A quantum information trade-off for Augmented Index

Rahul Jain  (Singapore)
and
Ashwin Nayak  (Waterloo)
Privacy in communication

Is $x > y$?
Privacy in communication

Two millionaires problem [Yao ’82]

Determine if $x > y$ without revealing any other information about their wealth.
Privacy in communication

Two millionaires problem [Yao ’82]

Determine if $x > y$ without revealing any other information about their wealth

Impossible without restriction on their computational power
How much information is revealed?
How much information is revealed?

- Similar to honest but curious model

  Follow the protocol, but use messages to gain information
How much information is revealed?

• **Similar to honest but curious model**
  
  Follow the protocol, but use messages to gain information

• **Extremes**

  Alice reveals all of $x$, Bob reveals only $f(x,y)$, and vice-versa
How much information is revealed?

• Similar to honest but curious model
  
  Follow the protocol, but use messages to gain information

• Extremes
  
  Alice reveals all of $x$, Bob reveals only $f(x,y)$, and vice-versa

• Better protocols are possible
  
  Equality: $O(\log n)$ one-way protocol, $1/poly(n)$ error, reveals only $O(1)$ bits about one input \[GV’10, FHS’10\]
Augmented Index

Variant of Index function

Bob has the prefix $x[l, k-1]$, and a guess $b$ for the value of $x_k$. 

$x = x_1 x_2 \ldots x_n$

Is $x_k = b$?

$k$, $x[l, k-1]$, $b$
Index function

Fundamental problem with a rich history

- communication complexity [KN’97]
- data structures [MNSW’98]
- private information retrieval [CKGS’98]
- learnability of states [KNR’95, A’07]
- finite automata [ANTV’99]
- formula size [K’07]
- locally decodable codes [KdW’03]
- sketching e.g., [BJKK’04]
- information causality [PPKSWZ’09]
- non-locality and uncertainty principle [OW’10]
- quantum ignorance [VW’11]
Results
Results

Theorem \[ \text{[JN'11]} \]

If a quantum protocol computes $\text{Ai}_n$ with probability $1 - \epsilon$ on the uniform distribution, either

Alice reveals $\Omega(n/t)$ information about $x$, or

Bob reveals $\Omega(1/t)$ information about $k$,

even when restricted to 0-inputs, where $t$ is the number of messages.
Results

**Theorem [JN’11]**

If a quantum protocol computes \( A_l^n \) with probability \( 1 - \epsilon \) on the uniform distribution, either

Alice reveals \( \Omega(n/t) \) information about \( x \), or

Bob reveals \( \Omega(1/t) \) information about \( k \),

even when restricted to 0-inputs, where \( t \) is the number of messages.

**Stronger theorem for classical protocols [JN’10]**

Alice reveals \( \Omega(n) \), or Bob reveals \( \Omega(1) \) information.
Related work
Related work

Privacy in communication (quantum)

- Klauck’04: w.r.t. hard distribution
- Index function: various flavours [JRS’02, ’09; KdW’04; LeG’11]
- Jain, Radhakrishnan, Sen’03: AND(a, b), w.r.t. superposition over 0-inputs
Related work

Privacy in communication (quantum)

- **Klauck’04**: w.r.t. hard distribution
- **Index function**: various flavours [JRS’02, ’09; KdW’04; LeG’11]
- **Jain, Radhakrishnan, Sen’03**: AND($a$, $b$), w.r.t. superposition over 0-inputs

Augmented Index (classical)

- **Magniez, Mathieu, N.’10**: In Alice-Bob-Alice classical protocols, Alice reveals $\Omega(n)$, or Bob reveals $\Omega(\log n)$ bits of information, even when restricted to 0-inputs.
- **Chakrabarti, Cormode, Kondapalli, McGregor’10**: independent and concurrent work, similar classical results as ours.
- Neither technique applies to quantum communication.
Why Augmented Index?
Why privacy w.r.t. 0-inputs?
Why Augmented Index?
Why privacy w.r.t. 0-inputs?

...010110010101011110010...

device with small memory
Why Augmented Index? Why privacy w.r.t. 0-inputs?

