Two-batch liar games on a general bounded channel

Robert B. Ellis\textsuperscript{1}  Kathryn L. Nyman\textsuperscript{2}

\textsuperscript{1}Illinois Institute of Technology

\textsuperscript{2}Loyola University, Chicago

AMS/PTM Joint Meeting, Warsaw
Outline

1 Background
   - The basic liar game
   - Motivating the general bounded channel

2 The liar game on a general bounded channel
   - Definitions and game play

3 New Results
   - Examples
   - A general sphere bound
   - A winning condition for Carole
   - A winning condition for Paul
   - Proof of Paul’s bound
Basic liar game setting

Two-person game:

1. **Carole** picks a number $x \in [n] := \{1, \ldots, n\}$
2. **Paul** asks $q$ questions to determine $x$:
   - given $[n] = A_1 \cup A_2 \cup \cdots \cup A_t$,
   - for what $i$ is $x \in A_i$?

Playing optimally, **Carole** answers with an adversarial strategy; it’s a perfect information game.

**Catch:** **Carole** is allowed to lie at most $k$ times.
Example ternary game

\( t = 3 \) (Ternary coding).

- **Paul** partitions \([n] = A_1 \cup A_2 \cup A_3\) and asks “for what \(i\) is \(x \in A_i\)?”
- **Carole** answers 1, 2, or 3

Example. \(n = 6, q = 4, t = 3, k = 1\)

<table>
<thead>
<tr>
<th>Rnd</th>
<th>(A_1)</th>
<th>(A_2)</th>
<th>(A_3)</th>
<th>Carole</th>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>{1, 2}</td>
<td>{3, 4}</td>
<td>{5, 6}</td>
<td>2</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>2</td>
<td>{3}</td>
<td>{4}</td>
<td>{1, 2, 5, 6}</td>
<td>3</td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>3</td>
<td>{1, 2}</td>
<td>{3, 4}</td>
<td>{5, 6}</td>
<td>3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>4</td>
<td>{5}</td>
<td>{6}</td>
<td>∅</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

Therefore \(x = 5\).
Binary symmetric case

- $t = 2$ binary case $\leftrightarrow$ “is $x \in A_1$?”
- symmetric lies: Carole may
  - lie with Yes when truth is No
  - lie with No when truth is Yes

**Question.** Given $q$, what is the maximum $n$ for which Paul has a winning strategy to find $x$?

- $k = 0$, binary search, $n = 2^q$
Binary symmetric case

- $t = 2$ binary case $\leftrightarrow$ “is $x \in A_1$?”
- symmetric lies: Carole may
  - lie with Yes when truth is No
  - lie with No when truth is Yes

**Question.** Given $q$, what is the maximum $n$ for which Paul has a winning strategy to find $x$?

- $k = 0$, binary search, $n = 2^q$
- $k = 1$, Pelc (1987)
- $k < \infty$, Spencer (1992) (*up to bounded additive error*)
Binary symmetric case

- $t = 2$ binary case $\leftrightarrow$ “is $x \in A_1$?”
- symmetric lies: Carole may
  - lie with Yes when truth is No
  - lie with No when truth is Yes

**Question.** Given $q$, what is the maximum $n$ for which Paul has a winning strategy to find $x$?

- $k = 0$, binary search, $n = 2^q$
- $k = 1$, Pelc (1987)
- $k < \infty$, Spencer (1992) (*up to bounded additive error*)
- $k/q \to f \in (0, 1/2)$, Berlekamp (1962+); Alshwede, Deppe, Lebedev (2005) (*still partially open*)
**Binary symmetric case, \( k = 1 \)**

**Question.** Given \( q \), what is the maximum \( n \) for which Paul has a winning strategy to find \( x \)?

- Let \( k = 1, \ y \in [n] \)
- \( q + 1 \) ways for \( y \) to be the distinguished element:

<table>
<thead>
<tr>
<th>Game response string ( w \in [2]^q )</th>
<th>0 lies</th>
<th>( w_1 )</th>
<th>( w_2 )</th>
<th>( w_3 )</th>
<th>( \cdots )</th>
<th>( w_{q-1} )</th>
<th>( w_q )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \overline{w_1} )</td>
<td>*</td>
<td>*</td>
<td>\cdots</td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>1 lie</td>
<td>( w_1 )</td>
<td>( \overline{w_2} )</td>
<td>*</td>
<td>\cdots</td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( w_1 )</td>
<td>( w_2 )</td>
<td>( w_3 )</td>
<td>( \cdots )</td>
<td>( w_{q-1} )</td>
<td>( \overline{w_q} )</td>
<td></td>
</tr>
</tbody>
</table>

