Capacity Approaching Coding for Low-noise Interactive Quantum Communication

Debbie Leung, Ashwin Nayak, Ala Shayeghi, Dave Touchette, Penghui Yao, Nengkun Yu

21st Conference on quantum information processing (QIP 2018)
QuTech, Delft University of Technology
16 January, 2018
Motivation

- **Communication Complexity**
  - Two parties (Alice & Bob) with classical inputs $x$ and $y$, resp.
  - Function $f$ known to both
  - Goal: Compute $f(x, y)$ by communicating over a noiseless channel
Motivation

- **Communication Complexity**
  
  Two parties (Alice & Bob) with classical inputs $x$ and $y$, resp.
  Function $f$ known to both
  Goal: Compute $f(x, y)$ by communicating over a noiseless channel

  - Quantum resources are powerful [Raz99,KR11, ...]
  - Interaction is a powerful resource [KNTZ01, ...]
Motivation

- **Communication Complexity**

  Two parties (Alice & Bob) with classical inputs $x$ and $y$, resp.
  Function $f$ known to both
  Goal: Compute $f(x, y)$ by communicating over a noiseless channel

  - Quantum resources are powerful [Raz99, KR11, ...]
  - Interaction is a powerful resource [KNTZ01, ...]

  - Can we get the same advantage in the noisy quantum communication setting?
  - How robust is communication complexity against noise?
Motivation

- **Communication Complexity**
  
  Two parties (Alice & Bob) with classical inputs $x$ and $y$, resp.
  Function $f$ known to both
  Goal: Compute $f(x, y)$ by communicating over a noiseless channel

  - Quantum resources are powerful [Raz99,KR11, ...]
  - Interaction is a powerful resource [KNTZ01, ...]

  - Can we get the same advantage in the noisy quantum communication setting?
  - How robust is communication complexity against noise?

- **Channel Coding**

  Achieve noiseless one-way communication using a noisy one-way channel

  Channel capacity: Optimal asymptotic achievable rate of such a procedure
Motivation

○ Communication Complexity

Two parties (Alice & Bob) with classical inputs $x$ and $y$, resp.
Function $f$ known to both
Goal: Compute $f(x, y)$ by communicating over a noiseless channel

• Quantum resources are powerful [Raz99,KR11, ...]
• Interaction is a powerful resource [KNTZ01, ...]

- Can we get the same advantage in the noisy quantum communication setting?
- How robust is communication complexity against noise?

○ Channel Coding

Achieve noiseless one-way communication using a noisy one-way channel

Channel capacity: Optimal asymptotic achievable rate of such a procedure

• Studied extensively in one-way setting (classical & quantum) [Shannon, HSW, LSD, ...]
Motivation

- **Communication Complexity**
  - Two parties (Alice & Bob) with classical inputs $x$ and $y$, resp.
  - Function $f$ known to both
  - Goal: Compute $f(x, y)$ by communicating over a noiseless channel
  - Quantum resources are powerful [Raz99,KR11, ...]
  - Interaction is a powerful resource [KNTZ01, ...]
  - Can we get the same advantage in the noisy quantum communication setting?
  - How robust is communication complexity against noise?

- **Channel Coding**
  - Achieve noiseless one-way communication using a noisy one-way channel
  - Channel capacity: Optimal asymptotic achievable rate of such a procedure
  - Studied extensively in one-way setting (classical & quantum) [Shannon, HSW, LSD, ...]
  - What about two-way/interactive capacity of a channel?
Noisy Interactive Quantum Communication

Noiseless protocol $\Pi$

$n$ two-way uses of Identity channel $I$

$\arrowvert \psi_{\text{in}} \rangle_{ABCR}$
Noisy Interactive Quantum Communication

Noiseless protocol $\Pi$

\[
\begin{array}{c}
\text{n two-way uses of Identity channel } I \\
\end{array}
\]

\[
|\psi_{in}^{ABCR}\rangle
\]

\[
U_1 \quad U_2 \quad U_3 \quad U_n \quad U_{n+1}
\]

\[
\begin{array}{c}
A \quad \quad C \quad \quad B \\
A \quad C \quad C \quad C \\
A \quad C \quad \quad \quad \quad B \\
A \quad \quad \quad \quad \quad B
\end{array}
\]

\[
|\psi_{out}\rangle
\]

