Alice and Bob Communication Complexity and Privacy under dependent inputs

Sergio Rajsbaum
UNAM, Mexico

Fontainebleau 2021

Organization

Part I : Information theory perspective, number of bits needed to communicate each others inputs

Part II: Wait-free perspective, how many bits are needed to solve a task

Part III: Russian cards perspective, how many bits are needed to communicate each others inputs *privately*

Part I

Information theory perspective

Information transmission with dependent inputs: interaction

PART III: SOURCE CODING
from the survey paper
(details in other paper cited later on):

Janos Korner and Alon Orlitsky, Zero-error information theory, IEEE Transactions on Information Theory, 1998

Hungarian, Sapienza University of Rome
2014 Claude E. Shannon Award
With Imre Csiszár book: Information Theory: Coding Theorems for Discrete Memoryless Systems

Israeli , at University of California, San Diego
2021 Claude E. Shannon Award (talk in ISIT July)

zero-error information theory

Shannon was the first to realize the significance of zero-error information theory.

His classic paper, “The zero-error capacity of a noisy channel”, is a true gem in graph theory and is one of his most cited articles.

The subject was subsequently studied extensively

: © Stanley Rowin

Tinkerer, Prankster, and Father of Information Theory, one of the greatest electrical engineering heroes of all time.
https://spectrum.ieee.org/tech-history/cyberspace/claude-shannon-tinkerer-prankster-and-father-of-information-theory

Source coding

basic problem

Source coding

basic problem

- Alice wants to transmit to Bob an element x from a set X. The element x is selected with probability $p(x)$

Source coding

basic problem

- Alice wants to transmit to Bob an element x from a set X. The element x is selected with probability $p(x)$
- Alice uses an encoding $P_A(x) = m$ to save bits.

Source coding

basic problem

- Alice wants to transmit to Bob an element x from a set X. The element x is selected with probability $p(x)$
- Alice uses an encoding $P_A(x) = m$ to save bits.
- Bob knows both X and P_A, and should learn x without error when it gets $P_A(x)$

Source coding

basic problem

- Alice wants to transmit to Bob an element x from a set X. The element x is selected with probability $p(x)$
- Alice uses an encoding $P_A(x) = m$ to save bits.
- Bob knows both X and P_A, and should learn x without error when it gets $P_A(x)$
- For a single instance (transmit one x, once), it is well known that the smallest number of bits needed in the *worst case* is $\lceil \log_2 |X| \rceil$, and the smallest *expected* number of bits is the entropy, achieved by Huffman coding, is between $H(X)$ and $H(X) + 1$

Source coding

basic problem

- Alice wants to transmit to Bob an element x from a set X. The element x is selected with probability $p(x)$
- Alice uses an encoding $P_A(x) = m$ to save bits.
- Bob knows both X and P_A, and should learn x without error when it gets $P_A(x)$
- For a single instance (transmit one x, once), it is well known that the smallest number of bits needed in the *worst case* is $\lceil \log_2 |X| \rceil$, and the smallest *expected* number of bits is the entropy, achieved by Huffman coding, is between $H(X)$ and $H(X) + 1$
- For n independent instances, it follows that $n\lceil \log_2 |X| \rceil$ bits are needed in the worst case, and on average between $nH(X)$ and $nH(X) + 1$.

Source coding

basic problem

- Alice wants to transmit to Bob an element x from a set X. The element x is selected with probability $p(x)$
- Alice uses an encoding $P_A(x) = m$ to save bits.
- Bob knows both X and P_A, and should learn x without error when it gets $P_A(x)$
- For a single instance (transmit one x, once), it is well known that the smallest number of bits needed in the *worst case* is $\lceil \log_2 |X| \rceil$, and the smallest *expected* number of bits is the entropy, achieved by Huffman coding, is between $H(X)$ and $H(X) + 1$
- For n independent instances, it follows that $n\lceil \log_2 |X| \rceil$ bits are needed in the worst case, and on average between $nH(X)$ and $nH(X) + 1$.
- Asymptotically, per instance, $\log_2 |X|$, and $H(X)$, resp.

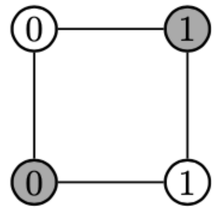
Side information
correlated inputs

Side information
correlated inputs

- Alice wants to transmit x using an encoding $P_A(x)$ to save bits, but now Bob knows something about the inputs of A. Bob should learn x, with no probability of error.

Side information
correlated inputs

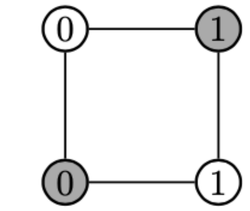
- Alice wants to transmit x using an encoding $P_A(x)$ to save bits, but now Bob knows something about the inputs of A. Bob should learn x, with no probability of error.
- I*nput graph* $\mathcal{I}$: bipartite, vertices of A correspond to her possible inputs X, and the vertices of B correspond to his possible inputs Y. An edge (x, y) means that it is possible that A has x and B has y.



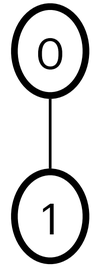
Alice white, Bob black

Side information
correlated inputs

- Alice wants to transmit x using an encoding $P_A(x)$ to save bits, but now Bob knows something about the inputs of A. Bob should learn x, with no probability of error.
- I*nput graph* $\mathcal{I}$: bipartite, vertices of A correspond to her possible inputs X, and the vertices of B correspond to his possible inputs Y. An edge (x, y) means that it is possible that A has x and B has y.
- *Characteristic graph*: the vertices are X, there is an edge (x, x') whenever (x, y) and (x', y) are possible inputs, for some input y of B



Alice white, Bob black



Main theorem

Single instance

Main theorem

Single instance

- The smallest number of possible messages the informant must transmit for a single instance is $\chi(G)$, the chromatic number of the characteristic graph G. In terms of bits, $\log_2 \chi(G)$.

Main theorem

Single instance

- The smallest number of possible messages the informant must transmit for a single instance is $\chi(G)$, the chromatic number of the characteristic graph G. In terms of bits, $\log_2 \chi(G)$.
- A and B agree in advance on a coloring of G, namely P_A

Main theorem

Single instance

- The smallest number of possible messages the informant must transmit for a single instance is $\chi(G)$, the chromatic number of the characteristic graph G. In terms of bits, $\log_2 \chi(G)$.
- A and B agree in advance on a coloring of G, namely P_A
- Given x, A sends its color, $P_A(x)$.

Main theorem

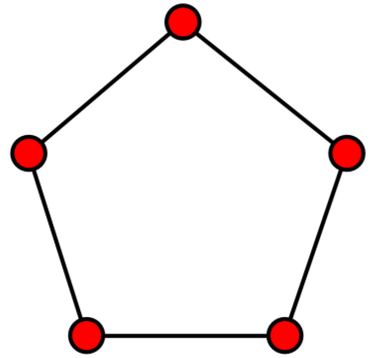
Single instance

- The smallest number of possible messages the informant must transmit for a single instance is $\chi(G)$, the chromatic number of the characteristic graph G. In terms of bits, $\log_2 \chi(G)$.
- A and B agree in advance on a coloring of G, namely P_A
- Given x, A sends its color, $P_A(x)$.
- B, knowing his input y, can determine x because there is a single x with this color that is jointly possible with y, i.e., (x, y) is in the source input graph $\mathcal{I}$.

Examples

Cycle and Kneser graphs

- $\chi(C_5) = 3$, and 2 bits are sufficient (and needed), instead of 3 bits needed for 5 elements

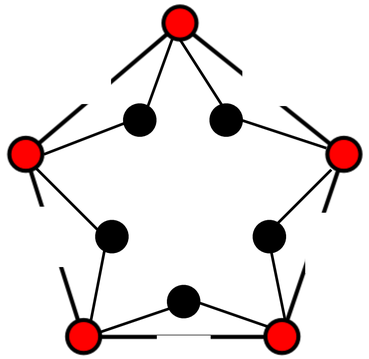


Cycle C_5

Examples

Cycle

- $\chi(C_5) = 3$, and 2 bits are sufficient (and needed), instead of 3 bits needed for 5 elements



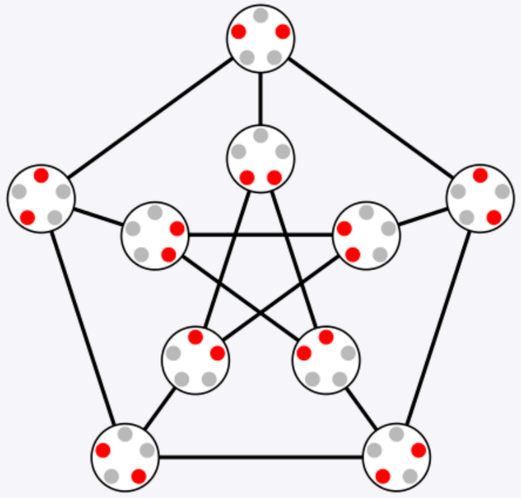
Cycle C_5

Input graph

Examples

Kneser graphs

- $K(u, t)$ consists of all t-element subsets of $\{1,\ldots, u\}$, two vertices are connected iff they are disjoint
- chromatic number $\chi(K(u, t)) = u - 2t + 2$
- $\chi(K(5,2)) = 3$



The Kneser graph $K(5, 2)$, isomorphic to the Petersen graph

Multiple instances

Main theorem

Multiple instances

Main theorem

- A knows $x_1, \ldots, x_n$ and B knows $y_1, \ldots, y_n$

Multiple instances

Main theorem

- A knows $x_1, \ldots, x_n$ and B knows $y_1, \ldots, y_n$
- The *and-power n characteristic graph* $G_\wedge^n$ has vertices $x_1, \ldots, x_n$, and two are in an edge iff for each entry x_i, x_i' , there is a y, such x, y and x', y are possible, ie edges of the input graph $\mathcal{I}$

Multiple instances

Main theorem

- A knows $x_1, \ldots, x_n$ and B knows $y_1, \ldots, y_n$
- The *and-power n characteristic graph* $G_\wedge^n$ has vertices $x_1, \ldots, x_n$, and two are in an edge iff for each entry x_i, x_i' , there is a y, such x, y and x', y are possible, ie edges of the input graph $\mathcal{F}$
- *Theorem*: The smallest number of possible messages A must transmit for a n instances is $\chi(G_\wedge^n)$, the chromatic number of this characteristic graph. In terms of bits, $\sigma_n = \log_2 \chi(G_\wedge^n)$.

Multiple instances

Saving on multiple instances

Multiple instances

Saving on multiple instances

Multiple instances

Saving on multiple instances

- Clearly encoding *two instances* should require at most twice as many bits as needed for one $\sigma_2 \leq 2\sigma_1$.

Multiple instances

Saving on multiple instances

- Clearly encoding *two instances* should require at most twice as many bits as needed for one $\sigma_2 \leq 2\sigma_1$.
- and since different instances of the source are completely independent of each other, it is not intuitively clear that it is possible $\sigma_2 < 2\sigma_1$. Indeed, not so for an uncorrelated source, where the graphs are complete

Examples

Cycle and Kneser multiple instances

Examples

Cycle and Kneser multiple instances

- Witsenhausen showed that for the pentagon, $\sigma_2 < 2\sigma_1$!

