CSE491/596 Lecture Mon. 11/23: Randomized (Classical) Complexity and BPP

I

For any natural number m, $\mathbb{Z}_m$ stands for the integers modulo m. If m is a prime number p, then $\mathbb{Z}_p$ is a *field* (so that one can divide as well as multiply) and we write it as $\mathbb{F}_p$. The simplest such case is $p = 2$ which is $\{0, 1\}$ with the usual addition modulo 2 and multiplication. The field structure helps us prove the following result more easily.

Lemma: Suppose A, B, C are $n \times n$ matrices over $\mathbb{F}_p$ such that $AB \neq C$. Then

$$\Pr_{u \in \mathbb{F}_p^n}[ABu \neq Cu] \geq \frac{p-1}{p}.$$

Proof: Write $D = AB - C$. Note that we are not going to *calculate* D, because that would take the (standardly cubic) time for multiplying A and B that we are trying to avoid, but we are allowed to *argue based on its existence*. By linearity, $ABu \neq Cu \iff Du \neq 0$. So D has at least one row i with a nonzero entry, and its use may give a nonzero entry in the i-th place of the column vector $v = Du$. Note that

$$v_i = \sum_{j=1}^{n} D[i, j]u_j .$$

Let j_0 be a column in which row i has entry $c = D[i, j_0] \neq 0$. For any vector u, we can write

$$v_i = cu_{j_0} + a \quad \textit{where} \quad a = \sum\nolimits_{j \neq j_0} D[i, j]u_j .$$

The key observation is that because $\mathbb{F}_p$ is a field, for any $c \neq 0$, the values cu_{j_0} run through all p possible values as u_{j_0} runs through all p possibilities. Regardless of the value of a determined by the rest of row i and the rest of the vector u, the values $cu_{j_0} + a$ run through all p possibilities with equal probability. Hence the probability that $v_i \neq 0$ is exactly $\frac{p-1}{p}$. The probability of getting $v \neq 0$ (which could come from other nonzero entries too) is at least as great. ☒

The upshot is:

- If $AB = C$ then you will never be deceived: you will always get equal values from $A(Bu)$ and Cu and will correctly answer "yes, equal."
- If $AB \neq C$ and you try k vectors u at random, if you ever get $A(Bu) \neq Cu$ then you will know to answer "no, unequal" with 100% confidence.
- If you get equality each time, you will answer "yes, equal" but there is a $\frac{1}{p^k}$ chance of being wrong.

If you consider, say, a 1-in-n^3 chance of being wrong as minuscule, then you only need to pick k so that $p^k > n^3$, so $k = \frac{3}{\log p} \log n$ will suffice. Presuming p is fixed, this means $O(\log n)$ trials will suffice. The resulting $O(n^2 \log n) = \widetilde{O}(n^2)$ running time handily beats the time for multiplying AB out. Thus we trade off *sureness* for *time*.

For arithmetic modulo m not prime, or without any modulus, the analysis is messier---but not only is the essence the same, but the asymptotic order of k in terms of n and the confidence target $\epsilon(n)$ is much the same---it didn't really depend on p to begin with.

II

The matrix example makes the probability easiest to figure because it is *linear*, but it does not show a difference between "polynomial" and "exponential". This is enshrined in the definitions of the complexity classes BPP, RP, and co-RP. It is convenient to think of polynomial-time computable predicates $R(x, y)$ where y ranges over $\{0,1\}^{p(n)}$ with equal length rather than say $|y| \leq p(n)$ (with $n = |x|$ as usual). Then y is a sequence of $p(n)$ coin-flips.

