An Approximate, Efficient Solver for LP Rounding

Srikrishna Sridhar*, Victor Bittorf, Ji Liu, Ce Zhang, Christopher Ré (Stanford University), Stephen J. Wright

University of Wisconsin-Madison

srikris@cs.wisc.edu

objective

Solve large combinatorial problems

- Several problems in **machine learning, computer vision** and **data analysis** can be formulated using NP-hard combinatorial optimization problems.
- In many of these applications, approximate solutions for these NP-hard problems are 'good enough'.

We develop **novel theory** and **algorithms** to solve combinatorial problems **approximately**, in **parallel**. On three different problems, our solver can outperform Cplex (a commercial solver) by up to an **order of magnitude** in **runtime**, while achieving **comparable solution quality**.

background

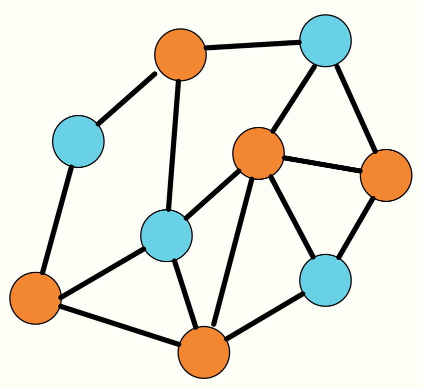
- LP rounding is a **4 step scheme** to approximate combinatorial problems with theoretical guarantees on solution quality.

Formulate Integer Program (IP)
↓
Relax IP to form a Linear Program (LP)
↓
Solve the LP ← difficult step!
↓
Round LP solution

example

The vertex cover problem

Find a set of vertices that cover the graph



Integer Program (IP)

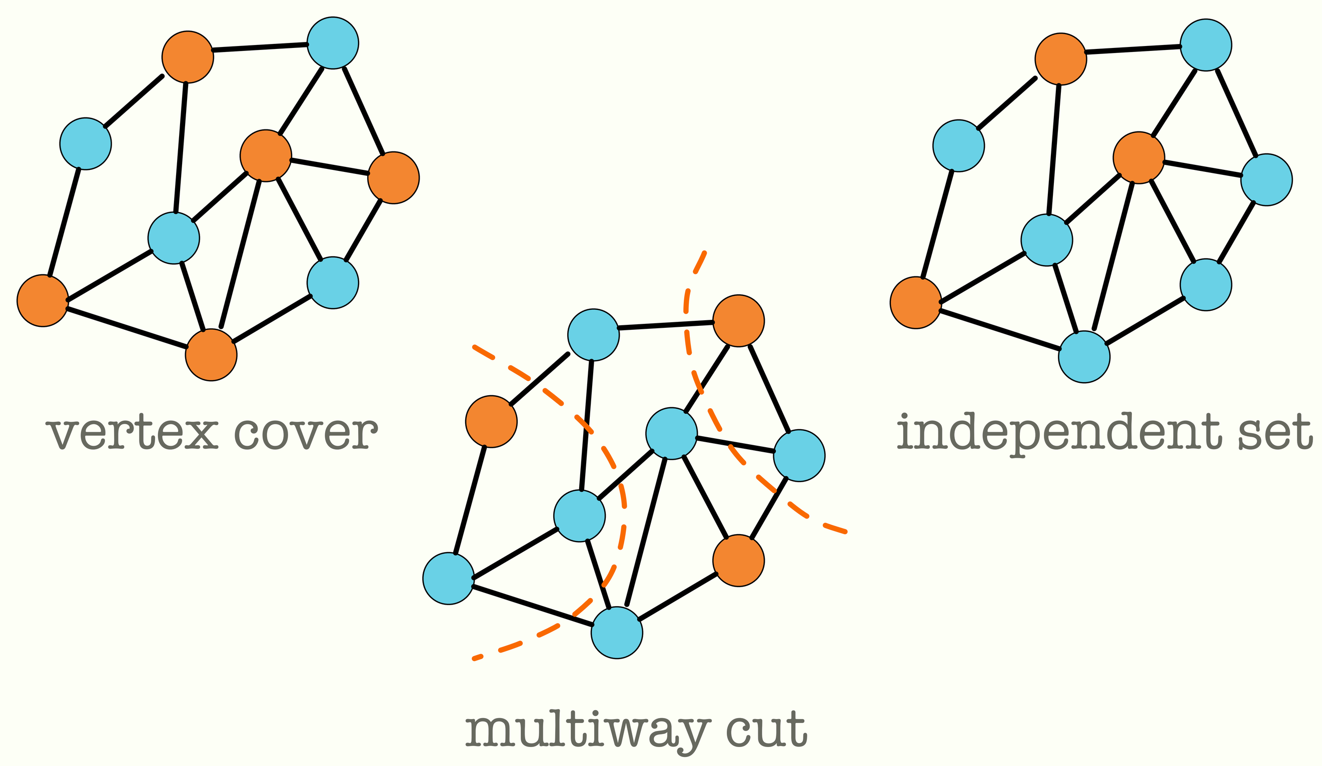
$$\min \sum_{v \in V} x_v \quad \text{s.t}$$
$$x_u + x_v \geq 1 \qquad \forall (u,v) \in E$$
$$x_v \in \{0,1\} \qquad \forall v \in V$$