Examples

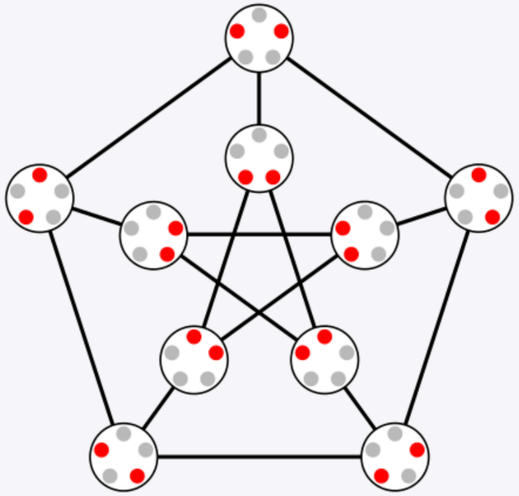
Cycle and Kneser multiple instances

- Witsenhausen showed that for the pentagon, $\sigma_2 < 2\sigma_1$!
- $\sigma_1 = \log_2 3 \approx 1.58$ while for 2 or more instances, $\log_2 5/2 \approx 1.16$ bits per instance

Examples

Cycle and Kneser multiple instances

- Witsenhausen showed that for the pentagon, $\sigma_2 < 2\sigma_1$!
- $\sigma_1 = \log_2 3 \approx 1.58$ while for 2 or more instances, $\log_2 5/2 \approx 1.16$ bits per instance
- For the Knesser graph, $\chi(K(u,t)) = u - 2t + 2$, n sports, u players and two teams of t players each

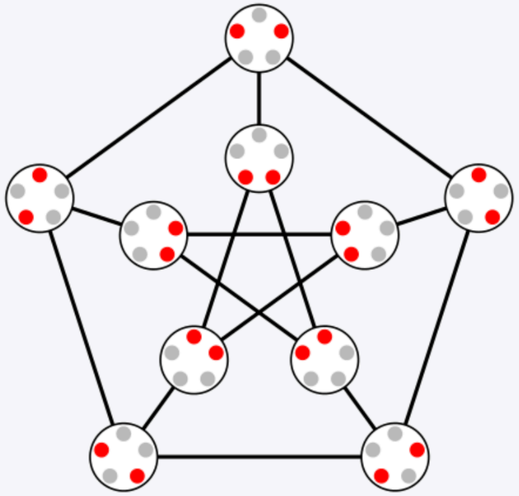


The Kneser graph $K(5, 2)$, isomorphic to the Petersen graph

Examples

Cycle and Kneser multiple instances

- Witsenhausen showed that for the pentagon, $\sigma_2 < 2\sigma_1$!
- $\sigma_1 = \log_2 3 \approx 1.58$ while for 2 or more instances, $\log_2 5/2 \approx 1.16$ bits per instance
- For the Knesser graph, $\chi(K(u,t)) = u - 2t + 2$, n sports, u players and two teams of t players each
- Asymptotically $\sigma_\infty = \log u/t$ since the chromatic number is asymptotically u/t (Alon, Orlitsky IEEE TIT'95)

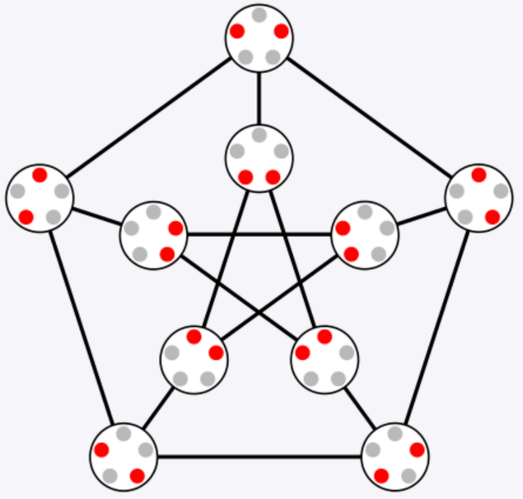


The Kneser graph $K(5, 2)$, isomorphic to the Petersen graph

Examples

Cycle and Kneser multiple instances

- Witsenhausen showed that for the pentagon, $\sigma_2 < 2\sigma_1$!
- $\sigma_1 = \log_2 3 \approx 1.58$ while for 2 or more instances, $\log_2 5/2 \approx 1.16$ bits per instance
- For the Knesser graph, $\chi(K(u,t)) = u - 2t + 2$, n sports, u players and two teams of t players each
- Asymptotically $\sigma_\infty = \log u/t$ since the chromatic number is asymptotically u/t (Alon, Orlitsky IEEE TIT'95)
- Let $u = (2+\epsilon)t$. Then for $n = 1$ need $\log_2(\epsilon t + 2)$ bits, while as $n \to \infty$ need $\log_2(2+\epsilon) \leq 1 + \epsilon$

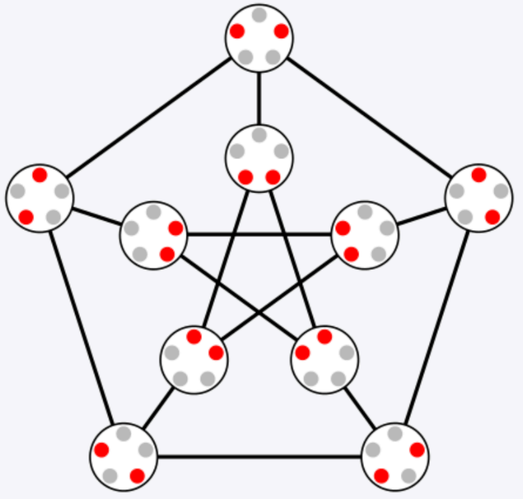


The Kneser graph $K(5, 2)$, isomorphic to the Petersen graph

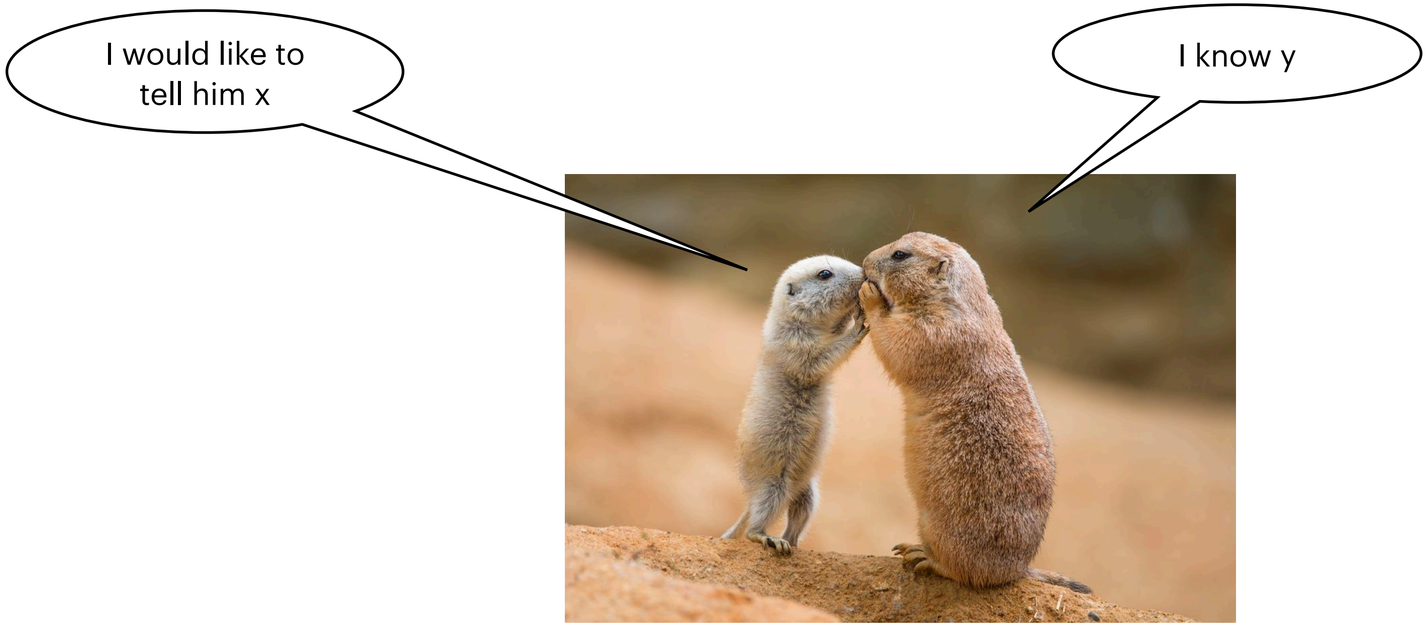
Examples

Cycle and Kneser multiple instances

- Witsenhausen showed that for the pentagon, $\sigma_2 < 2\sigma_1$!
- $\sigma_1 = \log_2 3 \approx 1.58$ while for 2 or more instances, $\log_2 5/2 \approx 1.16$ bits per instance
- For the Knesser graph, $\chi(K(u,t)) = u - 2t + 2$, n sports, u players and two teams of t players each
- Asymptotically $\sigma_\infty = \log u/t$ since the chromatic number is asymptotically u/t (Alon, Orlitsky IEEE TIT'95)
- Let $u = (2+\epsilon)t$. Then for $n = 1$ need $\log_2(\epsilon t + 2)$ bits, while as $n \to \infty$ need $\log_2(2+\epsilon) \leq 1 + \epsilon$
- Keeping ϵ small and increasing t, we see that a single sport requires arbitrarily many bits while many sports require roughly one bit per instance!



The Kneser graph $K(5, 2)$, isomorphic to the Petersen graph



Interaction

https://earth-chronicles.com/science/the-most-striking-ways-of-communicating-animals.html

Interactive communication

Section XII

Interactive communication

Section XII

- for some sources, interaction can reduce transmission to the logarithm of the one-way number of bits, and

Interactive communication

Section XII

- for some sources, interaction can reduce transmission to the logarithm of the one-way number of bits, and
- for a large class of sources, interaction can reduce transmission to about the same number of bits required when A knows y in advance.

Interactive communication

Section XII

- for some sources, interaction can reduce transmission to the logarithm of the one-way number of bits, and
- for a large class of sources, interaction can reduce transmission to about the same number of bits required when A knows y in advance.
- For communication of *multiple instances*, interaction can reduce transmission by an arbitrary amount, and can always achieve the number of bits needed when A knows y in advance.

Interactive communication

Section XII

Interactive communication

Section XII

- Model: A and B alternate in transmitting messages, determined by an agreed-upon, deterministic protocol.

Interactive communication

Section XII

- Model: A and B alternate in transmitting messages, determined by an agreed-upon, deterministic protocol.
- The input is a pair (x, y), Alice wants to transmit x to Bob, who knows y

Interactive communication

Section XII

- Model: A and B alternate in transmitting messages, determined by an agreed-upon, deterministic protocol.
- The input is a pair (x, y), Alice wants to transmit x to Bob, who knows y
- C_1 = number of bits required in the worst case when B cannot transmit to A, while

Interactive communication

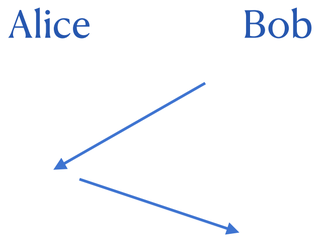
Section XII

- Model: A and B alternate in transmitting messages, determined by an agreed-upon, deterministic protocol.
- The input is a pair (x, y), Alice wants to transmit x to Bob, who knows y
- C_1 = number of bits required in the worst case when B cannot transmit to A, while
- C_2 = number of bits when B transmits a message reflecting y, then A responds with a message from which B must infer x

Interactive communication

Section XII

- Model: A and B alternate in transmitting messages, determined by an agreed-upon, deterministic protocol.
- The input is a pair (x, y), Alice wants to transmit x to Bob, who knows y
- C_1 = number of bits required in the worst case when B cannot transmit to A, while
- C_2 = number of bits when B transmits a message reflecting y, then A responds with a message from which B must infer x
- C_∞ = number of bits of a protocol with any number of message exchanges



One message requires exponentially more bits than the minimum necessary

League problem

One message requires exponentially more bits than the minimum necessary

League problem

- Sports league has t teams.

One message requires exponentially more bits than the minimum necessary

League problem

- Sports league has t teams.
- B knows two teams that played in a game, and

One message requires exponentially more bits than the minimum necessary

League problem

- Sports league has t teams.
- B knows two teams that played in a game, and
- A knows the team that won the game, but not against whom.

One message requires exponentially more bits than the minimum necessary

League problem

- Sports league has t teams.
- B knows two teams that played in a game, and
- A knows the team that won the game, but not against whom.
- They communicate in order for B to learn the winning team.