Definition: A language A belongs to BPP if there is a polynomial p and a polynomial-time decidable predicate $R(x, y)$ such that for all n and x of length n:

$$x \in A \implies \Pr_{|y|=p(n)}[R(x,y)] > 3/4;$$
$$x \notin A \implies \Pr_{|y|=p(n)}[R(x,y)] < 1/4.$$

If the second probability is always 0 then A is in RP; if instead the first probability is always 0 then A is in co-RP; together these cases are called having *one-sided error*. Note that the first probability being always 1 is equivalent to saying it is always 0 for the complementary predicate $\widetilde{R}(x, y)$, which is where RP and co-RP start to get confusing. The same ability to flip between R and its negation tells right away that BPP is closed under complements, which makes it less confusing. For BPP, we can also combine the conditions into one, namely

$$\Pr_{|y|=p(n)}[A(x) = R(x,y)] > 3/4.$$

But this is often less helpful than having the two separate probabilities. Note that if the second probability is 0 then $R(x, y)$ is impossible when $x \notin A$. It follows that having $R(x, y)$ be true makes y a valid *witness* for $x \in A$, so we have proved the following:

Proposition: RP $\subseteq$ NP and co-RP $\subseteq$ co-NP. $\boxtimes$

Of course $L \in \text{RP} \iff \widetilde{L} \in \text{co-RP}$, so whether a problem belongs to RP or to co-RP depends on which side one takes as the "yes" side. If you regard $AB = C$ as the yes side and $ABu = Cu$ as the verifying predicate "$R(\langle A, B, C\rangle, u)$", then the matrix example has one-sided error of the "co-RP type",

meaning that if the answer is yes then you can never be bluffed into thinking the answer is no; but in a true-negative case there is a tiny chance of getting a false positive (i.e., thinking $AB = C$ because every u that you tried gave $A(Bu) = Cu$). You could say that the language

$$L = \{\langle A, B, C\rangle : AB = C\} \text{ belongs to co-RPTIME}\left[\widetilde{O}(n^2)\right],$$

but this notation gets ugly and hides the dependence between the error probability and the time allowed for multiple trials. For polynomial bounds it is even more favorable than for "Oh-tilde" type bounds:

Amplification Lemma: If $A \in$ BPP with associated $R(x, y)$ and $p(n)$, then for any polynomial $q(n)$ we can build a polynomial-time decidable $R'(x, z)$ and associated polynomial $p'(n)$ such that for all x,

$$x \in A \implies \Pr_{|z|=p'(n)}[R'(x, z)] > 1 - 2^{-q(n)};$$
$$x \notin A \implies \Pr_{|z|=p'(n)}[R'(x, z)] < 2^{-q(n)}.$$

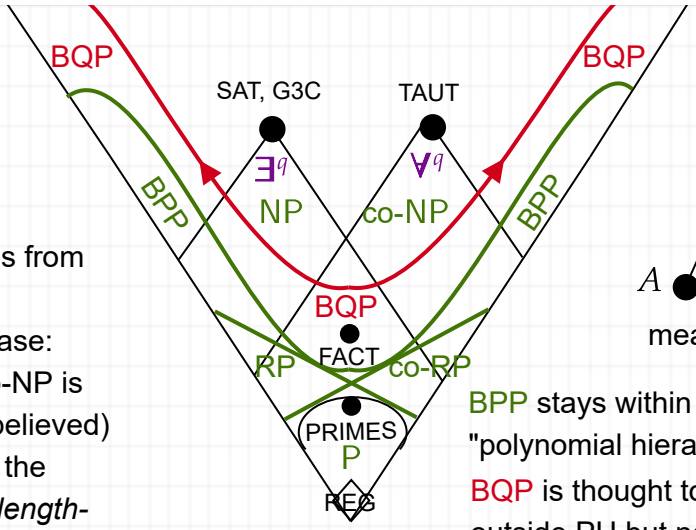
Moreover, we can achieve this even if the original R and p only give a "non-negligible" advantage, meaning that for some polynomial $r(n) \geq n$,

$$x \in A \implies \Pr_{|y|=p(n)}[R(x, y)] > \frac{1}{2} + \frac{1}{r(n)};$$
$$x \notin A \implies \Pr_{|y|=p(n)}[R(x, y)] < \frac{1}{2} - \frac{1}{r(n)}.$$