Relaxed Linear Program (LP)

$$\min \sum_{v \in V} x_v \quad \text{s.t}$$
$$x_u + x_v \geq 1 \qquad \forall (u,v) \in E$$
$$x_v \in [0,1] \qquad \forall v \in V$$

Vertex cover has an approximation factor of 2

combinatorial problems



Problem Family	Approximation Factor	Applications
Set Covering	K or log(n)	Advertising, Classification, Tracking
Set Packing	eK + o(K)	MAP-Inference, Language processing
Multiway Cut	1.5 - 1/K	Entity resolution, Computer vision
Graphical Models	Heuristic	Clustering, Role labeling

main result

- **We can round approximate solutions to LPs (instead of exact solutions) without losing out on quality.**

$$\min c^T x \quad \text{s.t}$$
$$Ax = b, \; x \geq 0$$

Definition We define **x** to be an (ϵ, δ) approximate LP solution if

- Its objective is at most δ away from the optimal objective.
$$|c^T x - c^T x^*| \leq \delta c^T x^*$$
- It is at most ϵ away from feasibility
$$||Ax - b||_\infty \leq \epsilon$$

Exact LP solution → (Exact LP rounding) → α-factor approx

(ϵ, δ)-approx LP solution → (Approx LP rounding) → $\alpha(1 + \delta(1-\epsilon)^{-1})$ factor approx

Rounding approximate LP solutions are good enough!

novel theory

Approximate LP solutions

- We use a quadratic penalty (QP) formulation to solve the LP relaxation of the combinatorial problem.
- We solve the QP formulation with a **parallel** asynchronous stochastic co-ordinate decent method.
- We view the QP solutions as exact solutions of a perturbed LP, and use Renegar's analysis to quantify the difference with the exact LP solution
- We derive convergence rates that translate to bounds on worst case running time for the entire LP rounding scheme.

LP
$$\min c^T x \quad \text{s.t}$$
$$Ax = b, \; x \geq 0$$
↓
$$x(\beta) := \arg\min_{x \geq 0} \; c^T x - \bar{u}^T(Ax - b) + \frac{\beta}{2}\|Ax - b\|^2 + \frac{1}{2\beta}\|x - \bar{x}\|^2$$

Quadratic penalty formulation

novel implementation

We compared our solver with Cplex v.12.5 (a state-of-the-art commercial LP solver).

- Our solver was, on average, around **10x faster** than Cplex on **vertex cover** and **independent set**.
- Cplex was unable to solve any of the **multiway cut** problems in 3600 seconds.
- Our solver produced feasible integral solutions that were of comparable quality with Cplex.

maximization problems

Dataset	Vertex Cover			
	Cplex		Thetis	
	Solution	Time (s)	Solution	Time (s)
Amazon	2.04e+5	24.8	1.97e+5	2.97
DBLP	2.08e+5	22.3	2.06e+5	2.70
Google+	1.31e+5	40.1	1.27e+5	4.47

Dataset	Multiway Cut			
	Cplex		Thetis	
	Solution	Time (s)	Solution	Time (s)
Amazon	-	-	5	55.8
DBLP	-	-	5	63.8
Google+	-	-	5	109.9

minimization problems

Dataset	Independent Set			
	Cplex		Thetis	
	Solution	Time (s)	Solution	Time (s)
Amazon	1.56e+5	23.0	1.43e+5	3.09
DBLP	1.41e+5	23.2	1.34e+5	2.72
Google+	9.39e+4	44.5	8.67e+4	4.37