Subhash Khot: Unique Games Conjecture

6

Subhash Khot is one of the great young complexity theorists in the world. He is perhaps best known for his wonderful conjecture—the Unique Games Conjecture. But also he has worked in other areas, and solved many other important problems. He won the 2010 Alan T. Waterman Award. It is named after the first Director of the NSF, and was created in 1975. Subhash joins an array of famous mathematicians winning this award, including:

- Charles Fefferman, 1976;
- William Thurston, 1979;
- Harvey Friedman, 1984;
- Herbert Edelsbrunner 1991;
- Gang Tian, 1994;
- Emmanuel Candes, 2006; and
- Terence Tao, 2008.

This is a wonderful list.

We must visit Khot's Unique Games Conjecture, and survey some recent results on his conjecture.

John Cherniavsky, Senior Advisor for Research at NSF and a longtime friend of mine, told me that Subhash's talk at NSF for his prize was stellar. John added,

> "It might be appropriate to blog about his Unique Games Conjecture work—which seems very relevant to your blog."

I agree. So here is my view of the conjecture in three acts.

6.1 Act I: The Unique Games Conjecture

Khot's conjecture is remarkable on many levels. It is simple to state, yet it has created a wealth of important ideas and results. Perhaps stating a great conjecture—whether true or false—is one of the best ways to advance any field. Steve Cook and Dick Karp have the P $\neq$ NP question, Juris Hartmanis the Isomorphism Conjecture, Leslie Valiant the Permanent Versus Determinant Conjecture, and so on. Now we have the Unique Games Conjecture of Khot.

R.J. Lipton, K.W. Regan, *People, Problems, and Proofs*,
DOI 10.1007/978-3-642-41422-0_6,

Many of you probably know the Unique Games Conjecture, but I will give an informal definition of it anyway. I think of it as a generalization—a very clever one—of the simple problem of 2-coloring a graph. Suppose G is an undirected connected graph. There is a linear-time algorithm for testing whether the graph is 2-colorable—you probably saw it in Computing 101:

(1) Pick any starting vertex v.
(2) Color v **red**.
(3) Select any vertex u that is yet uncolored and is adjacent to a vertex that is colored. If u is adjacent to a **red** vertex, then color it **green**; if it is adjacent to a **green** vertex, then color it **red**.
(4) Continue until all the vertices are colored.
(5) The graph is 2-colorable if and only if there are no conflicts when all vertices are colored.

Khot's idea was to change the notion of 2-colorable in several simple ways. First, he allows a fixed but arbitrary set of colors—finite of course. This sounds like it might become the general coloring problem, but his brilliance is to pull back and make it closer to 2-coloring. He notes that the essence of 2-coloring is the rule:

> If a vertex v is **red**, then any neighbor is **green**.

Khot allows each edge to have its own rule, provided it obeys the following deterministic condition. If v and u are adjacent, then the color of one must **uniquely** determine the color of the other.

Generally a graph with rules restricting allowed values for the vertices on each edge is called a *constraint graph*—but the uniqueness makes these graphs special. It is still easy to tell whether or not such a graph can be colored so that all edge rules are followed. Just take the 2-coloring algorithm and modify it slightly:

(1) Pick any starting vertex v and a color c.
(2) Color v with the color c.
(3) Select any vertex u that is yet uncolored and is adjacent to a vertex v that is colored. Use the edge rule to color u. Note, there is **no** choice here once v is selected.
(4) Continue until all the vertices are colored.
(5) If all the edge rules are satisfied the graph is satisfiable. If not go back to step (1) and try another color c'. If all the colors have been tried, then the graph is not satisfiable.

This still takes only linear time, and still lies within Computing 101.

So far we have an easy problem—what is all the excitement? Why is this so important that Subhash was awarded the Waterman Prize? Khot adds one final ingredient: *approximation*. Suppose G is again a unique-constraint graph, but now I *promise* that one of two situations is true:

- there **exists** an assignment of colors to the vertices of G so at least $1 - \varepsilon$ of the edges are satisfied; or
- there is **no** assignment of colors to the vertices of G so more than ε edges are satisfied.

Telling which case is true is the nub of the Unique Games Conjecture (UGC). Khot conjectured it is NP-hard to tell which is true, for large enough sets of colors and all constraint graphs. Specifically, his conjecture states that for all $\varepsilon > 0$ there is an $R > 0$ such that if some polynomial-time algorithm distinguishes the above two situations for all unique-constraint graphs with R colors, then $\mathsf{P} = \mathsf{NP}$. The point is that allowing some edges to fail destroys the above linear-time algorithm. Indeed, tolerating εn failures could lead to $2^{\varepsilon n}$ amount of backtracking.

6.2 Act II: The Conjecture's Applications

The beauty of the problem is the power that goes with its simplicity. The ability to let each edge have its own rule allows many problems to be encoded into a UG problem. Since Khot made his conjecture in 2002 there have been many papers proving theorems of the form:

Theorem 6.1 *Obtaining a Y-approximation to problem X is as hard as solving the Unique Games problem.*

One of the best examples is the famous algorithm of Michel Goemans and David Williamson for maximum cut of a graph. If Khot's UGC is true, then their algorithm would be essentially the best one can hope for. That connection alone is a pretty neat result.