Streaming model

- massive input, cannot be stored entirely in memory
- input arrives sequentially, read one symbol at a time
- device processes each symbol quickly, while maintaining small workspace

Device with small memory
Why Augmented Index? Why privacy w.r.t. 0-inputs?

Streaming model

- massive input, cannot be stored entirely in memory
- input arrives sequentially, read one symbol at a time
- device processes each symbol quickly, while maintaining small workspace

Attractive model for quantum computation
Streaming quantum algorithms
Streaming quantum algorithms

Advantage over classical

- Quantum finite automata: streaming algorithms with constant memory and time per symbol. E.g., may be exponentially smaller than classical FA.

- Use exponentially smaller amount of memory for certain problems [LeG’06, GKKRdW’06]
Streaming quantum algorithms

Advantage over classical

- Quantum finite automata: streaming algorithms with constant memory and time per symbol. E.g., may be exponentially smaller than classical FA.

- Use exponentially smaller amount of memory for certain problems [LeG’06, GKKRdW’06]

Advantage for natural problems?

- For context-free languages: e.g., checking whether a sentence is grammatical.

- For Dyck(2), checking if an expression in two types of parentheses is well-formed? Canonical CFL, used in practice.
Streaming algorithms for Dyck(2)
Streaming algorithms for Dyck(2)

Magniez, Mathieu, N.’10:

- A single pass randomized algorithm that uses $O((n \log n)^{1/2})$ space, $O(\text{polylog } n)$ time/symbol
- 2-pass algorithm, uses $O(\log^2 n)$ space, $O(\text{polylog } n)$ time/symbol, second pass in reverse
- Space usage of 1 pass algorithm is optimal, via study of information revealed in classical protocols for Augmented Index.
Streaming algorithms for Dyck(2)

Magniez, Mathieu, N.’10:

- A single pass randomized algorithm that uses \( O( (n \log n)^{1/2} ) \) space, \( O(\text{polylog } n) \) time/symbol
- 2-pass algorithm, uses \( O(\log^2 n) \) space, \( O(\text{polylog } n) \) time/symbol, second pass in reverse
- Space usage of 1 pass algorithm is optimal, via study of information revealed in classical protocols for Augmented Index.

Better quantum algorithms?

- Classical version shows limitations of multiple (unidirectional) passes over input.
- The information cost trade-off would give a similar negative answer, provided a conjectured information inequality holds.
The information cost trade-off

If a quantum protocol computes $A_l^n$ with probability $1 - \epsilon$ on the uniform distribution, either

- Alice reveals $\Omega(n/t)$ information about $x$, or
- Bob reveals $\Omega(1/t)$ information about $k$,

even when restricted to 0-inputs, where $t$ is the number of messages.
Intuition behind proof
(2 messages, no private workspace)

\[ x = x_1 x_2 \ldots x_n \]

\[ M_A \]

\[ M_B \]

output

\[ k, x[1,k-1], b \]
Intuition behind proof
(2 messages, no private workspace)

Consider uniformly random $X$, $K$, let $B = X_K$. 

$x = x_1 \ x_2 \ldots \ x_n$

$\downarrow$

output

$M_A$

$M_B$

$k, \ x[1, k-1], \ b$
Consider uniformly random $X$, $K$, let $B = X_K$.

- Consider $K$ in $[n/2]$. If $M_A$ has $o(n)$ information about $X$, then it is nearly independent of $X_L$, $L > n/2$. Flipping Alice’s $L$-th bit does not perturb $M_A$ much.
Consider uniformly random \( X, K \), let \( B = X_K \).

- Consider \( K \) in \([n/2]\). If \( M_A \) has \( o(n) \) information about \( X \), then it is nearly independent of \( X_L, L > n/2 \). Flipping Alice’s \( L \)-th bit does not perturb \( M_A \) much.

- If \( M_B \) has \( o(1) \) information about \( K \), then \( M_B \) is nearly the same for most pairs \( j \leq n/2, L > n/2 \). Switching Bob’s index from \( j \) to \( L \) does not perturb \( M_B \) much.
Consider uniformly random $X$, $K$, let $B = X_K$.

- Consider $K$ in $[n/2]$. If $M_A$ has $o(n)$ information about $X$, then it is nearly independent of $X_L$, $L > n/2$. Flipping Alice’s $L$-th bit does not perturb $M_A$ much.