**Sphere Bound** \( y, y' \) can’t both be \( x \) \( \implies \) \( n \leq 2^q / \binom{q}{\leq 1} \)
Binary symmetric case, \( k < \infty \)

\( X_i := \) elements of \([n]\) with \( i \) accumulated lies

Paul balances \( A_1 \cup A_2 \) by solving each round

\[
|A_1 \cap X_i| = \frac{|X_i|}{2}, \quad \text{for } 0 \leq i \leq k.
\]

**Sphere Bound** \((\binom{q}{\leq k})\) ways for \( y \in [n] \) to be the distinguished element

\[
\implies n \leq 2^q / \binom{q}{\leq k}
\]
Asymmetric lying

- **asymmetric lies**: Carole may
  - lie with *Yes* (1) when truth is *No* (2)
  - But not vice versa!

Called the *Z*-channel

\[
\begin{array}{c}
\text{1} \\
\rightarrow \\
\text{2} \\
	ext{Z-channel}
\end{array}
\]  
\[
\begin{array}{c}
\text{1} \\
\rightarrow \\
\text{2} \\
	ext{Z’-channel}
\end{array}
\]

- \( k < \infty \), Dumitriu & Spencer (2004)
- \( k < \infty \) w/improved asymptotics, Spencer & Yan (2003)

Asymmetric strategy: still based on **balancing**.
In 2005 Ioana Dumitriu was giving a talk on liar games, and Nathan Linial asked:

What if Paul knows that Carole is lying according to one of the Z-channels, but not which one?
A motivating question

Meanwhile, an equivalent question: What is the liar game version of packing/covering with unidirectional Hamming balls?

Our answer: Generalize the “channel” constraining Carole’s lies as much as possible.
A closer look: game lie strings

<table>
<thead>
<tr>
<th>Rnd</th>
<th>Paul</th>
<th>Carole</th>
<th>6's lie string</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$A_1$</td>
<td>$A_2$</td>
<td>$A_3$</td>
</tr>
<tr>
<td>1</td>
<td>{1, 2}</td>
<td>{3, 4}</td>
<td>{5, 6}</td>
</tr>
<tr>
<td>2</td>
<td>{3}</td>
<td>{4}</td>
<td>{1, 2, 5, 6}</td>
</tr>
<tr>
<td>3</td>
<td>{1, 2}</td>
<td>{3, 4}</td>
<td>{5, 6}</td>
</tr>
<tr>
<td>4</td>
<td>{5}</td>
<td>{6}</td>
<td>$\emptyset$</td>
</tr>
</tbody>
</table>

Truthful string for $y = 6$: $w' = 3\ 3\ 3\ 2$

Lie string for $y = 6$: $u = 3\ 2\ 2\ 1$

Response string: $w = 2\ 3\ 3\ 1$

Write $u = (3, 2)(2, 1)$; we say $w' \rightarrow w$
The general bounded $t$-ary channel

- **Lies**: $L(t) := \{(a, b) \in [t] \times [t] : a \neq b\}$  
  (truth = $a$, Carole: $b$)
- **Lie strings**: $L(t)^j := \{(a_1, b_1) \cdots (a_j, b_j) : (a_i, b_i) \in L(t)\}$
- **Empty string**: $L(t)^0 := \{\epsilon\}$

**Definition (General bounded channel)**

Fix $k \geq 0$. A channel $C$ of order $k$ is an arbitrary subset

$$C \subseteq \bigcup_{j=0}^{k} L(t)^j,$$

such that $C \cap L(t)^k \neq \emptyset$. 

(July 31, 2007)
Element survival and winning for Paul

Definition
An element $y \in [n]$ survives the game iff its lie string is in $C$.