Simulation protocol $\Pi'$

\[
\begin{array}{c}
\text{n’ two-way uses of noisy channel } N \\
\end{array}
\]

\[
|\psi_{in}^{ABCR}\rangle
\]

\[
E_1 \quad E_2 \quad E_3 \quad E_{n'} \quad E_{n'+1}
\]

\[
\begin{array}{c}
A' \quad \quad A \quad \quad \quad \quad \quad A \\
A' \quad C \quad C \quad B \quad B' \\
A' \quad C \quad \quad \quad \quad B \\
A' \quad \quad \quad \quad \quad B
\end{array}
\]

\[
|\psi_{out}\rangle
\]

\[
\approx |\psi_{out}\rangle
\]

\[
|\varphi_{out}\rangle
\]
Question: How efficiently is it possible to simulate $\Pi$ using a noisy two-way communication channel $N$? How many two-way uses of channel $N$ is needed to simulate $n$ two-way uses of the identity channel?
Noisy Interactive Quantum Communication

**Noiseless protocol Π**

\[ \Psi_{in}^{ABCR} \]

\[ U_1 \]

\[ U_2 \]

\[ U_3 \]

\[ U_{n+1} \]

\[ \Psi_{out} \]

\( n \) two-way uses of Identity channel I

**Simulation protocol Π’**

\[ \Psi_{in}^{ABCR} \]

\[ E_1 \]

\[ E_2 \]

\[ E_3 \]

\[ E_{n' + 1} \]

\[ \Psi_{out} \]

\( n' \) two-way uses of noisy channel N

**Question:** How efficiently is it possible to simulate Π using a noisy two-way communication channel N? How many two-way uses of channel N is needed to simulate \( n \) two-way uses of the identity channel?

Communication rate: \( R := n/n' \)

Interactive/two-way capacity of N: Optimal communication rate in the limit of large \( n \) and vanishing distance \( \delta \)
Challenges

We already know how to protect each message!
Challenges

We already know how to protect each message!  
Not useful with highly interactive protocols!
Challenges

We already know how to protect each message! Not useful with highly interactive protocols!
Challenges

We already know how to protect each message! Not useful with highly interactive protocols!

Constant dilation of each message not sufficient to get constant overall fidelity!
Challenges

We already know how to protect each message! Not useful with highly interactive protocols!

- Standard error correcting codes are inapplicable (classical & Quantum)
  - Need an **online** coding strategy which collectively encodes **multiple** messages together
  - Use interaction as an advantage to detect and correct errors!
Challenges

We already know how to protect each message!  
Not useful with highly interactive protocols!

- Standard error correcting codes are inapplicable (classical & Quantum)
  - Need an online coding strategy which collectively encodes multiple messages together
  - Use interaction as an advantage to detect and correct errors!

- Impossible to directly backtrack to a non-corrupted point

  No-Cloning —> No way to record the state before it gets evolved further

  - Need to actively reverse the simulation
  - Actively reversing the simulation can cause more errors!
Challenges

We already know how to protect each message! Not useful with highly interactive protocols!

- Standard error correcting codes are inapplicable (classical & Quantum)
  - Need an online coding strategy which collectively encodes multiple messages together
  - Use interaction as an advantage to detect and correct errors!

- Impossible to directly backtrack to a non-corrupted point
  - No-Cloning: No way to record the state before it gets evolved further
  - Need to actively reverse the simulation
  - Actively reversing the simulation can cause more errors!
Challenges

We already know how to protect each message! Not useful with highly interactive protocols!

- Standard error correcting codes are inapplicable (classical & Quantum)
  - Need an **online** coding strategy which collectively encodes **multiple** messages together
  - Use interaction as an advantage to detect and correct errors!

