6.841/18.405J: Advanced Complexity Theory	May 8, 2002

Lecture 22

Lecturer: Madhu Sudan *Scribe: Emanuele Viola*

In this lecture we will talk about Quantum Complexity. However, let's first say some last things about Average Case Complexity.

1 Average Case Complexity

Average Case Complexity is vastly non understood. One of the main open problems is understanding the average complexity of $3-SAT$. For every n consider the following distribution on instances of $3-SAT$: pick a random formula on n variables with Δn clauses, where by a random formula we mean that every clause is chosen uniformly and independently at random among all possible clauses. It can be proven that:

- If $\Delta \geq 6$ then, with probability going exponentially (in n) to 1, the formula is not satisfiable.
- If $\Delta \leq 3$ then, with probability going exponentially (in n) to 1, the formula is satisfiable.

The threshold seems to be 4.2. In other words, $3-SAT$ with respect to the above distribution for $\Delta = 4.2$ seems to be a good candidate for a problem hard on average.

In practice, there are many heuristics that people use to find satisfying assignments that work well. The drawback is that when the heuristic algorithm fails, we do not know if it failed because there is no satisfying assignment or because it needs more time. Our hope would be an algorithm that provides us with a *proof* that the given formula is unsatisfiable, if that is the case. But do such proofs exist? This question is one of the main questions of *Proof Complexity*, and, modulo distribution on the instances, is the $co-NP$ vs NP question.

Our hope is relating NP-completeness and DNP-completeness. Some possibility of doing this have already been ruled out. In particular, consider a reduction from some NP-hard problem R' to a problem (R, D) (where D is a polynomial time samplable distribution) of the following kind: on input x we reduce the problem of deciding whether $x \in R'$ to the problem of solving (R, D) on, say, four instances x_1, x_2, x_3, x_4, where $\forall i$ x_i is distributed according to D, but they are *not* necessarily independent. [Feigenbaum & Fortnow] show that the existence of such a reduction implies that the PH collapses.

There are at least two ways out of this:

- Consider a 'classical' Turing reduction.
- Try a reduction from some other complexity class, such as *Statistical Zero Knowlegde*.

2 Quantum Complexity

For a physicist, physics is the goal, while computers are the tool. For a computer scientist is true the opposite: computers are the goal, physics is the tool. In particular, a computer scientist tries to come up with an abstract model of computation, and then show (1) it is physically realizable and (2) it is the strongest possible.

Very influencing has been the

- *Turing-Church Thesis*: every physically realizable computing device can be simulated by a Turing machine.

What is really relevant to Complexity Theory, however, is the

- *Strong Turing-Church Thesis*: every physically realizable *efficient* computing device can be simulated by a Turing machine *with a polynomial time slow-down*.

There have been two main challenges to the Strong Turing-Church Thesis.

The first one comes from *Randomness*: some problems are efficiently solvable (with high probability) using randomness, but are not known to be solvable in deterministic polynomial time. This is the BPP vs P (open) question.

The more recent challenge comes from *Quantum Physics*. We start with an experiment: Consider a wall with two small holes A and B. In front of the wall, at equal distance from A and B, we put a source of light, and behind the wall we put a screen where we can measure the intensity of the received light.

If we close one hole A or B, and let the light go through the other, we measure the same intensity on the screen. However, when opening both holes, the measured intensity is *not* the sum of the intensities previously measured. In particular, at the point on the screen at equal distance from A and B we observe *no light*, while there was some when opening just one hole.

This experiment can be turned in a computational problem as follows: given n walls each with n holes which can be open or closed, a source of light in front of all the walls and a point P on a screen behind all of them: do we measure light in P? This problem was considered by Feynman in the 80's.

2.1 Quantum bits

We define the *state* of a system consisting of one *Quantum Bit* (*qubit*) as a vector $\in \mathbb{C}^2$ of unit length (for this lecture, you can think of it as being a vector with real components). We can think of a qubit as of a coin which has been tossed but has not landed yet. The components of the state represent the probability that this coin will land on head or tails.

Similarly, the *state* of a system consisting of n qubits is a vector $\in \mathbb{C}^{2^n}$ of unit length. Again, we can think of each component of this vector as expressing the probability that the n coins will land in some particular configuration.

What can we *see* of the qubits, and how can we *manipulate* them? These two aspects are referred to as Quantum Measurement and Quantum Evolution, respectively.

2.2 Quantum Measurement

Consider two qubits in the state $(\frac{1}{\sqrt{2}}, \frac{1}{2}, -\frac{1}{2}, 0)$. We use the following notation (useful because states are often sparse, i.e. most of their coordinates are 0):

$$\frac{1}{\sqrt{2}}|00> \; + \frac{1}{2}|01> \; - \frac{1}{2}|10> \; + 0 \cdot |11> .$$

This means that if we measure the state then with probability $\left(\frac{1}{\sqrt{2}}\right)^2$ we will see 00, with probability $\left(\frac{1}{2}\right)^2$ we will see 01 and so on.

We can also measure one qubit at a time. In this case we follow the rule of *conditional probability*. For example, if we observe the first (leftmost) qubit in the above example, then with probability $1/2 + 1/4$ we see a 0, and then we are in the state

$$\sqrt{\frac{4}{3}} \cdot \frac{1}{\sqrt{2}}|00> \; + \sqrt{\frac{4}{3}} \cdot \frac{1}{2}|01>,$$

while if we see a 1 then we are in the state

$$1|10> .$$

2.3 Quantum Evolution

Consider some state on n qubits $v \in \mathbb{C}^{2^n}$. For every $2^n \times 2^n$ matrix U over $\mathbb{C}$ which is *unitary* (i.e. $U \cdot U^H = I_{2^n}$, where U^H is the transposed conjugate of U) the tranformation $v \to Uv$ is physically realizable.

For example, the matrix $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ negates a single qubit, the matrix $\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$ flips one (the value of the bit is preserved in the sign, so that we can unflip it), and the general form of these 1-qubit operations is $\begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$.

A *quantum circuit* is a circuit where we have such matrices as gates. It can be proven that we can postpone every measurement at the end paying only a little overhead. Rather than computing some function, quantum circuits *sample* from some distribution, and in particular they can sample from some distributions which are not known to be polynomial time samplable.

A *quantum Turing machine* is similar to a classical Turing machine, but the transition function is given by a unitary matrix, representing the quantum evolution of the state. [Bernstein & Vazirani] show that a quantum Turing machine can be initialized in such a way that it can simulate every given circuit. In particular, we can get ϵ close to the distribution sampled by the circuit with only polynomial time slow-down. Note we cannot ask for the quantum Turing machine to sample *exactly* the same distribution sampled by the circuit, because the circuit may have gates with values (e.g. $\sqrt{3}$) not included in the definition of the quantum Turing machine.