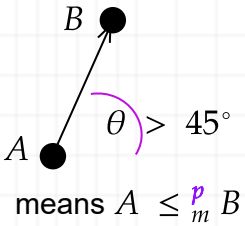
Proof Sketch: Regard $z = \langle y_1, y_2, \ldots, y_{q'(n)}\rangle$ where $q'(n) = O(q(n))$ and define $R'(x, z)$ to be the majority vote of the polynomially-many trials $R(x, y_j)$. ☒

There is a similar amplification lemma for one-sided error; in fact, the details of getting the exponentially small error are simpler because you don't need majority vote. A philosophical point is that the the theoretical software error can be reduced below the chance of hardware error---but when you see something like https://www.wnycstudios.org/podcasts/radiolab/articles/bit-flip (which I heard on NPR two weeks ago), maybe that's not so reassuring...

The definition of the quantum complexity class BQP is similar, except that in place of getting y such that $R(x, y)$ by rolling classical dice, we have a *quantum circuit* C in place of R and get the effect of y by measurements. Amplification and many other properties hold similarly; the main external difference is that the factoring problem and some others belong to BQP but (hopefully!) not to BPP. Added: The "landscape" of current knowledge is:



Note differences from the unbounded computability case: NP intersect co-NP is not known (or believed) to equal P, and the quantifiers are *length-bounded* by a polynomial.



BPP stays within the second level of the "polynomial hierarchy" (PH).
BQP is thought to hug the walls even outside PH but never have NP-hard sets.

III

Before 2002, the usual first example of a language in BPP was the language of prime numbers, which was long known to belong to $\text{ZPP} = \text{RP} \cap \text{co-RP}$. That is, before it was **derandomized** by being shown to belong to P. The deterministic algorithm runs with a higher polynomial exponent than the randomized ones, however, so many software primality tests are still randomized. Except for the following bellwether problem, it is hard to find other examples, and even harder to find ones that are only known to have two-sided error. Well, here is the big example (which Debray's notes call "Zero-Poly" and define only for $\mathbb{Z}$ not fields $\mathbb{F}$, but this is the standard name in all cases):

Polynomial Identity Testing (PIT).
Instance: A polynomial formula $f(x_1, \ldots, x_n)$ over $\mathbb{Z}$, $\mathbb{Z}_m$, or a field $\mathbb{F}$ (see notes on degree below).
Question: Does f when "multiplied out" cancel to the all-zero polynomial?

Multiplying out is not so simple---it can take exponential time. Consider how in reducing from **Exactly One 3SAT** to **Binary Linear Equations** (presentation topic 4 of HW6) we get equations $E_1 = 1, E_2 = 1, \ldots, E_m = 1$ from the m clauses. Each equation has 3 variables plus maybe a constant term. We can multiply them together to get a single equation of degree (only!) m:

$$(E_1)(E_2) \cdots (E_m) - 1 = 0.$$

Multiplying this out, however, gives somewhere between 3^m and 4^m terms. Even so, we're hung up on the "NP-side" of looking for one solution, rather than the "co-NP side" of seeing whether all assignments are solutions. This can be done, but you still have to mix in the non-linear equations $x_i^2 - x_i = 0$ to force each variable to be 0 or 1, and even then, the resulting polynomial might not cancel entirely symbolically, as we show with a simple one-variable example next.

A "yes" answer certainly implies that $f(\vec{a}) = 0$ for all arguments $\vec{a} = (a_1, \ldots, a_n) \in \mathbb{F}^n$. Hence if we find an argument $\vec{a} = (a_1, \ldots, a_n)$ such that $f(a_1, \ldots, a_n) \neq 0$, then we know the answer is "no".