- Impossible to directly backtrack to a non-corrupted point
  
  No-Cloning ➔ No way to record the state before it gets evolved further
  
  - Need to **actively reverse** the simulation
  - Actively reversing the simulation can cause more errors!
Challenges

We already know how to protect each message! Not useful with highly interactive protocols!

- Standard error correcting codes are inapplicable (classical & Quantum)
  - Need an **online** coding strategy which collectively encodes **multiple** messages together
  - Use interaction as an advantage to detect and correct errors!

- Impossible to directly backtrack to a non-corrupted point
  - No-Cloning  →  No way to record the state before it gets evolved further
    - Need to **actively reverse** the simulation
    - Actively reversing the simulation can cause more errors!
Challenges

We already know how to protect each message! Not useful with highly interactive protocols!

- Standard error correcting codes are inapplicable (classical & Quantum)
  - Need an **online** coding strategy which collectively encodes **multiple** messages together
  - Use interaction as an advantage to detect and correct errors!

- Impossible to directly backtrack to a non-corrupted point
  - No-Cloning  ➔  No way to record the state before it gets evolved further
    - Need to **actively reverse** the simulation
    - Actively reversing the simulation can cause more errors!
Challenges

We already know how to protect each message! Not useful with highly interactive protocols!

- Standard error correcting codes are inapplicable (classical & Quantum)
  - Need an **online** coding strategy which collectively encodes **multiple** messages together
  - Use interaction as an advantage to detect and correct errors!

- Impossible to directly backtrack to a non-corrupted point
  - No-Cloning \[ \rightarrow \] No way to record the state before it gets evolved further
  - Need to **actively reverse** the simulation
  - Actively reversing the simulation can cause more errors!
**Previous Work**

**Classical:**
- Noisy interactive communication problem introduced by Schulman [Sch92, Sch93]
  
  Possible to simulating noiseless interactive communication over a two-way noisy channel with constant overhead ($C > 0$)

- Active field of research:
  - Results focused on improving tolerable error-rate and computational efficiency:
    
    [BR11, GMS11, BK12, FGOS13, BN13, BE14, GH14, GHS14, BKN14, EGH15, ...]

  Mostly based on tree codes, Huge communication overhead even for vanishing error rate

  - [KR13], [Hae14] introduced capacity approaching codes:
    
    Characterized interactive capacity up to leading order: $C \to 1$ with error-rate $\epsilon \to 0$

    - **Random noise:** $C > 1 - O(\sqrt{\epsilon})$
    - **Adversarial noise:** $C > 1 - O\left(\sqrt{\epsilon \log \log \frac{1}{\epsilon}}\right)$

- More recent results: [BEGH16, GH17, HV17, BE17, ...]

**Quantum:**
- Recently, [BNTTU14] proved constant factor communication overhead is possible ($C > 0$)

  Computationally inefficient, Huge communication overhead even for vanishing error rate ($C \ll 1$)
Main Result

**Theorem:** Rate $1 - O(\sqrt{\epsilon})$ is achievable, with success prob. $1 - 2^{-\Omega(n\epsilon)}$, over fully adversarial qubit channel of error rate at most $\epsilon$. 
Main Result

**Theorem:** Rate $1 - O(\sqrt{\epsilon})$ is achievable, with success prob. $1 - 2^{-\Omega(n\epsilon)}$, over fully adversarial qubit channel of error rate at most $\epsilon$.

- First **capacity approaching** result in noisy interactive quantum communication
  Characterizing interactive/two-way capacity to leading order: $C \to 1$ as error-rate $\epsilon \to 0$

- First **computationally efficient** coding scheme
  Computational complexity of coding operations: $O(n^2)$

- **Plain quantum model**: No pre-shared resources
  Outperforms conjectured optimal bound in plain classical model!
Main Result

**Theorem:** Rate $1 - O(\sqrt{\varepsilon})$ is achievable, with success prob. $1 - 2^{-\Omega(n\varepsilon)}$, over fully adversarial qubit channel of error rate at most $\varepsilon$.

