Query Complexity and Symmetries

Vincent Nesme

LIP, ENS Lyon

LRI, Université Paris-Sud

11th May 2007

Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

What is a Problem?

Problem: **instance** $\longmapsto$ **solution**

Example

- list of a fixed size N containing only 0's
  $[0; 0; \dots; 0] \longmapsto$"TRUE";
- list of size N containing 0's everywhere expect for one 1
  $[0; \dots; 0; 1; 0; \dots; 0] \longmapsto$"FALSE".

This is the **search in an unordered list of size** N.

A **solver** must attempt to find the solution to every instance it is shown, at least with a good probability.

Complexity

The complexity of a problem is the **minimum amount of resources** that the solver has to use. These resources may take such form as. . .

- computing time;
- memory;
- energy/money/. . .

Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

Query Complexity

The solver has **illimited computing ressources**.

However, it has a **limited knowledge of the instance**. It must spend resources to know it.

Probabilities may be involved. We then worry about **worst-case** error. When we allow an error ε, it means that for every instance, the solver must find the correct result with probability **at least** $1 - \varepsilon$.

What is a Problem?

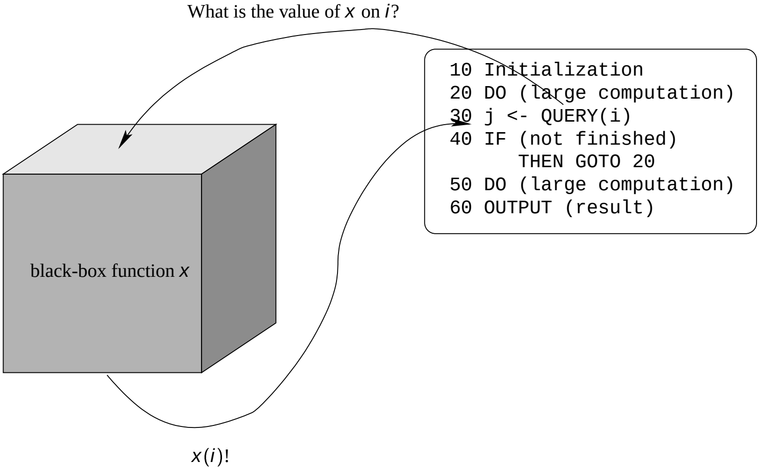
Problems are formalized as partial functions from J^I to R.

- An instance is a **function** from I to J
- Some functions from I to J may not be instances. In that event, the problem is said to be a **promise problem**.
- The solution to the instance $x \in J^I$ is a result in R.
- I is the set of possible queries, and J the set of possible queries. Questions like “What is $x(i)$?” are the only way the solver may know of x.

Example

A list $[a_1; a_2; \ldots; a_n]$ is formalized as the fonction $\begin{pmatrix} \{1,\ldots,n\} & \rightarrow & A \\ i & \mapsto & a_i \end{pmatrix}$, with R being $\{“TRUE”, “FALSE”\}$.

Dialogue between the Solver and the Black Box



What is the value of x on i?

black-box function x

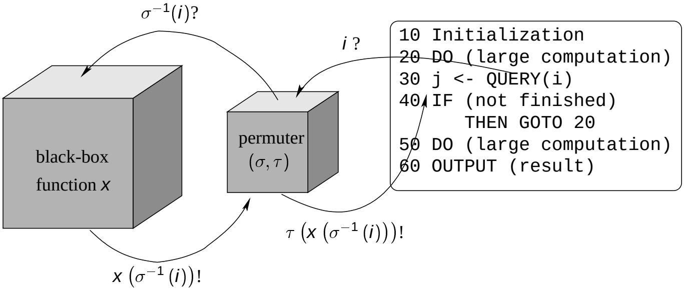
```
10 Initialization
20 DO (large computation)
30 j <- QUERY(i)
40 IF (not finished)
       THEN GOTO 20
50 DO (large computation)
60 OUTPUT (result)
```

$x(i)$!

Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

A problem has **symmetries** when some permutations may be applied on I (the queries) and J (the answers), that do not change the outcome.


$\sigma^{-1}(i)$?
i ?
black-box
function x
permuter
(σ, τ)
$\tau\left(x\left(\sigma^{-1}(i)\right)\right)$!
$x\left(\sigma^{-1}(i)\right)$!

```
10 Initialization
20 DO (large computation)
30 j <- QUERY(i)
40 IF (not finished)
      THEN GOTO 20
50 DO (large computation)
60 OUTPUT (result)
```

Example

All the permutations of the queries are symmetries of the search in an unordered list of size N.

Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

Fundamental Duality

Randomized algorithms are difficult to analyze!

To help us, tools exists, that transpose the difficulty to studying the behaviour of **deterministic** algorithms on a **probabilistic superposition** of instances.