6.3 Act III: Is It True?

As you probably know I am unsure of the answer to $\mathsf{P} \neq \mathsf{NP}$, so I have similar thoughts about the Unique Games Conjecture. Perhaps my doubts are stronger since it is unknown whether or not solving unique-games instances is NP-hard. I do believe whether it is eventually shown to be false or not changes little about the brilliance of the conjecture. Mathematics is moved forward by great conjectures, and Khot has certainly done this for computational complexity. It takes insight, creativity of the highest order, and a bit of "guts" to make such a conjecture.

However, in recent years there have been a series of results showing that unique-games problems *may* not be so difficult. The ideas in these beautiful papers would not have existed without the conjecture, and the techniques used may be used to solve other problems—no matter what eventually happens to the conjecture.

I have neither the expertise nor time right now to give an exhaustive list of the results that have been chipping away at Khot's original hardness assertion. I would like to mention one direction in detail, and then just state the other more recent ones.

The first direction was a series of results by many to show that solving unique-games problems on expander graphs is easy. This is work of many, including Sanjeev Arora, Subhash Khot, Alexandra Kolla, David Steurer, Madhur Tulsiani, and Nisheeth Vishnoi. I apologize for not listing all, but a later expansion of the story may do justice to all.

A sample result is due to Konstantin and Yury Makarychev:

Theorem 6.2 *There exists a polynomial-time approximation algorithm that, given a $(1-\varepsilon)$-satisfiable instance of Unique Games on a d-expander graph G with $\varepsilon/\lambda_G \leq c_1$, finds a solution of cost*

$$1 - c_2 \frac{\varepsilon}{h_G},$$

where c_1 and c_2 are some positive absolute constants.

Note that h_G is the normalized edge expansion constant, while λ_G is the smallest positive eigenvalue of the normalized Laplacian of the graph G. Alexandra Kolla had similar results, but as I said earlier I cannot state and compare all the known results.

There is even more recent work due to Arora, Boaz Barak, and Steurer (ABS) that could be the start of the end of the UGC. They show, roughly, that any unique-games problem can be solved in time

$$2^{n^{\varepsilon}}$$

for any $\varepsilon > 0$. This does not rule out unique-games instances being NP-hard, but it certainly shows if the conjecture is true there would be great consequences.

The ABS paper shows that if the conjecture is true then either some famous NP-hard problems have subexponential time algorithms, or any reduction showing NP-hardness of some general set of unique-games instances must take time with a polynomial whose exponent depends on the ε parameter, which seems to defeat local-gadget reductions used for other approximation problems. Absolutely ABS shows that there are smarter ways of solving unique-games problems than simple backtracking—the result shows that a general divide-and-conquer paradigm can be employed. Their result contains a new insight into the structure of any graph, one that could have far-ranging consequences beyond the important application to the Unique Games Conjecture.

6.4 A Comment on Expanders

I believe the bounds for playing Unique Games on expanders are a bit misleading. They are right, they are deep, and they are important. But, unless I am confused—always a possibility—we need to be careful in reading too much practicality into these results.

This is my concern. Let G be a d-regular graph. Then, there is a trivial method to always find a solution to a unique-games problem with $1/(d+1)$ fraction of the edges satisfied. By Brooks' Theorem such a graph has a $d+1$ edge coloring: pick the most common color and these edges can be satisfied. Thus, if we are trying to separate a $1-\varepsilon$ from ε fraction of the edges, the promise problem is meaningful only for $\varepsilon > 1/(d+1)$.

Thus, if we are trying to separate a $1-\varepsilon$ from ε fraction of the edges, $\varepsilon > 1/(d+1)$. The theorem by Makarychev and Makarychev has a large constant $c_2 \approx 100$, and

$$1 - c_2\frac{\varepsilon}{h_G} > \varepsilon$$

forces the degree d to be very large. Unless I am wrong, if the graph is a spectral expander, then d must be order 10,000—and in any case it must be order 100.

These bounds are not extremely large by theory standards, but they do leave open the question of what happens on expander graphs of modest degree d. This is perhaps a small point, but one I find interesting.

6.5 Open Problems

What happens when we try to solve the UG problem on degree-3 graphs? What values of ε versus $1-\varepsilon$ are distinguishable? What happens on other graphs of low degree? The last word may be due to Oded Goldreich:

> I'm happy to see yet another application of the paradigm of decomposing any graph into parts that are rapidly mixing, while omitting relatively few edges. Let me seize this opportunity and mention the application of this paradigm in the bipartite tester for bounded-degree graphs.

He was referring to a paper by Jonathan Kelner and Aleksander Mądry on generating random spanning trees, but this could apply to the Arora–Barak–Steurer breakthrough. Perhaps graph property testing techniques can shed light on the Unique Games Conjecture.

6.6 Notes and Links

Original post:
http://rjlipton.wordpress.com/2010/05/05/unique-games-a-three-act-play/
Survey by Khot:
http://cs.nyu.edu/~khot/papers/UGCSurvey.pdf
Wikipedia reference on the conjecture:
http://en.wikipedia.org/wiki/Unique_games_conjecture
Semidefinite programming algorithm:
http://en.wikipedia.org/wiki/Semidefinite_programming#Examples
Paper by Makarychev and Makarychev:
http://eccc.hpi-web.de/eccc-reports/2009/TR09-021/Paper.pdf
Brooks' Theorem:
http://en.wikipedia.org/wiki/Brooks'_theorem
Goldreich source:
http://www.wisdom.weizmann.ac.il/~oded/MC/044.html