- First **capacity approaching** result in noisy interactive quantum communication
  Characterizing interactive/two-way capacity to leading order: $C \rightarrow 1$ as error-rate $\varepsilon \rightarrow 0$

- First **computationally efficient** coding scheme
  Computational complexity of coding operations: $O(n^2)$

- **Plain quantum model:** No pre-shared resources
  Outperforms conjectured optimal bound in plain classical model!

**Note:** This work is not an extension of [BNTTU14]:

[BNTTU14] : Based on tree codes (computationally inefficient)
  $C \ll 1$ even for vanishing error $\varepsilon \rightarrow 0$
  Tolerates adversarial error rates up to 1/2
Development of Framework

Focus on adversarial noise (includes random noise)

Teleportation-based Model

- Perfect pre-shared entanglement
- Noisy classical communication
- Large alphabet

Plain Model

- No pre-shared entanglement
- Noisy quantum communication
- Large alphabet

Large Alphabet

- Perfect pre-shared entanglement
- Noisy classical communication
- Large alphabet

Small Alphabet

- No pre-shared entanglement
- Noisy quantum communication
- Small alphabet
Development of Framework

Focus on adversarial noise (includes random noise)

Teleportation-based Model

- Perfect pre-shared entanglement
- Noisy classical communication
- Large alphabet

Plain Model

- No pre-shared entanglement
- Noisy quantum communication
- Large alphabet

Large Alphabet

- Perfect pre-shared entanglement
- Noisy classical communication
- Large alphabet
- Large Alphabet

Small Alphabet

- Perfect pre-shared entanglement
- Noisy classical communication
- Small alphabet
- Small Alphabet

- No pre-shared entanglement
- Noisy quantum communication
- Small alphabet
Noisy Interactive Communication: Natural Approach

Haeupler’s Template (Classical)

- Both parties conduct their original conversation as if there were no noise
- At regular intervals exchange concise summaries of the conversation so far
- If summaries consistent, continue
- Otherwise, error detected, backtrack to earlier stage and resume
Noisy Interactive Communication: Natural Approach

Haeupler’s Template (Classical)

- Both parties conduct their original conversation as if there were no noise
- At regular intervals exchange concise summaries of the conversation so far
- If summaries consistent, continue
- Otherwise, error detected, backtrack to earlier stage and resume

- An online error-correcting code over multiple messages
  - Trivial encoding of each message
  - Summaries measure the error syndrome
Noisy Interactive Communication: Natural Approach

Haeupler’s Template (Classical)

- Both parties conduct their original conversation as if there were no noise
- At regular intervals exchange concise summaries of the conversation so far
- If summaries consistent, continue
- Otherwise, error detected, backtrack to earlier stage and resume

- An online error-correcting code over multiple messages
  - Trivial encoding of each message
  - Summaries measure the error syndrome
- Efficient: involves evaluating hash functions
Noisy Interactive Communication: Natural Approach

Haeupler’s Template (Classical)

- Both parties conduct their original conversation as if there were no noise
- At regular intervals exchange concise summaries of the conversation so far
- If summaries consistent, continue
- Otherwise, error detected, backtrack to earlier stage and resume

- An online error-correcting code over multiple messages
  - Trivial encoding of each message
  - Summaries measure the error syndrome
- Efficient: involves evaluating hash functions
- As simulation proceeds, gain more trust in earlier conversation → any detected error is recent with high prob.
Noisy Interactive Communication: Natural Approach

Haeupler’s Template (Classical)

- Both parties conduct their original conversation as if there were no noise
- At regular intervals exchange concise summaries of the conversation so far
- If summaries consistent, continue
- Otherwise, error detected, backtrack to earlier stage and resume

Remarks:

- How frequently check for inconsistency?
  - More checks → communication lost even if no error
  - More checks → detect errors earlier, less communication lost
Noisy Interactive Communication: Natural Approach

Haeupler’s Template (Classical)

- Both parties conduct their original conversation as if there were no noise
- At regular intervals exchange concise summaries of the conversation so far
- If summaries consistent, continue
- Otherwise, error detected, backtrack to earlier stage and resume

Remarks:

- How frequently check for inconsistency?
  - More checks → communication lost even if no error
  - More checks → detect errors earlier, less communication lost

- How to backtrack?
  - Requirement: communication wasted by a single error should be constant!
Our Framework

Follow natural approach!