Definition
Paul wins the original liar game iff at most one element survives after $q$ rounds.
Paul wins the pathological liar game iff at least one element survives after $q$ rounds.

$$A_C(q) := \max n \quad \text{such that Paul can win the original liar game with } n \text{ elements.}$$
$$A_C^*(q) := \min n \quad \text{such that Paul can win the pathological liar game with } n \text{ elements.}$$
Example channels

- Binary, symmetric, two lies.  \((t = 2, k = 2)\)

  \[
  C = \{ \epsilon, (1, 2), (2, 1), (1, 2)(1, 2), (1, 2)(2, 1), (2, 1)(2, 1), (2, 1)(1, 2) \}
  \]

  \[
  \frac{2^q}{q^{\leq 2}} - O(1) = A_C(q) \leq A^*_C(q) = \frac{2^q}{q^{\leq 2}} + O(1)
  \]

  Guzicki ('90); E., Ponomarenko, Yan ('05)

- Binary, \(Z\)-channel, two lies.  \((t = 2, k = 2)\)

  \[
  C = \{ \epsilon, (2, 1), (2, 1)(2, 1) \}
  \]

  \[
  A_C(q), A^*_C(q) \sim \frac{2^{q+2}}{q^{\leq 2}}, \quad \text{Spencer, Yan ('03); here}
  \]
Example channels (con’t)

- **Binary, unidirectional, two lies.** \( (t = 2, k = 2) \)

\[
C = \{ \epsilon, (1, 2), (2, 1), (1, 2)(1, 2), (2, 1)(2, 1) \}
\]

\[
A_C(q), A_C^*(q) \sim \frac{2^{q+1}}{q}, \text{ here}
\]

- **Selective lies.**
  - Pick arbitrary \( L' \subseteq L(t) \).
  - Let \( C = \bigcup_{j=0}^k (L')^j \).

\[
A_C(q), A_C^*(q) \sim \frac{t^{q+k}}{|L'|^k \left( \begin{array}{c} q \\ \leq k \end{array} \right)}
\]

Dumitriu, Spencer (‘05); here
The proposed sphere bound

- **Select** Paul’s strategy tree to be entirely random partitions 
  \[ [n] = A_1 \cup \cdots \cup A_t \]
- The **expected number of response strings** for which \( y \) survives is:
  \[
  \sum_{u \in C} \binom{q}{|u|} t^{-|u|} \sim |C \cap L(t)^k| \binom{q}{k} t^{-k}.
  \]

<table>
<thead>
<tr>
<th>Truthful string for ( y )</th>
<th>( w' = w'<em>1 \cdots w'</em>{i_1} \cdots w'<em>{i</em>\ell} \cdots w'_i \cdots w'_j \cdots w'_q )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lie string for ( y )</td>
<td>( u = )</td>
</tr>
<tr>
<td>Response string</td>
<td>( w = w_1 \cdots w_{i_1} \cdots w_{i_\ell} \cdots w_i \cdots w_j \cdots w_q )</td>
</tr>
</tbody>
</table>

**Compatibility:** \( \Pr(w'_{i_\ell} = a_{i_\ell}) = t^{-1} \)
The proposed sphere bound

- **Select** Paul’s strategy tree to be entirely random partitions
  \[ [n] = A_1 \cup \cdots \cup A_t \]
- The expected number of response strings for which \( y \) survives is:
  \[
  \sum_{u \in C} \binom{q}{|u|} t^{-|u|} \sim |C \cap L(t)^k| \binom{q}{k} t^{-k}.
  \]

**Definition (Asymptotic Sphere Bound)**

For \( q \) rounds, base \( t \), and an order \( k \) channel \( C \), the sphere bound is

\[
SB_C(q) := \frac{t^{q+k}}{|C \cap L(t)^k| \binom{q}{k}}.
\]

(July 31, 2007)
Carole’s bound

Theorem (Carole’s bound)

\[
A_C(q) \leq SB_C(q)(1 + o(1)),
\]
\[
A^*_C(q) \geq SB_C(q)(1 - o(1)).
\]

Proof idea.

- Most strings of \([t]^q\) are balanced.
- The response string set for which \(y\) survives “looks random” when all its strings are balanced.
- \(n\) too large \(\Rightarrow\) response string sets collide
  too small \(\Rightarrow\) response string sets fail to cover \([t]^q\)
Paul’s bound

Theorem (Paul’s bound)

\[ A_C(q) \geq SB_C(q)(1 - o(1)), \]
\[ A^*_C(q) \leq SB_C(q)(1 + o(1)). \]

Furthermore, (1) we may restrict Paul to two nonadaptive batches of questions of sizes \( q_1 \) and \( q_2 \), with

\[ q_1 + q_2 = q \quad \text{and} \quad (\log_t q)^{3/2} \ll q_2 \leq \text{cst} \cdot q^{k/(2^k-1)}, \]

(2) the response sets for \( A_C(q) \) are a subset of those for \( A^*_C(q) \).