Alice	Bob
$x =$ Mexico	$y =$ Mexico vs France

One message requires exponentially more bits than the minimum necessary

League problem

One message requires exponentially more bits than the minimum necessary

League problem

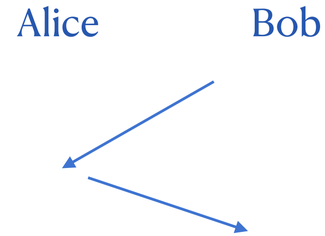
- *One message.* $C_1 = \lceil \log_2 t \rceil$ bits are necessary: Different messages must be sent when one wins than when another wins

One message requires exponentially more bits than the minimum necessary

League problem

- *One message.* $C_1 = \lceil \log_2 t \rceil$ bits are necessary: Different messages must be sent when one wins than when another wins

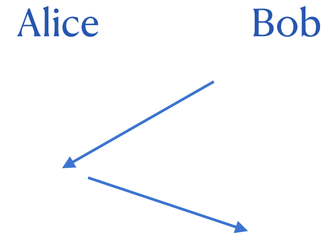
- *Two messages.* B considers the binary representations of the two teams that played and transmits $\lceil \log_2 \log_2 t \rceil$ bits describing the location of the first bit where they differ



One message requires exponentially more bits than the minimum necessary

League problem

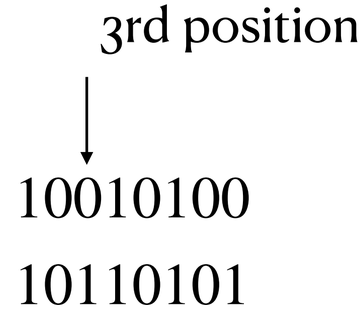
- *One message.* $C_1 = \lceil \log_2 t \rceil$ bits are necessary: Different messages must be sent when one wins than when another wins
- *Two messages.* B considers the binary representations of the two teams that played and transmits $\lceil \log_2 \log_2 t \rceil$ bits describing the location of the first bit where they differ
- A responds by transmitting a single bit describing the bit value of the winning team in that location , $C_2 = \lceil \log_2 \log_2 t \rceil + 1$



One message requires exponentially more bits than the minimum necessary

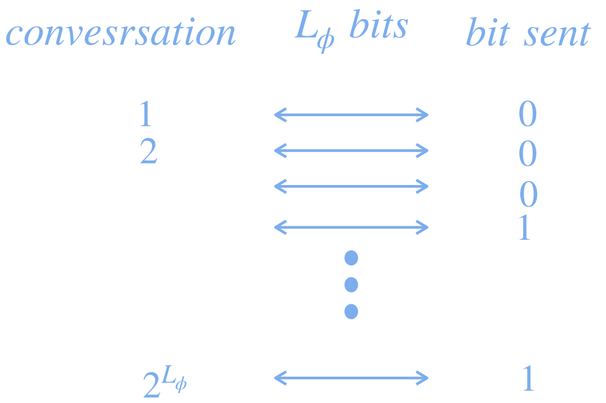
League problem

- *One message.* $C_1 = \lceil \log_2 t \rceil$ bits are necessary: Different messages must be sent when one wins than when another wins
- *Two messages.* B considers the binary representations of the two teams that played and transmits $\lceil \log_2 \log_2 t \rceil$ bits describing the location of the first bit where they differ
- A responds by transmitting a single bit describing the bit value of the winning team in that location , $C_2 = \lceil \log_2 \log_2 t \rceil + 1$



The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

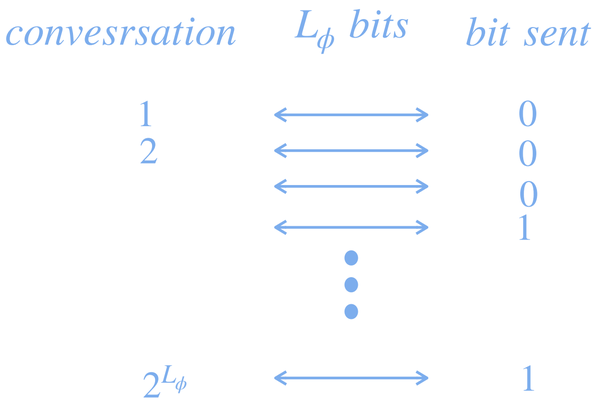


In the paper $\beta \preccurlyeq \alpha$ if string β is a prefix of α but we may assume for simplicity that all conversations are of the same length

The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- *Theorem*: the number of bits of any number of interactions $C_\infty \geq \lceil \log_2 C_1 \rceil$ sending only once

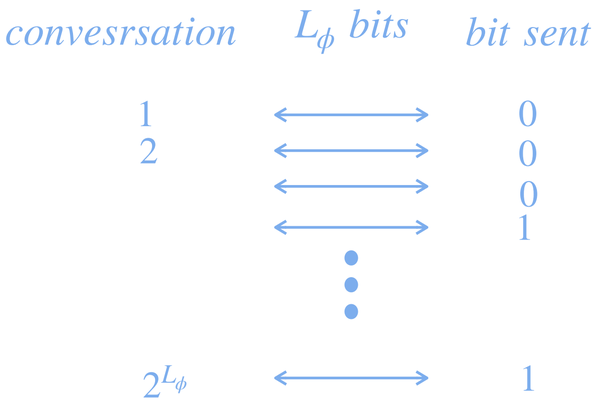


In the paper $\beta \preccurlyeq \alpha$ if string β is a prefix of α but we may assume for simplicity that all conversations are of the same length

The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- *Theorem*: the number of bits of any number of interactions $C_\infty \geq \lceil \log_2 C_1 \rceil$ sending only once
- *Proof*: Simulate a protocol for C_∞ by sending just one message

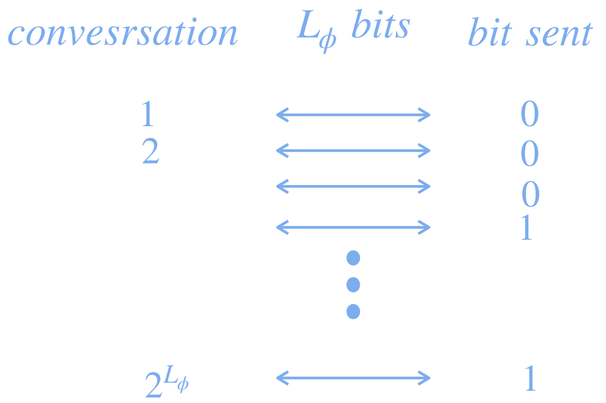


In the paper $\beta \preccurlyeq \alpha$ if string β is a prefix of α but we may assume for simplicity that all conversations are of the same length

The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- *Theorem*: the number of bits of any number of interactions $C_\infty \geq \lceil \log_2 C_1 \rceil$ sending only once
- *Proof*: Simulate a protocol for C_∞ by sending just one message
- Given a protocol ϕ with complexity L_ϕ we construct a one way protocol P_A with complexity 2^{L_ϕ} as follows.

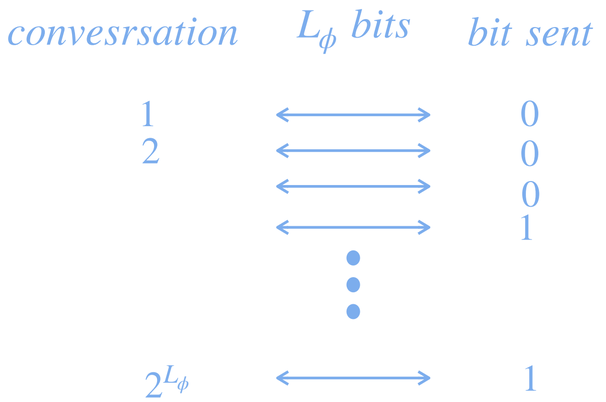


In the paper $\beta \preccurlyeq \alpha$ if string β is a prefix of α but we may assume for simplicity that all conversations are of the same length

The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- *Theorem*: the number of bits of any number of interactions $C_\infty \geq \lceil \log_2 C_1 \rceil$ sending only once
- *Proof*: Simulate a protocol for C_∞ by sending just one message
- Given a protocol ϕ with complexity L_ϕ we construct a one way protocol P_A with complexity 2^{L_ϕ} as follows.
- Given x, Alice considers all possible 2^{L_ϕ} conversations of L_ϕ bits.

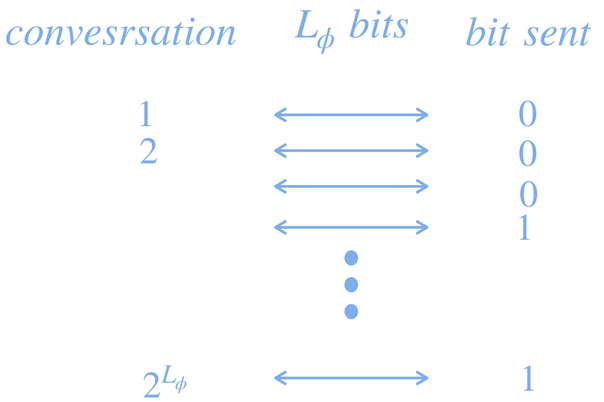


In the paper $\beta \preccurlyeq \alpha$ if string β is a prefix of α but we may assume for simplicity that all conversations are of the same length

The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- *Theorem*: the number of bits of any number of interactions $C_\infty \geq \lceil \log_2 C_1 \rceil$ sending only once
- *Proof*: Simulate a protocol for C_∞ by sending just one message
- Given a protocol ϕ with complexity L_ϕ we construct a one way protocol P_A with complexity 2^{L_ϕ} as follows.
- Given x, Alice considers all possible 2^{L_ϕ} conversations of L_ϕ bits.
- For each such α, A transmites bit $f_x(\alpha) = 1$ if $\sigma(x, y) \preccurlyeq \alpha$ for some y compatible with x, and bit $f_x(\alpha) = 0$ otherwise

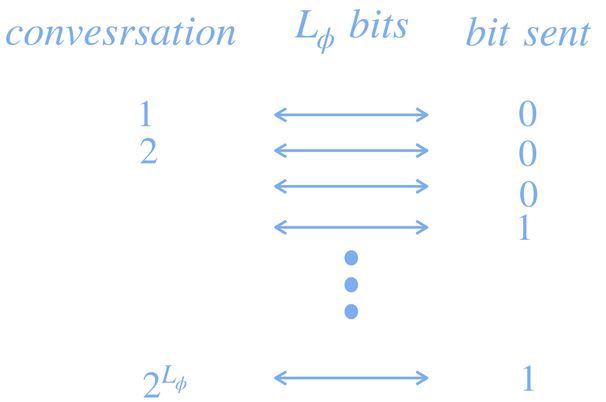


In the paper $\beta \preccurlyeq \alpha$ if string β is a prefix of α but we may assume for simplicity that all conversations are of the same length

The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- *Theorem*: the number of bits of any number of interactions $C_\infty \geq \lceil \log_2 C_1 \rceil$ sending only once
- *Proof*: Simulate a protocol for C_∞ by sending just one message
- Given a protocol ϕ with complexity L_ϕ we construct a one way protocol P_A with complexity 2^{L_ϕ} as follows.
- Given x, Alice considers all possible 2^{L_ϕ} conversations of L_ϕ bits.
- For each such α, A transmites bit $f_x(\alpha) = 1$ if $\sigma(x, y) \preccurlyeq \alpha$ for some y compatible with x, and bit $f_x(\alpha) = 0$ otherwise
- Then B decodes: simply finds an L_ϕ bit sequence α for which $f_x(\alpha) = 1$. He then finds and x compatible with his y, for which the conversation with $\sigma(x, y) \preccurlyeq \alpha$ and decides x.

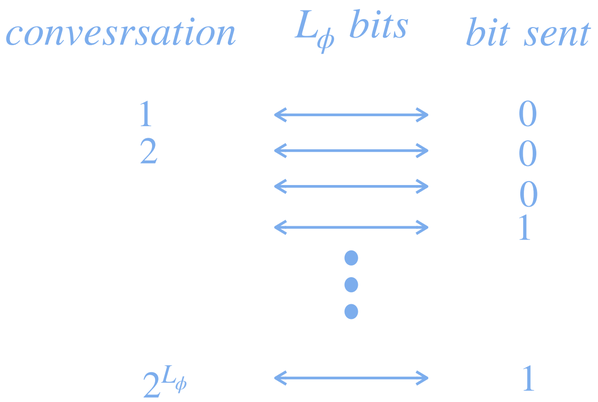


In the paper $\beta \preccurlyeq \alpha$ if string β is a prefix of α but we may assume for simplicity that all conversations are of the same length

The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- *Theorem*: the number of bits of any number of interactions $C_\infty \geq \lceil \log_2 C_1 \rceil$ sending only once
- *Proof*: Simulate a protocol for C_∞ by sending just one message
- Given a protocol ϕ with complexity L_ϕ we construct a one way protocol P_A with complexity 2^{L_ϕ} as follows.
- Given x, Alice considers all possible 2^{L_ϕ} conversations of L_ϕ bits.
- For each such α, A transmites bit $f_x(\alpha) = 1$ if $\sigma(x, y) \preccurlyeq \alpha$ for some y compatible with x, and bit $f_x(\alpha) = 0$ otherwise
- Then B decodes: simply finds an L_ϕ bit sequence α for which $f_x(\alpha) = 1$. He then finds and x compatible with his y, for which the conversation with $\sigma(x, y) \preccurlyeq \alpha$ and decides x.
- Argument (1) there is an L_ϕ-bit seq α for which $f_x(\alpha) = 1$ and the conversation $\preccurlyeq \alpha$ and (2) there is no one more x' for which a conversation $\sigma(x', y) \preccurlyeq \alpha$



In the paper $\beta \preccurlyeq \alpha$ if string β is a prefix of α but we may assume for simplicity that all conversations are of the same length

The largest possible discrepancy

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- it is possible to show that the largest possible discrepancy is 1 more,
- *Theorem*: $C_\infty \geq \lceil \log_2 C_1 \rceil + 1$
- *Proof*: uses a recursive argument, basically on the sizes of the cliques after comunication reduced at most in 1/2

Limits of interaction

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- *Theorem*: just two messages always suffice to reduce communication to almost the minimum: $C_2 \leq 4C_\infty + 3$ for all sources

Limits of interaction

ORLITSKY: WORST-CASE INTERACTIVE COMMUNICATION I, IV. THE LIMITS OF INTERACTION

- Is there an m such that an m-message protocol is asymptotically optimum?