Make sure both parties know joint quantum state before deciding their next action!

- Introduce sufficient but concise data structure to track:
  - Stage in protocol
  - Type of action in each iteration
  - Teleportation measurement outcomes
  - Received instructions for teleportation decoding
  - Recovery operations
  - Which MESs to use next for teleportation
  - ...

- Each party maintains their own data and an estimate of other party's data
- At the beginning of each iteration, check if the estimates match the actual data (by hashing)
  - No → resolve the inconsistency in classical data
    - Adapt synchronization mechanism developed by [Hae14] in classical setting
  - Yes → Compute the joint state → Decide next action
In each iteration, Alice & Bob engage in one of three actions:

1. Simulate next block in $\Pi$
2. Reverse the last block of simulation
3. Exchange classical data
In each iteration, Alice & Bob engage in one of three actions:

1. Simulate next block in $\Pi$
2. Reverse the last block of simulation
3. Exchange classical data
Our Framework

In each iteration, Alice & Bob engage in one of three actions:

1. Simulate next block in $\Pi$
2. Reverse the last block of simulation
3. Exchange classical data

Error or hash collision $\rightarrow$ different actions!
Out-of-Sync Teleportation

What if Alice proceeds with simulation of $\Pi$ (forward or reverse) while Bob exchanges classical data?!

- Alice: teleports quantum data, interprets Bob’s classical data as teleportation measurement outcomes
- Bob: sends classical data, interprets Alice’s instructions for teleportation decoding as classical data
- They become out-of-sync on which MESs to use to teleport next
Out-of-Sync Teleportation

What if Alice proceeds with simulation of $\Pi$ (forward or reverse) while Bob exchanges classical data?!

- Alice: teleports quantum data, interprets Bob’s classical data as teleportation measurement outcomes
- Bob: sends classical data, interprets Alice’s instructions for teleportation decoding as classical data
- They become out-of-sync on which MESs to use to teleport next

Can Alice and Bob recover from this?!
Out-of-Sync Teleportation

- Information does not leak to environment (adversary)
  Quantum data reside somewhere in the **closed system**
Out-of-Sync Teleportation

- Information does not leak to environment (adversary)
  - Quantum data reside somewhere in the **closed system**
- Need to **redirect** quantum data back to $A, B, C$ registers
  - Resolve inconsistencies in classical data
  - Determine which MES to use next
  - “Complete the teleportations”
Development of Framework

Focus on adversarial noise (includes random noise)

Teleportation-based Model

- Perfect pre-shared entanglement
- Noisy classical communication
- Large alphabet

Plain Model

- No pre-shared entanglement
- Noisy quantum communication
- Large alphabet

Large Alphabet

- Perfect pre-shared entanglement
- Noisy classical communication
- Large alphabet

Small Alphabet

- Perfect pre-shared entanglement
- Noisy classical communication
- Small alphabet

- No pre-shared entanglement
- Noisy quantum communication
- Small alphabet
Development of Framework

Focus on adversarial noise (includes random noise)

Teleportation-based Model

- Perfect pre-shared entanglement
- Noisy classical communication
- Large alphabet

Plain Model

- No pre-shared entanglement
- Noisy quantum communication
- Large alphabet

Large Alphabet

- Perfect pre-shared entanglement
- Noisy classical communication
- Large alphabet

Small Alphabet

- Perfect pre-shared entanglement
- Noisy classical communication
- Small alphabet

- No pre-shared entanglement
- Noisy quantum communication
- Small alphabet
Framework for Plain Model

- A similar data structure is used to maintain a global view of simulation
Framework for Plain Model

- A similar data structure is used to maintain a global view of simulation
- To protect the messages: Teleportation → Quantum Vernam Cipher [Leu00]
Framework for Plain Model

- A similar data structure is used to maintain a global view of simulation
- To protect the messages: Teleportation \(\rightarrow\) Quantum Vernam Cipher [Leu00]