- If $M_B$ has $o(1)$ information about $K$, then $M_B$ is nearly the same for most pairs $J \leq n/2$, $L > n/2$. Switching Bob’s index from $J$ to $L$ does not perturb $M_B$ much.

Consequences of Average Encoding Theorem \cite{KNTZ07, JRS03}
Intuition continued...
Intuition continued...

<table>
<thead>
<tr>
<th>Alice’s input</th>
<th>Bob’s input</th>
<th>Protocol state</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$X[I, K]$</td>
<td>$</td>
</tr>
<tr>
<td>1</td>
<td>$X[I, K]$</td>
<td>$</td>
</tr>
</tbody>
</table>

flip $L$-th bit

same index
Intuition continued...

<table>
<thead>
<tr>
<th>Alice’s input</th>
<th>Bob’s input</th>
<th>Protocol state</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>0</code></td>
<td><code>X[I, K]</code></td>
<td>`</td>
</tr>
<tr>
<td>flip L-th bit</td>
<td><code>X[I, K]</code></td>
<td>`</td>
</tr>
<tr>
<td><code>1</code></td>
<td><code>X[I, K]</code></td>
<td>`</td>
</tr>
<tr>
<td>same L-th bit</td>
<td><code>X[I, L]</code></td>
<td>`</td>
</tr>
</tbody>
</table>

0-input
Intuition continued...

<table>
<thead>
<tr>
<th>Alice’s input</th>
<th>Bob’s input</th>
<th>Protocol state</th>
</tr>
</thead>
<tbody>
<tr>
<td>flip $L$-th bit</td>
<td>$X[1, K]$</td>
<td>$</td>
</tr>
<tr>
<td>$0$</td>
<td>$X[1, K]$</td>
<td>$</td>
</tr>
<tr>
<td>same $L$-th bit</td>
<td>$X[1, L]$</td>
<td>$</td>
</tr>
<tr>
<td>$0$</td>
<td>$X[1, K]$</td>
<td>$</td>
</tr>
<tr>
<td>$0$</td>
<td>$X[1, L]$</td>
<td>$</td>
</tr>
<tr>
<td>flip $L$-th bit</td>
<td>$X[1, K]$</td>
<td>$</td>
</tr>
<tr>
<td>$1$</td>
<td>$X[1, L]$</td>
<td>$</td>
</tr>
</tbody>
</table>

0-input

1-input
Finally...

**Alice’s input**

<table>
<thead>
<tr>
<th>Flip L-th bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flip L-th bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

**Bob’s input**

\[ X[1, K] \]

\[ X[1, L] \]

**Protocol state**

\[ | \psi \rangle \]

\[ | \varphi \rangle \approx | \psi \rangle \]?
Finally...

<table>
<thead>
<tr>
<th>Alice’s input</th>
<th>Bob’s input</th>
<th>Protocol state</th>
</tr>
</thead>
<tbody>
<tr>
<td>flip L-th bit</td>
<td>$X[1, K]$</td>
<td>$\psi$</td>
</tr>
<tr>
<td></td>
<td>$X[1, L]$</td>
<td>$\varphi \approx \psi$</td>
</tr>
</tbody>
</table>

$$|\psi\rangle = V_K U_X |0\rangle,$$

$$|\psi'\rangle = V_K U_{X'} |0\rangle,$$

$$|\psi''\rangle = V_L U_X |0\rangle$$
Finally...

Alice’s input

<table>
<thead>
<tr>
<th>flip L-th bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

Bob’s input

$X[1, K]$

Protocol state

$| \psi \rangle$

switch index

$X[1, L]$

$| \varphi \rangle \approx | \psi \rangle$ ?

$| \psi \rangle = V_K U_X | 0 \rangle$,  $| \psi' \rangle = V_K U_X' | 0 \rangle$,  $| \psi'' \rangle = V_L U_X | 0 \rangle$

$| \varphi \rangle = V_L U_X' | 0 \rangle$
### Finally...

