

Value for Max-Cut Problems (I)

Graph	**DSDP**	**CirCut**	**Ratio**
G11	542	524	1.0344
G12	540	512	1.0547
G13	564	536	1.0522
G14	2922	3016	0.9688
G15	2938	3011	0.9758
G20	838	901	0.9301
G21	841	887	0.9481
G22	12960	13148	0.9857
G23	13006	13197	0.9855
G24	12933	13195	0.9801
G30	3038	3234	0.9394
G31	2851	3146	0.9062
G32	1338	1306	1.0245
G33	1330	1290	1.0310
G34	1334	1276	1.0455
G48	6000	5676	1.0571
G49	6000	5734	1.0464
G50	5880	5748	1.0230
G55	9960	10000	0.9960
G56	3634	3757	0.9673
G57	3320	3202	1.0369
G60	13610	13765	0.9887
G61	5252	5429	0.9674
G62	4612	4486	1.0281
G64	7624	8216	0.9279
G70	9456	9280	1.0190
G72	6644	6444	1.0310
G77	9418	9108	1.0340
G81	13448	12830	1.0482
Mean	———	———	**1.0011**

Time for Max-Cut Problems (II)

Graph	Size	DSDP	CirCut
G11	(800, 1600)	16.61	3.883
G12	(800, 1600)	17.66	3.758
G13	(800, 1600)	18.19	3.453
G14	(800, 4694)	35.23	5.531
G15	(800, 4661)	32.11	5.914
G20	(800, 4672)	32.03	5.563
G21	(800, 4667)	37.56	5.563
G22	(2000, 19990)	4123.31	22.313
G23	(2000, 19990)	3233.52	18.852
G24	(2000, 19990)	3250.70	27.297
G30	(2000, 19990)	3718.93	23.766
G31	(2000, 19990)	3835.70	19.609
G32	(2000, 4000)	142.61	13.086
G33	(2000, 4000)	132.48	12.617
G34	(2000, 4000)	156.66	9.820
G48	(3000, 6000)	343.64	13.125
G49	(3000, 6000)	303.30	11.430
G50	(3000, 6000)	264.59	15.711
G55	(5000, 12498)	1474.77	39.727
G56	(5000, 12498)	15618.60	33.516
G57	(5000, 10000)	1819.76	32.227
G60	(7000, 17148)	58535.13	56.750
G61	(7000, 17148)	52719.63	63.570
G62	(7000, 14000)	5187.23	47.039
G64	(7000, 41459)	102163.92	67.555
G70	(10000, 9999)	33116.22	94.391
G72	(10000, 20000)	12838.07	86.586
G77	(14000, 28000)	32643.36	109.414
G81	(20000, 40000)	131778.21	140.461
Total		**467589.73**	**992.527**

Value for Max-Cut Problems (II)

Graph	DSDP	CirCut	Ratio
G11	542	554	0.9783
G12	540	552	0.9783
G13	564	572	0.9860
G14	2922	3053	0.9571
G15	2938	3039	0.9668
G20	838	939	0.8924
G21	841	921	0.9131
G22	12960	13331	0.9722
G23	13006	13269	0.9802
G24	12933	13287	0.9734
G30	3038	3377	0.8996
G31	2851	3255	0.8759
G32	1338	1380	0.9696
G33	1330	1352	0.9837
G34	1334	1358	0.9823
G48	6000	6000	1.0000
G49	6000	6000	1.0000
G50	5880	5856	1.0041
G55	9960	10240	0.9727
G56	3634	3943	0.9216
G57	3320	3412	0.9730
G60	13610	14081	0.9666
G61	5252	5690	0.9230
G62	4612	4740	0.9730
G64	7624	8575	0.8891
G70	9456	9529	0.9923
G72	6644	6820	0.9742
G77	9418	9670	0.9739
G81	13448	13662	0.9843
Mean	———	———	**0.9606**