Balanced and correlated files

Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

https://earth-chronicles.com/science/the-most-striking-ways-of-communicating-animals.html

Balanced and correlated files

Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

- For general sources, one way communication may require exponential more bits than the minimum necessary (over arbitrary many exchanges), yet, for balanced sources, *one way requires at most twice the minimum* $C_1 \leq 2C_\infty + 1$

https://earth-chronicles.com/science/the-most-striking-ways-of-communicating-animals.html

Balanced and correlated files

Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

- For general sources, one way communication may require exponential more bits than the minimum necessary (over arbitrary many exchanges), yet, for balanced sources, *one way requires at most twice the minimum* $C_1 \leq 2C_\infty + 1$
- This bound is almost tight

https://earth-chronicles.com/science/the-most-striking-ways-of-communicating-animals.html

Balanced and correlated files

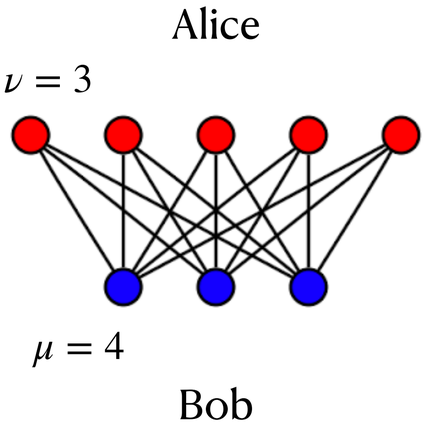
Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

- For general sources, one way communication may require exponential more bits than the minimum necessary (over arbitrary many exchanges), yet, for balanced sources, *one way requires at most twice the minimum* $C_1 \leq 2C_\infty + 1$
- This bound is almost tight
- 3 rounds is asymptotically optimal

https://earth-chronicles.com/science/the-most-striking-ways-of-communicating-animals.html

Balanced Sources

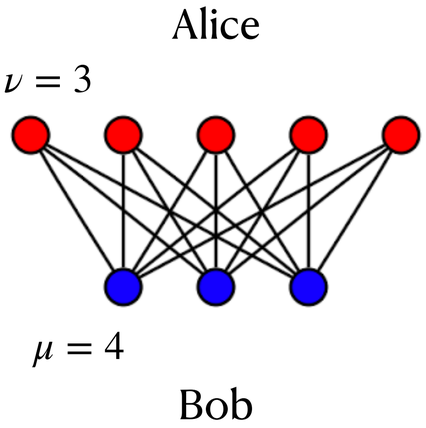
Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993



Balanced Sources

Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

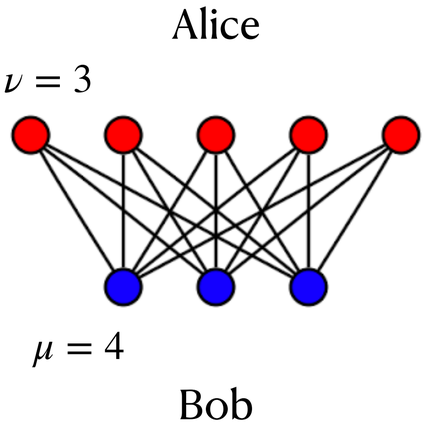
- A's ambiguity ν is the max over all x, of the number of possible inputs for B, ie the max degree of an A-node in the input graph $\mathcal{I}$



Balanced Sources

Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

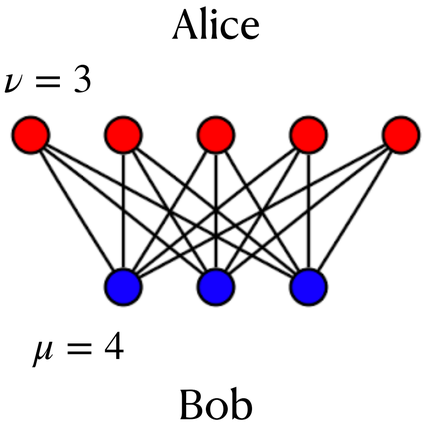
- A's ambiguity ν is the max over all x, of the number of possible inputs for B, ie the max degree of an A-node in the input graph $\mathcal{I}$
- B's ambiguity μ is the max over all y, of the number of possible inputs for A, ie the max degree of a B-node in the input graph $\mathcal{I}$



Balanced Sources

Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

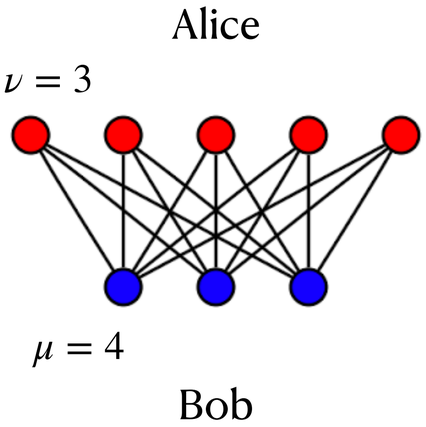
- A's ambiguity ν is the max over all x, of the number of possible inputs for B, ie the max degree of an A-node in the input graph $\mathcal{I}$
- B's ambiguity μ is the max over all y, of the number of possible inputs for A, ie the max degree of a B-node in the input graph $\mathcal{I}$



Balanced Sources

Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

- A's ambiguity ν is the max over all x, of the number of possible inputs for B, ie the max degree of an A-node in the input graph $\mathcal{I}$
- B's ambiguity μ is the max over all y, of the number of possible inputs for A, ie the max degree of a B-node in the input graph $\mathcal{I}$

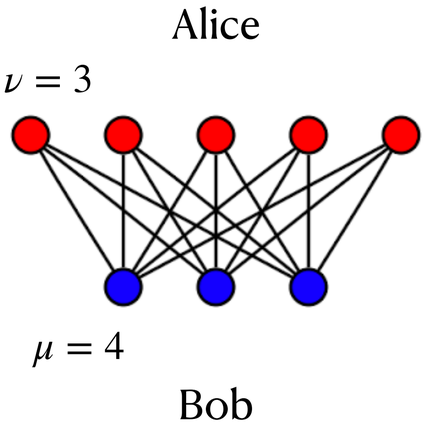


- In the leagues problem: these numbers are $\nu = t - 1, \mu = 2$

Balanced Sources

Orlitzky, Interactive communication of balanced distributions, SIAM DM 1993

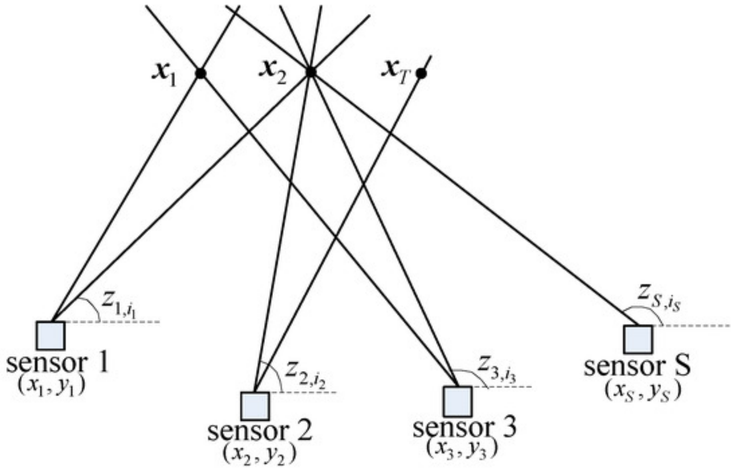
- A's ambiguity ν is the max over all x, of the number of possible inputs for B, ie the max degree of an A-node in the input graph $\mathcal{I}$
- B's ambiguity μ is the max over all y, of the number of possible inputs for A, ie the max degree of a B-node in the input graph $\mathcal{I}$



- In the leagues problem: these numbers are $\nu = t - 1, \mu = 2$
- *Balanced sources* have both numbers equal $\mu = \nu$

Balanced sources

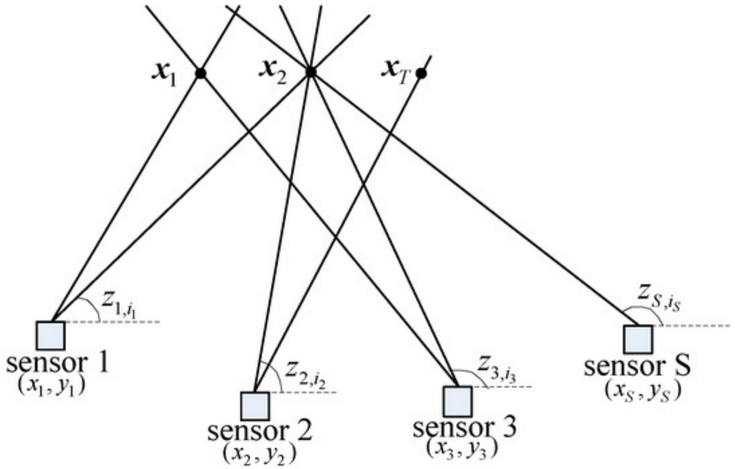
examples: measurements



Balanced sources

examples: measurements

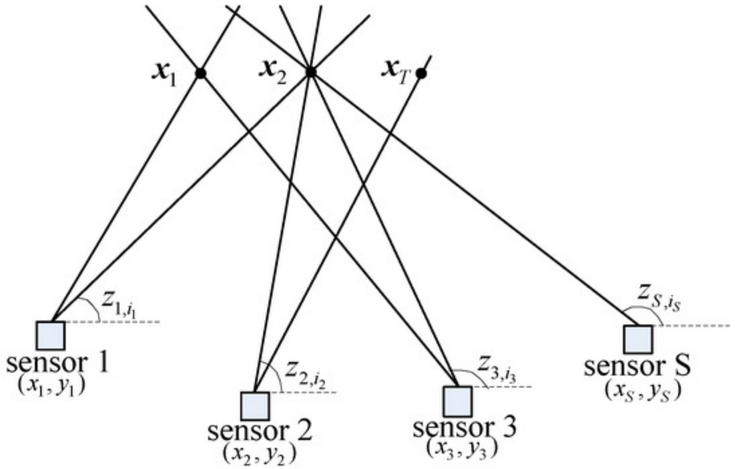
1. inaccurate measurements of the same quantity, are integers within a bounded absolute difference from each other.



Balanced sources

examples: measurements

1. inaccurate measurements of the same quantity, are integers within a bounded absolute difference from each other.
2. are obtained from a faulty memory, n-bit strings within a bounded Hamming distance from each other.



Balanced sources

examples: correlated files

Balanced sources

examples: correlated files

1. *edit distance* between two binary strings x and y is the minimum number of deletions and insertions to x needed to derive y.

Balanced sources

examples: correlated files

1. *edit distance* between two binary strings x and y is the minimum number of deletions and insertions to x needed to derive y.

- edit distance between 01010 and 10101 is 2

Balanced sources

examples: correlated files

1. *edit distance* between two binary strings x and y is the minimum number of deletions and insertions to x needed to derive y.

- edit distance between 01010 and 10101 is 2
- *correlated-files problem*: x, y are binary strings within a *small* edit distance from each other. A knows x while B knows y and wants to learn x.

Balanced sources

examples: correlated files

1. *edit distance* between two binary strings x and y is the minimum number of deletions and insertions to x needed to derive y.