We are to simplify that even more, by resorting to nonadaptivity. The goal is to have **exact** and **“simple”** (purely combinatorial) expressions of the complexity.

A “Simple” Formula

Proposition 4.12

$$\min\left\{ T \middle/ \left(\min_{\substack{p_r \geq 0 \\ \sum_{r\in R} p_r = 1}} \max_{\substack{A \in I^T \\ \bigsqcup_{r\in R} X_r = J^T}} \sum_{r\in R} p_r \cdot \mathbb{P}_A^r(X_r) \right) \geq 1 - \varepsilon \right\}$$

What is nonadaptivity?

Definition

The solver is said to be **nonadaptive** when the queries are **independent** of the previous answers.

Equivalently, it decides in advance which questions it is going to ask the black box.

Symmetries and Nonadaptivity

Nonadaptivity $\equiv$ Adaptivity? difficult question!

Fortunately, on some symmetry conditions, this is true. For instance:

Facts 4.21 and 4.23

- *problems where each permutation of the queries is a symmetry;*
- **collision** *problems where every permutation of the answers are symmetries.*

A collision problem is a decision problem where positive instances are one-to-one, whereas none of the negative instances is.

Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

Some Query Complexities

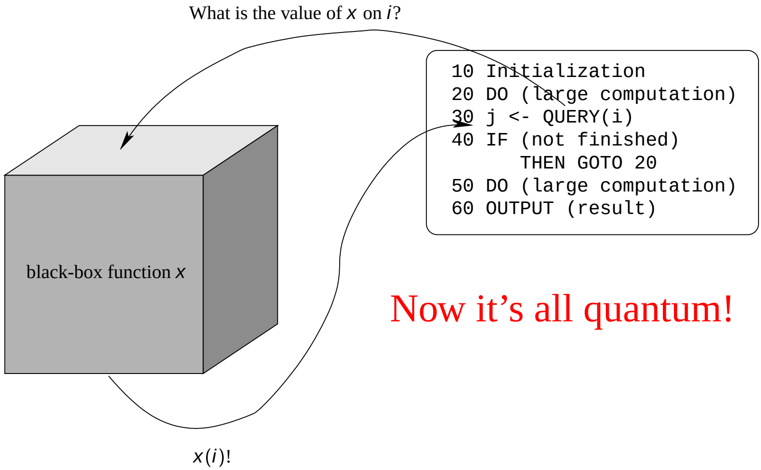
Symmetries thus allow us to compute exact query complexities for some classes of problems. Here are some examples.
The maximal error probability ε is fixed in $\left[0; \frac{1}{2}\right]$.

Problem	**Complexity**
Search in an unordered list of size N	$\left\lceil \frac{1-2\varepsilon}{1-\varepsilon} N \right\rceil$
Hidden translation for functions from $\{0,1\} \times \{1,\ldots,N\}$ to $\{1,\ldots,2N\}$	$\left\lceil 2\sqrt{\frac{1-2\varepsilon}{1-\varepsilon} N} \right\rceil$
One-to-one versus two-to-one for functions from $\{1,\ldots,2N\}$ to itself	$\sim 2\sqrt{N \ln\left(\frac{1}{\varepsilon} - 1\right)}$

Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

Quantum Dialogue between the Solver and the Black Box



```
10 Initialization
20 DO (large computation)
30 j <- QUERY(i)
40 IF (not finished)
      THEN GOTO 20
50 DO (large computation)
60 OUTPUT (result)
```

Now it’s all quantum!

Quantum Mechanics: The Shortest Introduction Ever

Let the S_i's be physical “sure” states.

Classical: $\sum_i p_i \cdot S_i$, with $p_i \geq 0$. The probability of measuring S_i is p_i. **Stochastic** matrices.

Quantum: $\sum_i a_i \cdot S_i$, with $a_i \in \mathbb{C}$. The probability of measuring S_i is $|a_i|^2$. **Unitary** matrices.

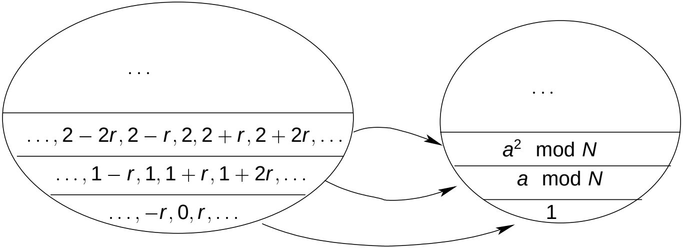
Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

Order Finding

We want to compute the multiplicative order r of an element a in $\mathbb{Z}/N\mathbb{Z}$. N is not known to the solver *a priori*. We may only query this function:

$$\left(\begin{array}{ccc} \mathbb{Z} & \to & \mathbb{Z} \\ i & \mapsto & a^i \mod N \end{array} \right)$$