<table>
<thead>
<tr>
<th>Alice’s input</th>
<th>Bob’s input</th>
<th>Protocol state</th>
</tr>
</thead>
<tbody>
<tr>
<td>flip $L$-th bit</td>
<td>$X[1, K]$</td>
<td>$</td>
</tr>
<tr>
<td>0</td>
<td>switch index</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>$X[1, L]$</td>
<td>$</td>
</tr>
</tbody>
</table>

\[
|\psi\rangle = V_K U_X |0\rangle, \quad |\psi'\rangle = V_K U_{X'} |0\rangle, \quad |\psi''\rangle = V_L U_X |0\rangle
\]

\[
|\varphi\rangle = V_L U_{X'} |0\rangle
\]

\[
|\varphi - \psi| \leq |\psi - \psi''| + |\varphi - \psi''|
\]
Finally...

<table>
<thead>
<tr>
<th>Alice's input</th>
<th>Bob's input</th>
<th>Protocol state</th>
</tr>
</thead>
<tbody>
<tr>
<td>flip (L)-th bit</td>
<td></td>
<td>switch index</td>
</tr>
<tr>
<td></td>
<td>(X[i, K])</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(X[i, L])</td>
<td></td>
</tr>
</tbody>
</table>

\[
|\psi\rangle = V_K U_X |0\rangle, \quad |\psi'\rangle = V_K U_{X'} |0\rangle, \quad |\psi''\rangle = V_L U_X |0\rangle
|\varphi\rangle = V_L U_{X'} |0\rangle

|\varphi - \psi| \leq |\psi - \psi''| + |\varphi - \psi''|

\leq \delta + |V_L U_{X'} |0\rangle - V_L U_X |0\rangle|
Finally...

Alice's input

Bob's input

Protocol state

\[
| \psi \rangle = V_K U_X |0\rangle, \quad | \psi' \rangle = V_K U_X' |0\rangle, \quad | \psi'' \rangle = V_L U_X |0\rangle
\]

| \varphi \rangle = V_L U_X' |0\rangle

| \varphi - \psi | \leq | \psi' - \psi'' | + | \varphi - \psi'' |

\[
\leq \delta + | V_L U_X' |0\rangle - V_L U_X |0\rangle |
\]

\[
= \delta + | V_K U_X' |0\rangle - V_K U_X |0\rangle |
\]
Finally...

<table>
<thead>
<tr>
<th>Alice’s input</th>
<th>Bob’s input</th>
<th>Protocol state</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$X[1, K]$</td>
<td>$</td>
</tr>
<tr>
<td>flip $L$-th bit</td>
<td>switch index</td>
<td>$X[1, L]$</td>
</tr>
<tr>
<td>1</td>
<td>$</td>
<td>\varphi \rangle \approx</td>
</tr>
</tbody>
</table>

\[
| \psi \rangle = V_K U_X | 0 \rangle, \quad | \psi' \rangle = V_K U_X' | 0 \rangle, \quad | \psi'' \rangle = V_L U_X | 0 \rangle
\]
\[
| \varphi \rangle = V_L U_{X'} | 0 \rangle
\]
\[
| \varphi - \psi | \leq | \psi - \psi'' | + | \varphi - \psi'' |
\]
\[
\leq \delta + | V_L U_{X'} | 0 \rangle - V_L U_X | 0 \rangle |
\]
\[
= \delta + | V_K U_{X'} | 0 \rangle - V_K U_X | 0 \rangle |
\]
\[
= \delta + | \psi - \psi' | \leq 2 \delta
\]
Complications swept under the rug

- How we quantify information that is revealed
- Alice and Bob may maintain private workspace
- Information about inputs may increase with each message, penalty for switch increases
- Most of these issues handled à la [JRS’03]
- Leads to a dependence of trade-off on the number of messages
- Connection with streaming algorithms à la [MMN’10] breaks down
Final remarks

• Established a trade-off in quantum information revealed by parties computing Augmented Index

• Stronger results in classical case, with implications for streaming algorithms

• Similar implications likely in the quantum case as well

• Dependence of trade-off on the number of messages unavoidable, without a different notion of information revealed

• Techniques developed for quantum gives conceptually simpler and tighter analysis of classical protocols

• Study of small space (streaming) algorithms is subtle, calls for further exploration