- edit distance between 01010 and 10101 is 2
- *correlated-files problem*: x, y are binary strings within a *small* edit distance from each other. A knows x while B knows y and wants to learn x.
- Example: A and B write a joint book and each updates his version individually or x and y are different versions of the same program or file

Balanced sources

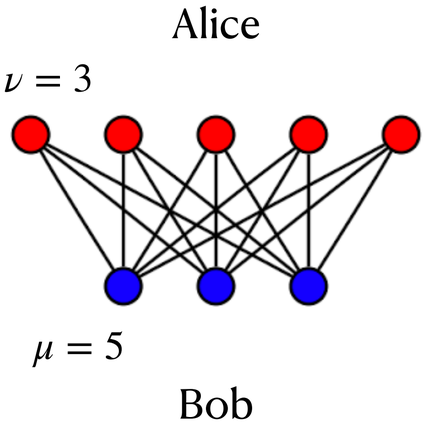
examples: correlated files

1. *edit distance* between two binary strings x and y is the minimum number of deletions and insertions to x needed to derive y.

- edit distance between 01010 and 10101 is 2
- *correlated-files problem*: x, y are binary strings within a *small* edit distance from each other. A knows x while B knows y and wants to learn x.
- Example: A and B write a joint book and each updates his version individually or x and y are different versions of the same program or file
- We are looking for a way to communicate x to B without transmitting all of it.

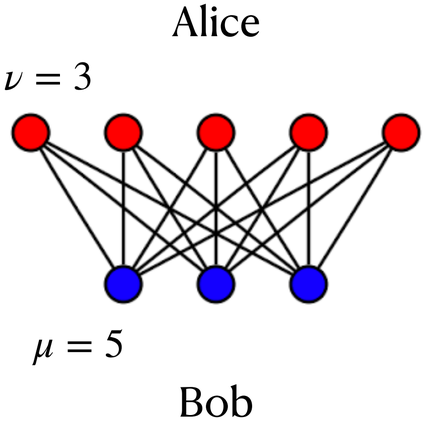
Results

Balanced sources



Results

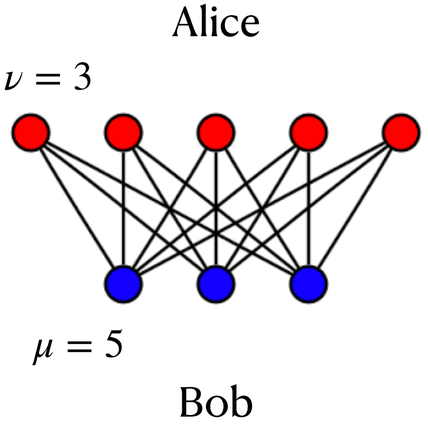
Balanced sources



- For balanced sources, there is almost no increase in communication due to A not knowing y !

Results

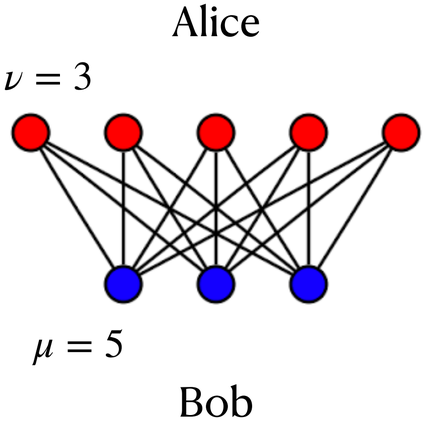
Balanced sources



- For balanced sources, there is almost no increase in communication due to A not knowing y !
- *Lemma 1*: for all sources $C_1 \leq \log_2 \mu + \log_2 \nu + 1$

Results

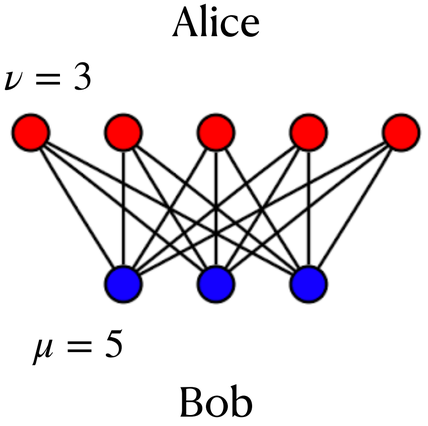
Balanced sources



- For balanced sources, there is almost no increase in communication due to A not knowing y !
- *Lemma 1*: for all sources $C_1 \leq \log_2 \mu + \log_2 \nu + 1$
- proof: Each vertex in the characteristic graph belongs to at most ν edges, and each edge contains at most μ vertices. Thus, $\chi(G) \leq \nu(\mu - 1) + 1 \leq \mu\nu$

Results

Balanced sources



- For balanced sources, there is almost no increase in communication due to A not knowing y !
- *Lemma 1*: for all sources $C_1 \leq \log_2 \mu + \log_2 \nu + 1$
- proof: Each vertex in the characteristic graph belongs to at most ν edges, and each edge contains at most μ vertices. Thus, $\chi(G) \leq \nu(\mu - 1) + 1 \leq \mu\nu$
- *Corollary 1*: $C_1 \leq 2\log_2 \mu + 1 \leq 2C_\infty + 1$ for balanced sources.

Three messages is asymptotically optimal

using perfect hash functions and Lovasz local lemma

- This is proved in Sec 3
- Section 5: correlated files: x, y are binary strings within edit distance of α from each other.
- It shows that the previos results can be used to obtain efficient 3 message protocols.

Rsynch

H. Yan; U. Irmak; T. Suel,
Algorithms for Low-Latency Remote File Synchronization, IEEE INFOCOM 2008

- The remote file synchronization problem is how to update an outdated version of a file located on one machine to the current version located on another machine with a minimal amount of network communication.
- It arises in many scenarios including web site mirroring, file system backup and replication, or web access over slow links.
- A widely used open-source tool called **rsync** uses a single round of messages to solve this problem.

In this paper, they study single-round synchronization techniques that achieve savings in bandwidth consumption while preserving many of the advantages of the rsync approach.

END

of Part I

Part II

Wait-free perspective

Communication Complexity of Wait-Free Computability in Dynamic Networks

Carole Delporte-Gallet, Hugues Fauconnier
University Paris 7
and
Sergio Rajsbaum
UNAM, Mexico

SIROCCO 2020

Distributed computability

Tasks

- The computational power of a distributed system depends on its communication, process relative speeds, and failure assumptions.

Distributed computability

Tasks

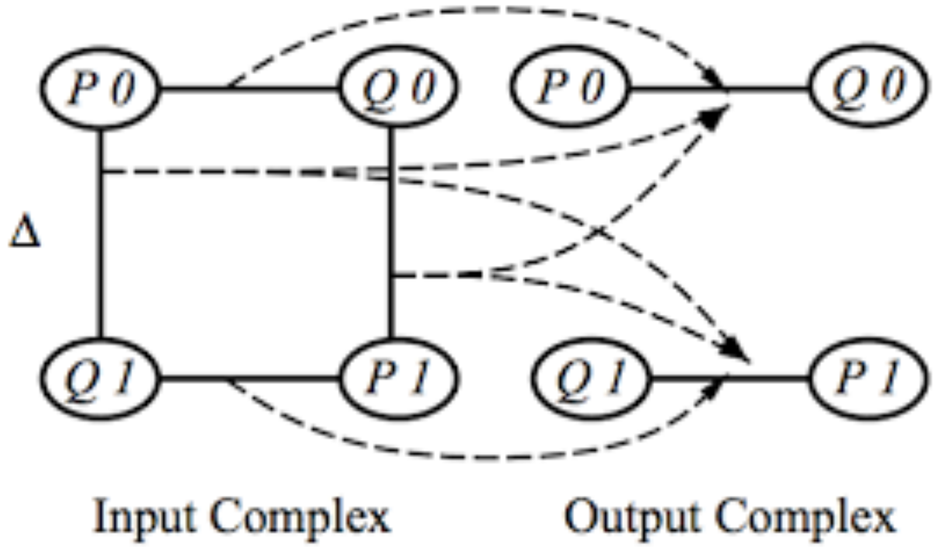
- The computational power of a distributed system depends on its communication, process relative speeds, and failure assumptions.
- A model's computational power is typically studied with respect to *tasks*, such as consensus, etc.
- Specified by an input/output relation Δ

Two process
binary consensus

Distributed computability

Tasks

- The computational power of a distributed system depends on its communication, process relative speeds, and failure assumptions.
- A model's computational power is typically studied with respect to *tasks*, such as consensus, etc.
- Specified by an input/output relation Δ



Two process
binary consensus

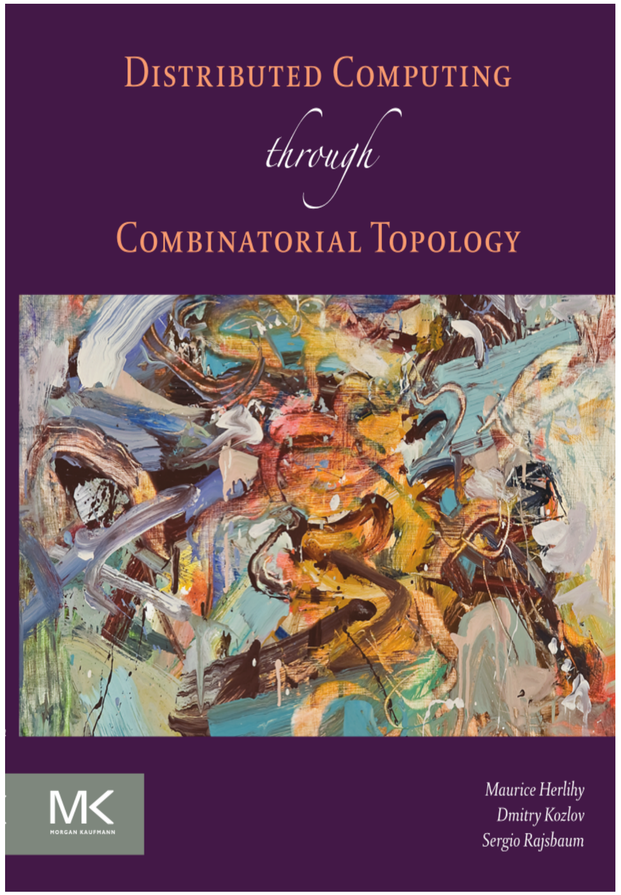
Distributed computability

Two milestones

- A characterization of the tasks that are solvable in an asynchronous message passing system where at most one process may crash [BiranMoranZaks'90]
- A wait-free characterization of the tasks, solvable in an asynchronous read/write shared memory system where any number of processes may crash [HerlihyShavit'99]

Nature of the characterizations

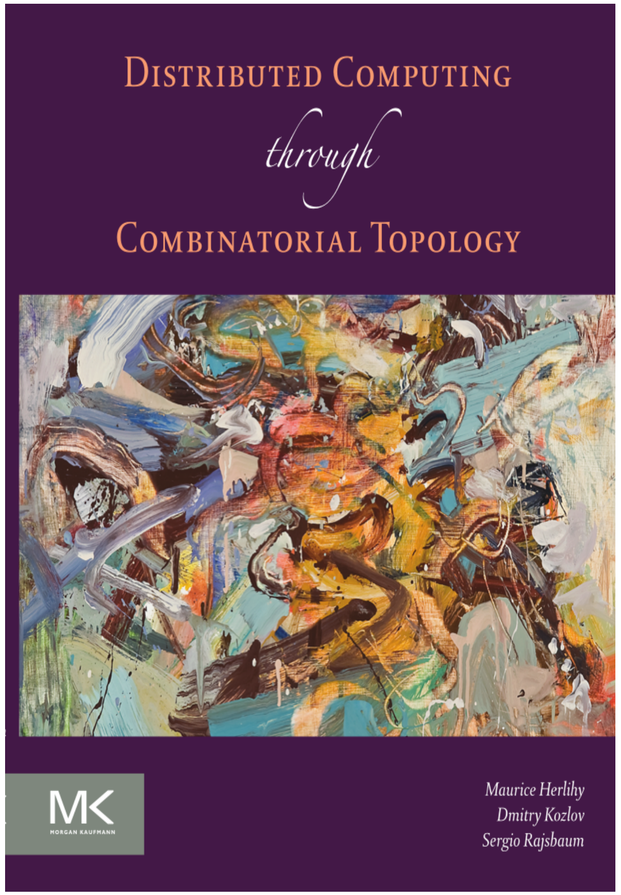
Task solvability and complexity



Nature of the characterizations

Task solvability and complexity

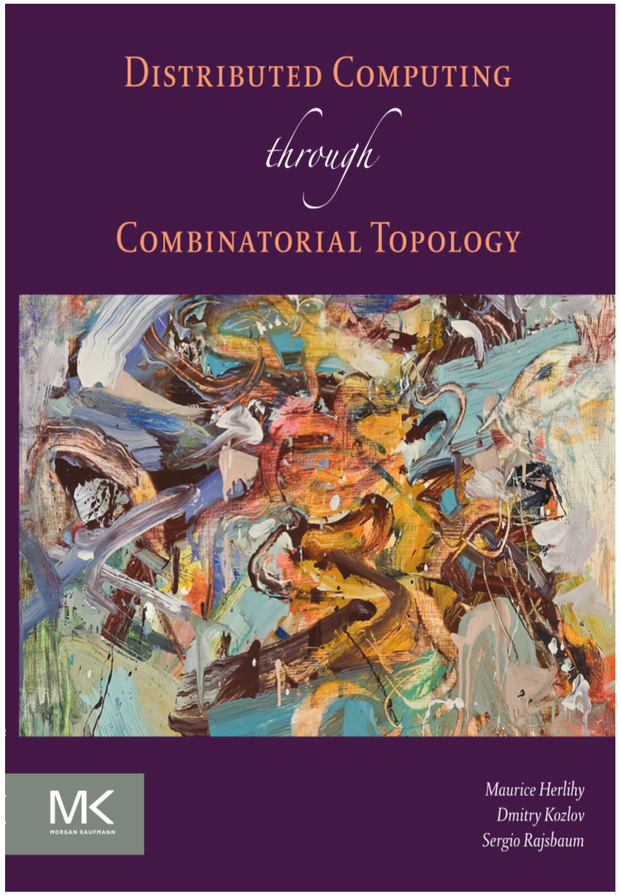
- These papers discovered the intimate relation with *topology*.