Key features: Allows for recycling MESs when no errors
Detection of errors with distributed syndrome
Framework for Plain Model

- A similar data structure is used to maintain a global view of simulation
- To protect the messages: Teleportation $\rightarrow$ Quantum Vernam Cipher [Leu00]

  Key features: Allows for recycling MESs when no errors
  Detection of errors with distributed syndrome

- Use quantum hashing due to [BDSW96] to detect errors
Framework for Plain Model

- A similar data structure is used to maintain a global view of simulation
- To protect the messages: Teleportation → Quantum Vernam Cipher [Leu00]

Key features: Allows for recycling MESs when no errors
Detection of errors with distributed syndrome

- Use quantum hashing due to [BDSW96] to detect errors
- Distributing small amount of entanglement is sufficient
  - Use a fraction to generate a secret key
  - Use the rest for quantum hashing and QVC
  - Recycle entanglement as needed
Framework for Plain Model

- A similar data structure is used to maintain a global view of simulation
- To protect the messages: Teleportation → Quantum Vernam Cipher [Leu00]

  Key features: Allows for recycling MESs when no errors
  Detection of errors with distributed syndrome

- Use quantum hashing due to [BDSW96] to detect errors
- Distributing small amount of entanglement is sufficient
  - Use a fraction to generate a secret key
  - Use the rest for quantum hashing and QVC
  - Recycle entanglement as needed
- New Obstacles: out-of-sync QVC, out-of-sync hashing, out-of-sync recycling
Open Questions

- Is $1 - O(\sqrt{\epsilon})$ the optimal achievable rate?
Open Questions

• Is $1 - O(\sqrt{\epsilon})$ the optimal achievable rate?
• If so, what is the constant in $O(\sqrt{\epsilon})$?
Open Questions

• Is $1 - O(\sqrt{\epsilon})$ the optimal achievable rate?
• If so, what is the constant in $O(\sqrt{\epsilon})$?
• What about non-alternating protocols?
Open Questions

• Is $1 - O(\sqrt{\varepsilon})$ the optimal achievable rate?
• If so, what is the constant in $O(\sqrt{\varepsilon})$?
• What about non-alternating protocols?
• What if local operations are also noisy? Extension to fault-tolerant setting
Open Questions

- Is $1 - O(\sqrt{\epsilon})$ the optimal achievable rate?
- If so, what is the constant in $O(\sqrt{\epsilon})$?
- What about non-alternating protocols?
- What if local operations are also noisy? Extension to fault-tolerant setting
- Privacy-preserving interactive communication
Open Questions

• Is \(1 - O(\sqrt{\varepsilon})\) the optimal achievable rate?
• If so, what is the constant in \(O(\sqrt{\varepsilon})\)?
• What about non-alternating protocols?
• What if local operations are also noisy? Extension to fault-tolerant setting
• Privacy-preserving interactive communication
• ...


Open Questions

- Is $1 - O(\sqrt{\epsilon})$ the optimal achievable rate?
- If so, what is the constant in $O(\sqrt{\epsilon})$?
- What about non-alternating protocols?
- What if local operations are also noisy? Extension to fault-tolerant setting
- Privacy-preserving interactive communication
- ...

Thanks!
Crude Analysis for Rate

Noiseless protocol of length $n$, $\frac{n}{r}$ blocks of length $r$

Number of errors = $\epsilon \cdot \frac{n}{C} = O(\epsilon n)$

Number of iterations to recover from an error = $O(1)$

Total # of iterations = # of iteration of forward simulation + # of iterations of recovery = $\frac{n}{r} + O(\epsilon n)$

Communication in each iteration = $r + O(1)$ (for checks)

Total communication = $\left(\frac{n}{r} + O(\epsilon n)\right)(r + O(1)) = n \left(1 + O(\epsilon r + \frac{1}{r})\right) = n \left(1 + O(\sqrt{\epsilon})\right)$

for $r = \Theta \left(\frac{1}{\sqrt{\epsilon}}\right)$

$$R = \frac{n}{n \left(1 + O(\sqrt{\epsilon})\right)} = 1 - O(\sqrt{\epsilon})$$