$\dots$
$\dots, 2-2r, 2-r, 2, 2+r, 2+2r, \dots$
$\dots, 1-r, 1, 1+r, 1+2r, \dots$
$\dots, -r, 0, r, \dots$
$\dots$
$a^2 \mod N$
$a \mod N$
1

Hidden Subgroup Problem

Definition

Group G, set X.
$\gamma : G \to X$ is a function s.t. there exists $H \leq G$ s.t.

$$\gamma(g) = \gamma(g') \iff \exists h \in H \quad g = g' \cdot h$$

$$\begin{tikzcd} G \arrow[rr, "\gamma"] \arrow[dr] & & X \\ & G/H \arrow[ur, hook, dashed, "\tilde{\gamma}"'] & \end{tikzcd}$$

Given such a γ, H is uniquely determined: it is the subgroup **hidden** by γ.

G and X are known to the solver, which must find H.

Why the HSP?

General aim: study the **power** of quantum relatively to classical.

HSP lie at the boundary.

While most probably intractable by classical means, some of them (notably **Abelian** ones) are theoretically solvable in short time with quantum computing.

HSP include both factorizing (which is classically hard, but quantumly easy) and Graph Isomorphism (which seems hard for both, though not NP-hard).

Theorem (KNP 05)

The query complexity of the Abelian HSP in G is in $\Theta(\mathrm{rank}(G))$*.*

Corollary

The query complexity of the HSP in G is
$\Omega\left(\max_{H \leq G, H \text{ abelian}} \mathrm{rank}(H)\right)$*.*

This remains true if. . .

- we deal with the associated decision problem;
- if we only care about average error for subgroups of a fixed size.

Proof of the upper bound

The **standard algorithm** is due to Simon (94), Shor (97), Ettinger, Høyer and Knill (04) and al.

It is known to require $\mathcal{O}(\log |G|)$ queries for the decision problem.

Actually, $\mathcal{O}(\text{rank}(G))$ queries are sufficient when G is Abelian, whether considering the decision problem or not.

Proof of the Lower Bound

Sketch of the proof:

- Consider only groups of the form $(\mathbb{Z}/p\mathbb{Z})^n$.
- Use polynomial method.
- Symmetrize to get a univariate polynomial.
- Finish it with an appropriate approximation lemma.

Outline

1. Models and Definitions
   - Problems and Complexity
   - Queries and Black Box
   - Symmetries
2. Classical Randomized
   - Duality and Nonadaptivity
   - Explicit Calculations
3. Quantum
   - Model
   - Hidden Subgroup Problems
   - Polynomial Method

Definitions

Problem $f : J^I \to R$. For $x \in J^I$, $i \in I$ and $j \in J$, we define

$$\Delta_{i,j}(x) = \begin{cases} 1 & \text{if } x(i) = j \\ 0 & \text{otherwise} \end{cases}$$

Theorem (BBCMW 98)

If a quantum algorithm makes T queries, then for every $r \in R$, the probability that it outputs r on instance x is a polynomial in the $\Delta_{i,j}$'s of degree at most $2T$.

Problem: This polynomial is ultrasupermultivariate.

Symmetrization

Following the symmetries of the HSP on $(\mathbb{Z}/p\mathbb{Z})^n$, coming from the automorphisms of the group, we average this polynomial. Namely, we consider $P(k)$, the probability that the algorithm outputs “H is trivial” when H is actually a subgroup of order k.

Thanks to the particular structure of $(\mathbb{Z}/p\mathbb{Z})^n$, P **still** is a polynomial in k, of degree at most $2T$. Moreover, it has interesting **constraints**:

- $P(1) \simeq 1$ and
- $P(p^i) \simeq 0$ for $i \in \{1, \dots, n\}$.

Approximation Lemma

Lemma (KNP 05)

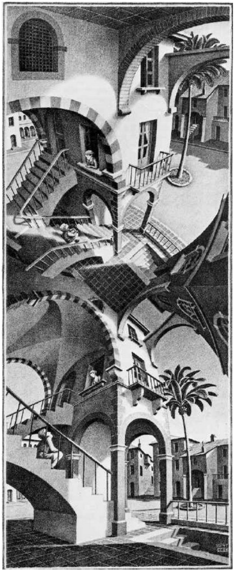
Let $c > 0$ and $\xi > 1$ be constants, and P a real polynomial s.t.

- *for every $i \in \{0, \ldots, n-1\}$, $\left|P(\xi^i)\right| \leq 1$, and*
- *there exists $x_0 \in [1; \xi]$ s.t. $|P'(x_0)| \geq c$.*

Then $\deg(P) = \Omega(n)$.

Conclusion: since P is of degree at most $2T$ but at least $\Omega(n)$, then T must be at least $\Omega(n)$.

Questions, Prospects and Perspective



- Nonadaptive quantum query complexity
- . . . or any kind of lower bound for non-abelian HSP, for that matter
- Method for proving quantum lower bounds for symmetric problems