Nature of the characterizations

Task solvability and complexity

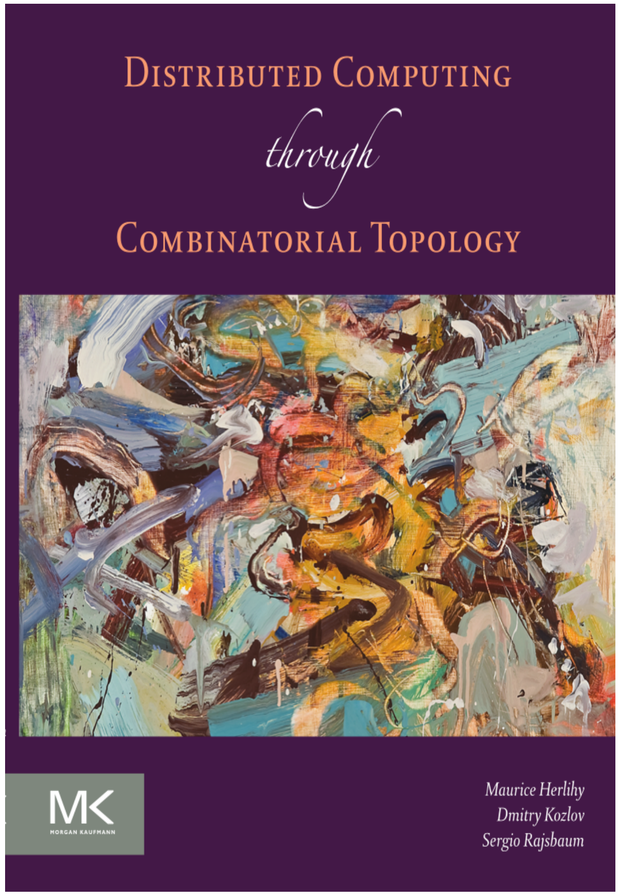
- These papers discovered the intimate relation with *topology*.
- Task solvability characterizations have been described for many models:



Nature of the characterizations

Task solvability and complexity

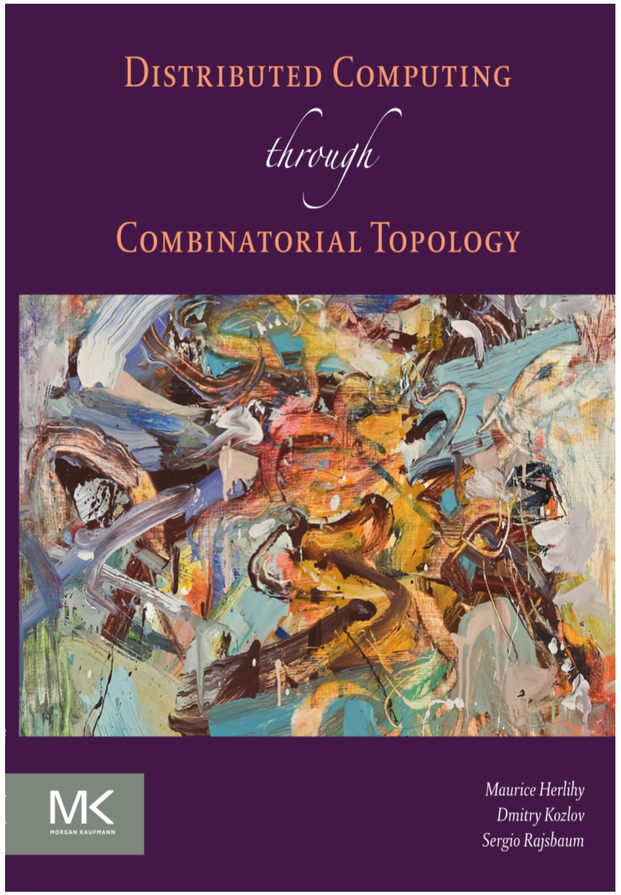
- These papers discovered the intimate relation with *topology*.
- Task solvability characterizations have been described for many models:
  - share memory, message passing, mobile robots, etc.



Nature of the characterizations

Task solvability and complexity

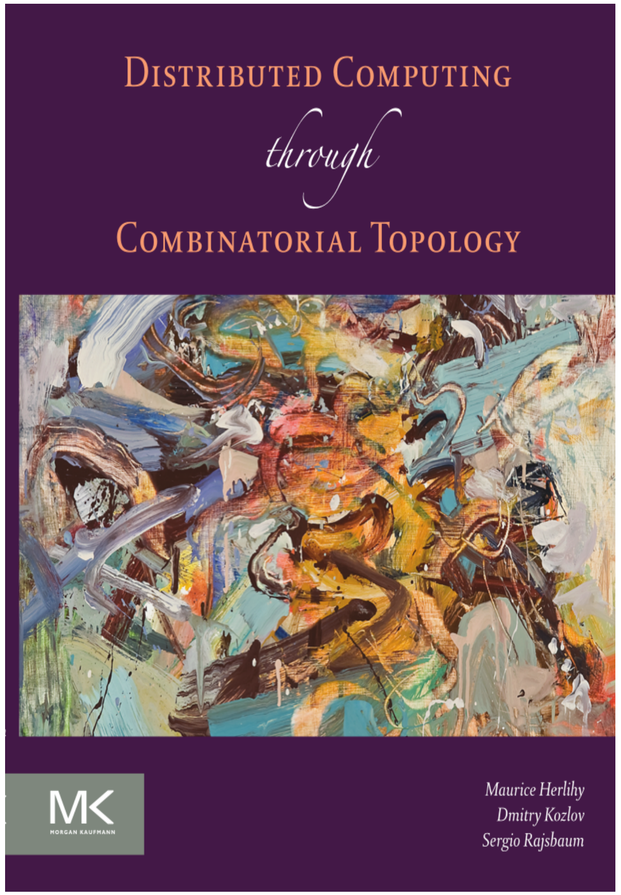
- These papers discovered the intimate relation with *topology*.
- Task solvability characterizations have been described for many models:
  - share memory, message passing, mobile robots, etc.
  - crash and Byzantine failures, both dependent and independent



Nature of the characterizations

Task solvability and complexity

- These papers discovered the intimate relation with *topology*.
- Task solvability characterizations have been described for many models:
  - share memory, message passing, mobile robots, etc.
  - crash and Byzantine failures, both dependent and independent
  - Determine solvability for many tasks: consensus, set agreement, renaming, symmetry breaking, equality negation, etc.



Full-information protocols

Motivation

Full-information protocols

Motivation

- All the task solvability characterizations considered in the literature assume a full-information protocol

Full-information protocols

Motivation

- All the task solvability characterizations considered in the literature assume a full-information protocol
- a process keeps in its state everything it knows, and each time it communicates with other processes, it sends its entire state

Full-information protocols

Motivation

- All the task solvability characterizations considered in the literature assume a full-information protocol
- a process keeps in its state everything it knows, and each time it communicates with other processes, it sends its entire state
- the size of the messages (or the values written to the shared-memory) grows with the number of rounds:

Full-information protocols

Motivation

- All the task solvability characterizations considered in the literature assume a full-information protocol
- a process keeps in its state everything it knows, and each time it communicates with other processes, it sends its entire state
- the size of the messages (or the values written to the shared-memory) grows with the number of rounds:
  - rounds become slower and slower, to implement the necessary information exchange

Full-information protocols

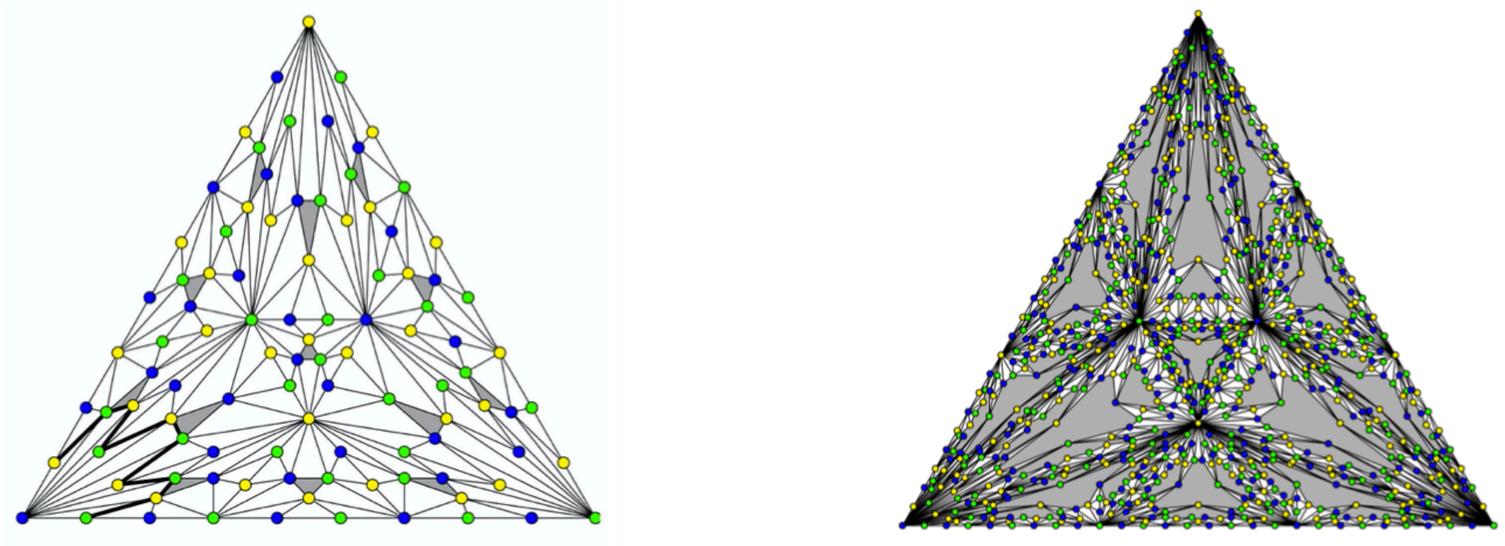
Motivation

- All the task solvability characterizations considered in the literature assume a full-information protocol
- a process keeps in its state everything it knows, and each time it communicates with other processes, it sends its entire state
- the size of the messages (or the values written to the shared-memory) grows with the number of rounds:
  - rounds become slower and slower, to implement the necessary information exchange
  - Notice the number of rounds needed to solve a task can grow so fast, that it is undecidable if a task has a wait-free protocol, even for three processes

Full-information protocols are convenient

Topology structure of the protocol is easier to analyze

- Intuitively, subdivisions are obtained, perhaps with holes in more powerful models



Three process examples
2 and 3 rounds

We pose the question:

What is the minimum number of bits per message, needed to encode a full-information protocol, without incurring a cost in communication rounds?

- It is easy to encode a full information protocol by sending smaller messages, at the cost of extra rounds

What is the minimum number of bits per message, needed to encode a full-information protocol, without a cost in communication rounds?

What is the cost in terms of communication rounds, of having *constant size* messages ?