Now observe the following examples/facts:

1. There are polynomials that are zero on all $\vec{a} \in \mathbb{F}^n$ without multiplying out to zero; a simple one-variable example ($n = 1$) with $p = 2$ is $g(x_1) = x_1^2 - x_1$ over $\mathbb{F}_2$.
2. However, if we enlarge the field to $\mathbb{F}' = \mathbb{F}_4$ or $\mathbb{F}_8$ (etc.) while keeping the mod-2 **characteristic** the same (note those are not the same as the integers mod 4 or mod 8), then $g(x_1)$ is no longer everywhere-zero over $\mathbb{F}'$.
3. Whereas, if $f(x_1, \ldots, x_n)$ multiplies out to zero over $\mathbb{F}$, then it multiplies out to zero over any $\mathbb{F}' \supseteq \mathbb{F}$ of the same characteristic (called an *extension field*), and vice-versa.

Points 1 and 2 are why the fact of **PIT** being in BPP---indeed, with one-sided error like in the $AB = C$ matrix example---does not put **SAT** into BPP. (While composing these notes, I thought of a possible allusion to how working with binary truth values involves the "law of excluded middle" while going to $\mathbb{F}' = \mathbb{F}_4$ or $\mathbb{F}_8$ (etc.) means doing without it---but I am not sure how meaningful it is.)

An important further point is that we can exponentiate the field size with only polynomial work: For any $k > 1$, $\mathbb{F}_{2^k}$ equals the binary vector space $\mathbb{F}_2^k$ augmented with an extra multiplication operation $u*v$ on binary k-tuples. Computing $u*v$ only involves multiplying and dividing by certain single-variable polynomials of degree k modulo 2. With n variables you wind up with (nk)-tuples but the arithmetic involves only $\widetilde{O}(nk)$ work per operation.

The upshot of this is that in stating PIT, we may suppose that the total degree d of the polynomial formula $f(x_1, \ldots, x_n)$ obeys $d < |\mathbb{F}|$. If it doesn't, then we can scale up $\mathbb{F}$ to $\mathbb{F}'$ to make it so---unless the degrees d_n of the formulas f_n for each n are horribly exponential. This allows us to apply the following "strong form" of the **Schwartz-Zippel-(de Millo-Lipton) Lemma**.

Lemma: Take any finite subset S of the field $\mathbb{F}$ (if $\mathbb{F}$ is already finite we can just take $S = \mathbb{F}$). Let $f(x_1, \ldots, x_n)$ have total degree at most d. Suppose f does not multiply out to 0. Then

$$\Pr_{a_1,\ldots,a_n \in S}[f(a_1, \ldots, a_n) = 0] \leq \frac{d}{|S|}.$$

There is an alternative weaker form in which d' is the maximum degree in any one of the n variables and the probability conclusion you get is $\leq \frac{d'n}{|S|}$. The weaker form also holds over $\mathbb{Z}$ and $\mathbb{Z}_m$ and other *rings* that are not *fields*. Note that the average degree of a variable is $\frac{d}{n}$ so the numerator $d'n$ is similar to just d, but because this defines d' to be the max, not the average, the result you get is technically weaker (but just as useful in most cases---this is how Debray gives it). Note that I partner with Lipton; I helped him explain at

https://rjlipton.wordpress.com/2009/11/30/the-curious-history-of-the-schwartz-zippel-lemma/

the story of how he and Rich de Millo originally had the weaker form in 1977, a year ahead of Jack

Schwartz's stronger form with Richard Zippel in-between. Moreover, I shared an office with Zippel at Cornell for some months in 1986 (if I recall correctly).

Corollary: **PIT** (over any of $\mathbb{Z}$, $\mathbb{Z}_m$, or fields $\mathbb{F}$, even infinite fields) belongs to co-RP.

The basic fact underlying the proof is that a *single*-variable polynomial of degree d has at most d roots. The fact of having n variables expands in the denominator and the numerator in a similar manner; formally, this is shown by induction on n. For those interested in the details,

https://nickhar.wordpress.com/2012/02/01/lecture-9-polynomial-identity-testing-by-the-schwartz-zippel-lemma/

also has a nice comparison of **PIT** with the evaluates-to-zero problem.