Remark. Proof builds on techniques of Dumitriu&Spencer.
Proof of Paul's bound

(M, r)-balanced strings in [t]^Q

\[
\frac{1}{t} \left\lceil \frac{Q}{M} \right\rceil - r(t-1) - \Theta(1) \leq \#1's, \#2's \leq \frac{1}{t} \left\lfloor \frac{Q}{M} \right\rfloor + r
\]

Lemma

Let \( u = (a_1, b_1) \cdots (a_j, b_j) \), and \( w \in [t]^Q \) be (M, r)-balanced. Then

\[
\binom{M}{j} \left( \frac{1}{t} \left\lceil \frac{Q}{M} \right\rceil - r(t-1) - \Theta(1) \right)^j \leq \left| \{ w' : w' \xrightarrow{u} w \} \right| \leq \binom{M+j-1}{j} \left( \frac{1}{t} \left\lfloor \frac{Q}{M} \right\rfloor + r \right)^j,
\]

\[
\binom{Q}{j} t^{-j}(1 - o(1)) \leq \left| \{ w' : w' \xrightarrow{u} w \} \right| \leq \binom{Q}{j} t^{-j}(1 + o(1)).
\]

(July 31, 2007)
First batch of $q_1$ questions

(Proof illustrated with $C = \{ \epsilon, (1, 2), (2, 1), (1, 2)(1, 2), (2, 1)(2, 1) \}$.)

Paul maps $n$ evenly to $(M, r)$-balanced vertices of $[t]^{q_1}$

Paul partitions $[n] q_1$ times based on each digit in mapping
Suppose Carole responds with balanced $w \in [t]^{q_1}$.
Which $y \in [n]$ survive?

Any $y$ identified with $w'$ such that:
- $u \in C$, and
- $w' \xrightarrow{u} w$
Paul’s second batch of $q_2$ questions

- $y$’s survive in various ways
- Fit $y$’s which can take more lies inside disjoint Hamming balls
- $(M, r)$-balance $\Rightarrow$ control on $|\{ w^{(i)} : w^{(i)} \xrightarrow{\cup} w \}|$, $|\{ z : z \xrightarrow{\vee} z' \}|$
- Greedily pack other $y$’s in unoccupied space
First batch, pathological case

(Proof illustrated with $C = \{\varepsilon, (1, 2), (2, 1), (1, 2)(1, 2), (2, 1)(2, 1)\}$.)

$$n = SB_C(q)(1 - o(1))$$
First batch, pathological case

(Proof illustrated with $C = \{\epsilon, (1, 2), (2, 1), (1, 2)(1, 2), (2, 1)(2, 1)\}$.)

- Paul adds negligibly many elements evenly over $[t]^{q_1}$

$$n = SB_C(q)(1 - o(1))$$
Paul’s second batch, pathological case
Paul’s second batch, pathological case

- Count only additional y’s for which Carole may not lie again
- Greedily convert packing into covering in $[t]^{q_2}$
Summary

Theorem

\[ SB_C(q)(1 + o(1)) \geq A_C(q) \geq SB_C(q)(1 - o(1)), \]
\[ SB_C(q)(1 - o(1)) \leq A^*_C(q) \leq SB_C(q)(1 + o(1)). \]

Furthermore, (1) we may restrict Paul to two nonadaptive batches of questions of sizes \( q_1 \) and \( q_2 \), with

\[ q_1 + q_2 = q \quad \text{and} \]
\[ (\log_t q)^{3/2} \ll q_2 \leq \text{cst} \cdot q^{k/(2^k-1)}, \]

(2) the response sets for \( A_C(q) \) are a subset of those for \( A^*_C(q) \).
Concluding remarks and open questions

Open Questions.

- Can we further reduce or eliminate completely the adaptiveness?
- Can these techniques be used to improved the asymptotic best known packings and coverings of $[t]^q$ with fixed-radius Hamming balls (not tight for radius $\geq 2$)?
- Will these techniques work for coin-weighing, fault-testing, and related search problems?

Thank you very much.
Preprint at http://math.iit.edu/~rellis/.

(July 31, 2007)