Characterizations depend on the protocol being full-information

Characterizations depend on the protocol being full-information

- In non full-information protocols the topology may change; the protocol complex changes

Characterizations depend on the protocol being full-information

- In non full-information protocols the topology may change; the protocol complex changes
- First fix a model…

Wait-free dynamic network

Wait-free dynamic network

- Equivalent to other wait-free models, at the core of the Herlihy-Shavit characterization

Wait-free dynamic network

- Equivalent to other wait-free models, at the core of the Herlihy-Shavit characterization
- Round-based

Wait-free dynamic network

- Equivalent to other wait-free models, at the core of the Herlihy-Shavit characterization
- Round-based
- A directed graph determines messages delivered in each round

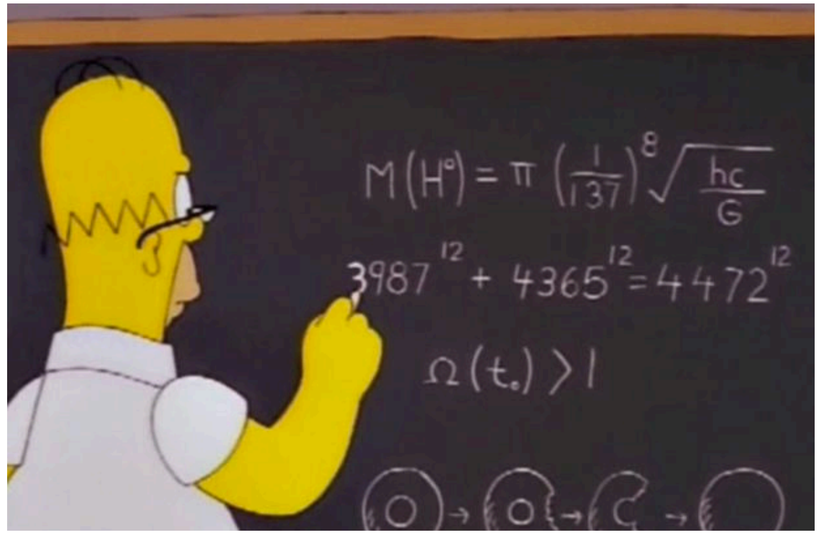
Wait-free dynamic network

- Equivalent to other wait-free models, at the core of the Herlihy-Shavit characterization
- Round-based
- A directed graph determines messages delivered in each round
- Processes do not crash

Wait-free dynamic network

- Equivalent to other wait-free models, at the core of the Herlihy-Shavit characterization
- Round-based
- A directed graph determines messages delivered in each round
- Processes do not crash

Results



https://www.thewrap.com/simpsons-predicted-future-gaga-nobel-disney-fox-hornets/

Results for two processes

Results for two processes

- Mostly single bit messages suffice, to solve any solvable task, without an extra cost in the number of rounds, w.r.t. to a full-information protocol

Results for two processes

- Mostly single bit messages suffice, to solve any solvable task, without an extra cost in the number of rounds, w.r.t. to a full-information protocol
- Additionally, to identify the input configuration (an edge), sometimes messages of log c bits must be sent, where c is a chromatic number of the characteristic, distance-2 input graph, but this vanishes as the number of rounds grows

Results for two processes

- Mostly single bit messages suffice, to solve any solvable task, without an extra cost in the number of rounds, w.r.t. to a full-information protocol
- Additionally, to identify the input configuration (an edge), sometimes messages of log c bits must be sent, where c is a chromatic number of the characteristic, distance-2 input graph, but this vanishes as the number of rounds grows
- No penalty on the number of rounds even sending only beeps

Results for more than 2 processes

- Already for 3 processes, messages of constant size do not suffice to solve every task in an optimal number of rounds
- Wait-free computability inherently requires rounds with growing information exchange

Techniques

Approximate agreement

- Wait-free task solvability is essentially a form of approximate agreement
  - e.g. see Book Herlihy, Kozlov, Rajsbaum
- Processes are required to compute values that are close to each other

7 edges => outputs are 1/7-th appart

k-approximate agreement, 3^k edges

$$\epsilon = 1/3^k$$

Two process
Approximate agreement

- For an input graph $\mathcal{I}$ of only one input edge
- Solving k-edge approximate agreement with 1-bit messages, in k rounds

AVERAGING N-APPROXIMATE AGREEMENT (ℓ)

```
1  round r from 1 to k do
2      send(ℓ)
3      m = receive()
4      if m ≠ ⊥ then
5          ℓ = ℓ/3 + 2m/3
6  ouput ℓ
```

- decide values arbitrarily close to each other, by running enough rounds, k. However, the size of messages sent grows with k.

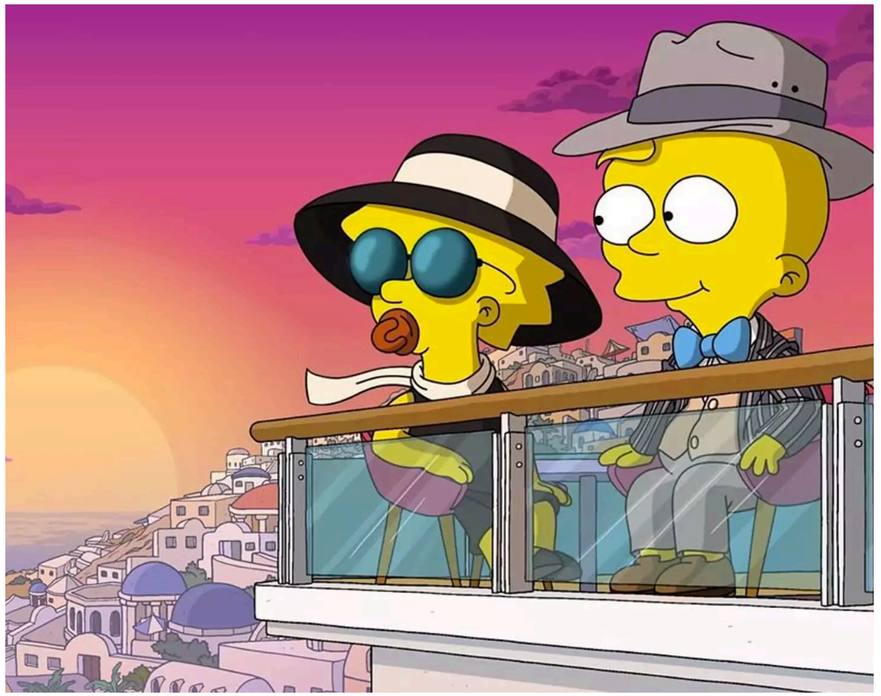
The foundation

Approximate agreement

Theorem 1. *A protocol that can solve k-approximate agreement on $\mathcal{I}$ in k rounds (for any given $k \geq 0$, and input graph $\mathcal{I}$), can be used to solve any solvable task, in an optimal number of rounds.*

Two processes results

first

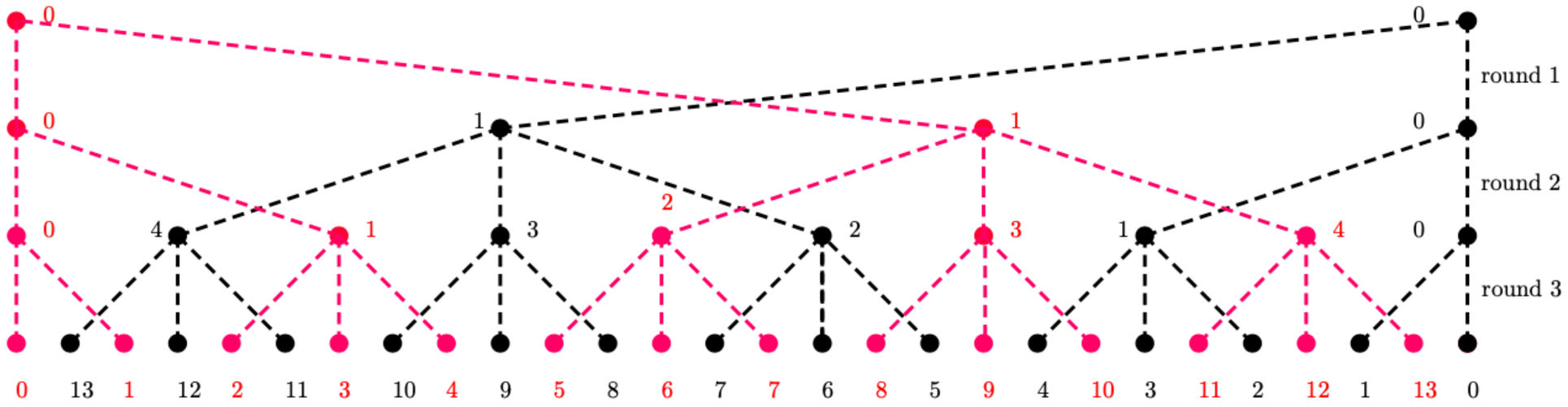


https://www.kiss.ie/disney-plus-drop-a-special-simpsons-episode-focussing-on-maggie/

Approximate agreement with few bits

First main result

- For an input graph $\mathcal{I}$ of only one input edge
- Solving k-edge approximate agreement with 1-bit messages, in k rounds



Approximate agreement with few bits

First main result

- For an input graph $\mathcal{I}$ of only one input edge
- Solving k-edge approximate agreement with 1-bit messages, in k rounds

ONE-BIT MESSAGES N-APPROXIMATE AGREEMENT (ℓ) /* input $\ell \in \{0,1\}$ */

1 view $= ()$ /* start with empty view */
2 nbMsg $= 0$ /* total number of messages received until now */
3 **round** r **from** 1 **to** k **do**
4 send((nbMsg $+ \ell$) mod 2) /* send the parity of nbMsg $+ \ell$ */
5 $m = receive()$ /* receive $m \in \{0, 1, \perp\}$ */
6 **if** $m \neq \perp$ **then** nbMsg $=$ nbMsg $+ 1$ /* update nbMsg */
7 view $=$ view $\cdot\, m$ /* append m to the view */
8 **output** $\delta(\ell, \text{view})$ /* decision depends on the input and the final view */

Approximate agreement with few bits

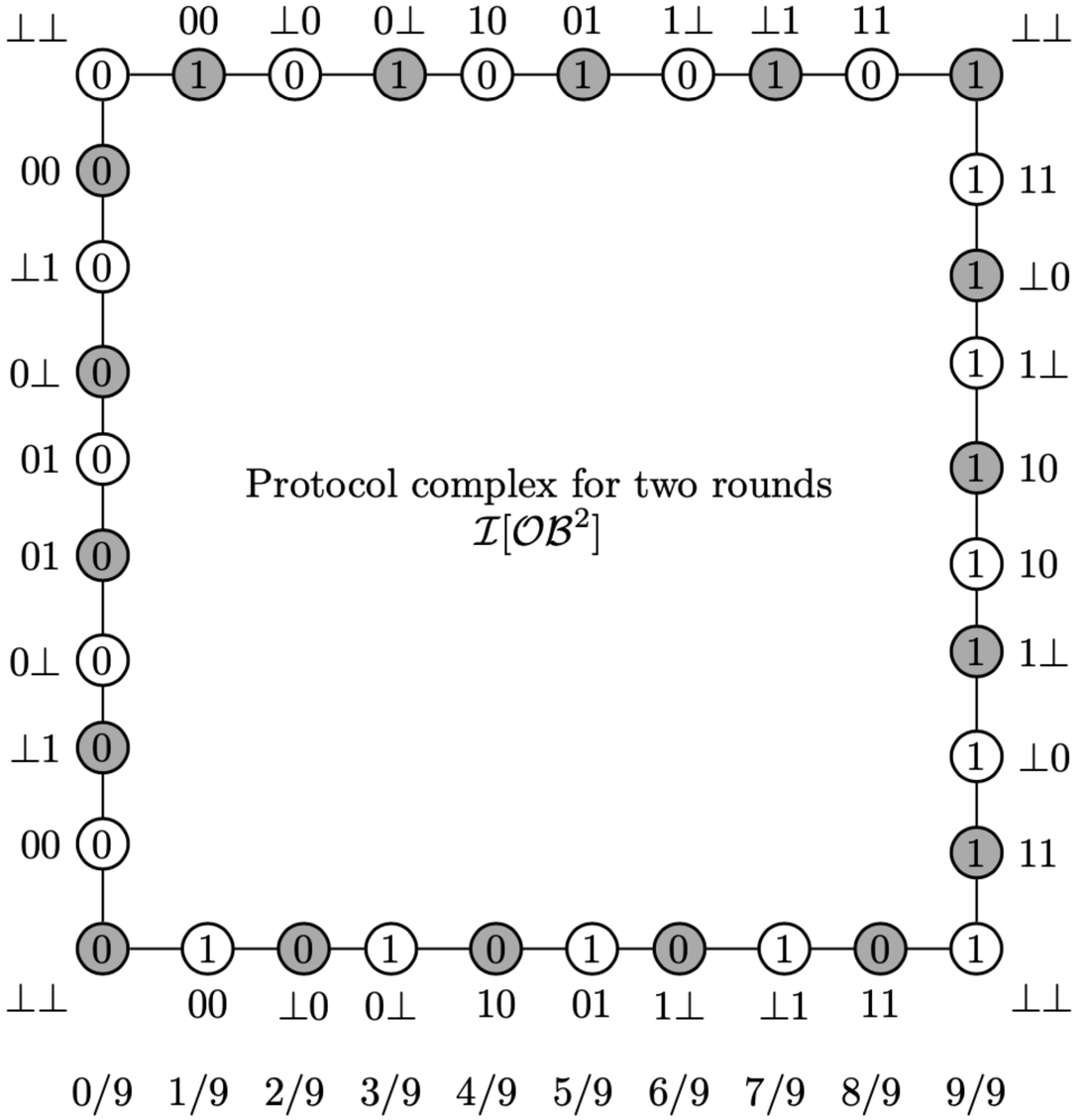
Lower bound result

- For a general input graph $\mathcal{I}$, at each round must send $\approx \log d$ bits, but in a constant number of executions

Approximate agreement with few bits

Lower bound result

- For binary consensus, two round protocol complex
- Formulation from:
  Armenta, Ledent, R. LANMR 2020,
- where using epistemic logic it is formalized what exactly a process learns, and why is optimal



Approximate agreement with few bits

Lower bound result

- For a general input graph $\mathcal{I}$, at each round must send $\approx \log d$ bits, but in a constant number of executions

Theorem 4. *Let d be the degree of a vertex (A, a) in $\mathcal{I}$ with $d \geq 2$. Every k-round k-approximate agreement protocol on $\mathcal{I}$ has d executions (starting in the d edges incident on (A, a) with schedule $(A \to B)^k$), where messages of at least $\frac{\log_2(d+2)}{2} - 1$ bits are sent by B at each round on average over these d executions. At least $\frac{k \log_2(d+2)}{2}$ bits are sent on average over these d executions.*

Approximate agreement with few bits

Lower bound result

- For a general input graph $\mathcal{I}$, at each round must send $\approx \log d$ bits, but in a constant number of executions

Theorem 4. *Let d be the degree of a vertex (A, a) in $\mathcal{I}$ with $d \geq 2$. Every k-round k-approximate agreement protocol on $\mathcal{I}$ has d executions (starting in the d edges incident on (A, a) with schedule $(A \to B)^k$), where messages of at least $\frac{\log_2(d+2)}{2} - 1$ bits are sent by B at each round on average over these d executions. At least $\frac{k \log_2(d+2)}{2}$ bits are sent on average over these d executions.*

Upper bound for general input graphs

Input information transmission as a coding problem

Upper bound for general input graphs

Input information transmission as a coding problem

- Process A, with input a, does not need to know the input of B, it is sufficient that it distinguishes the different possible inputs of B

Upper bound for general input graphs

Input information transmission as a coding problem

- Process A, with input a, does not need to know the input of B, it is sufficient that it distinguishes the different possible inputs of B
- View a protocol as a code: a vertex coloring of two graphs

Upper bound for general input graphs

Input information transmission as a coding problem

- Process A, with input a, does not need to know the input of B, it is sufficient that it distinguishes the different possible inputs of B
- View a protocol as a code: a vertex coloring of two graphs
  - the distance 2 graph of the A-vertices and of the distance 2-graph of the B-vertices

Input information transmission as a coding problem

Result

$$c = \max\{chromNumb(\mathcal{I}_A), chromNumb(\mathcal{I}_B)\}$$

Theorem 5. *The average number of bits sent in an execution of the protocol in Fig. 4, over all executions, is at most* $2k + 4\log c$ *bits. Namely, messages are of size at most* $1 + \frac{2\log c}{k}$ *bits on average.*

Beep model

https://giphy.com/gifs/season-16-the-simpsons-16x7-xT5LMS2wWNZb5qZmGA

The beep model

The beep model

- all messages sent by a protocol are identical: they consist of a unary signal

The beep model

- all messages sent by a protocol are identical: they consist of a unary signal
- A process can decide in each round to send a message or not

The beep model

- all messages sent by a protocol are identical: they consist of a unary signal
- A process can decide in each round to send a message or not
- Notice that if both A and B decide to send a beep in the same round, at least one of them receives a beep. But if only one of them sends, possibly no-one receives.

Protocols in the beep model

For approximate agreement

- it is possible to solve k-edge approximate agreement in 3k rounds

A	B	A	B	A	B	A	B	A	B
0	4	1	3	2	2	3	1	4	0
0	1	1	0	1	1	0	1	1	0
1	0	1	1	0	1	1	0	1	1
1	1	0	1	1	0	1	1	0	1

0 means no beep is sent

Beep model

Only a constant cost on the extra number of rounds over the optimal

Let c_a (resp. c_b) be the chromatic number of the graph $\mathcal{I}_A$ (resp. $\mathcal{I}_B$).

Theorem 6. *The k-approximate agreement task on $\mathcal{I}$ is solvable in the beep model in less than $4k \cdot c_a \cdot c_b$ rounds.*

Three processes

There is no solution with constant overhead on the number of rounds

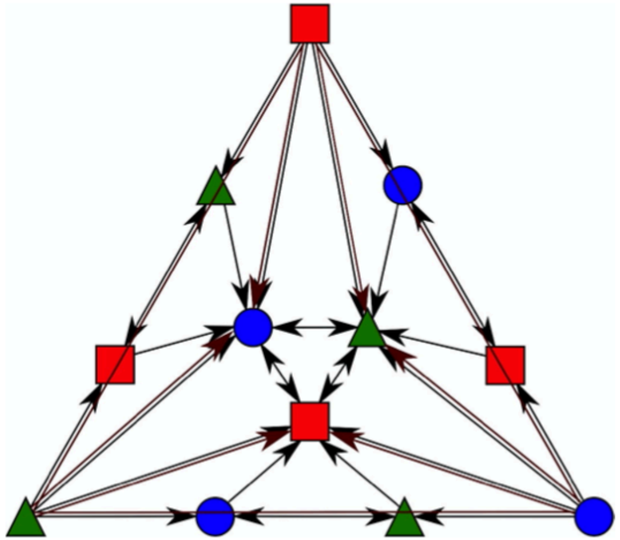
https://www.pinterest.com/pin/631418810241227883/

Dynamic network wait-free model

For 3 processes

Each triangle represents a possible digraph

describing which messages are delivered



The impossibility

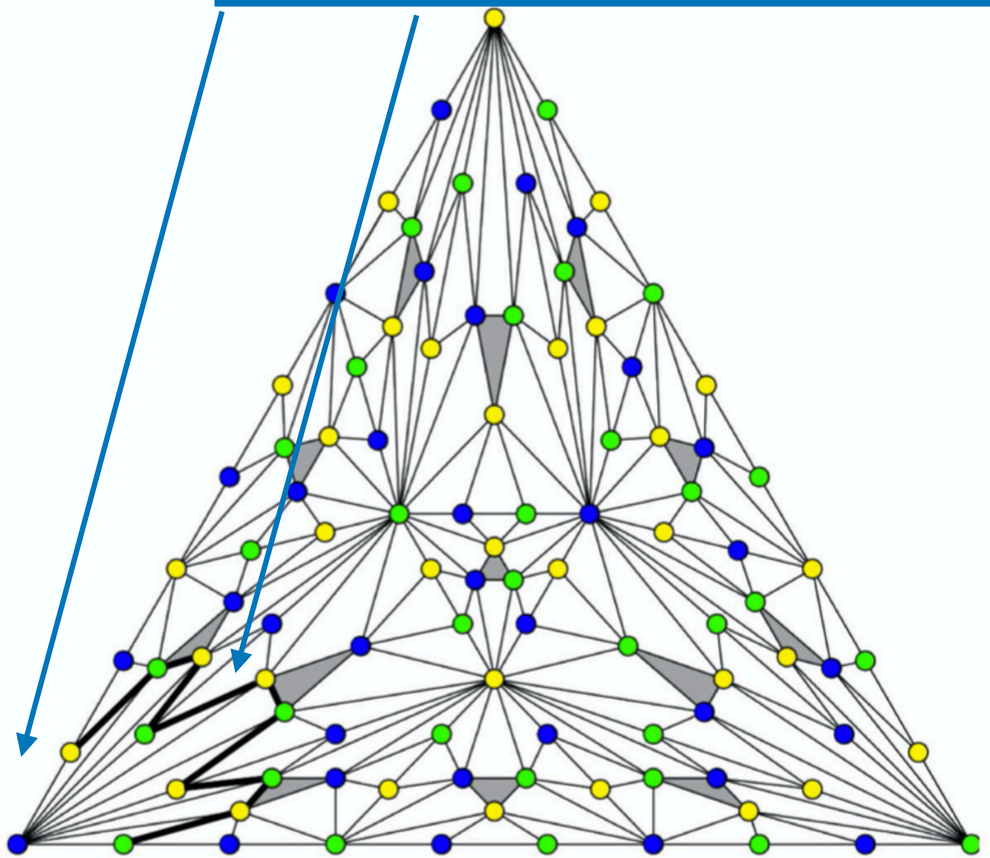
Intuition

- As the number of rounds grows, the size of the code needed to distinguish executions grows

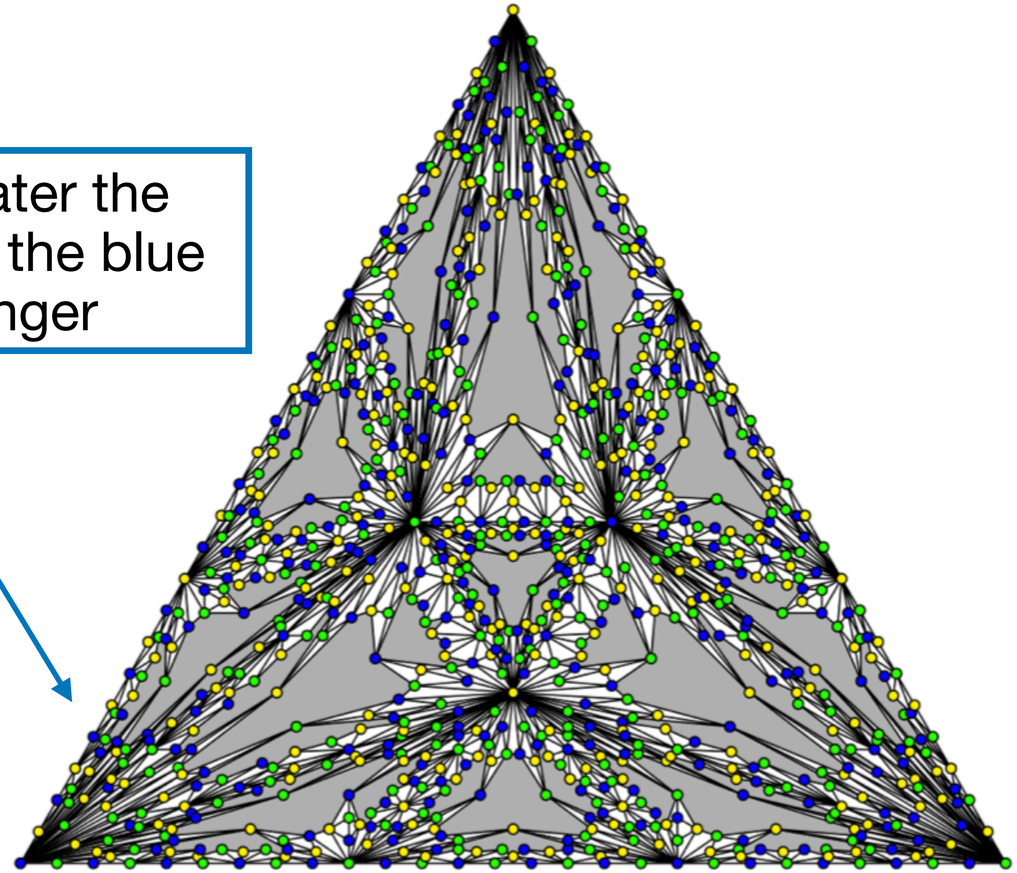
The impossibility

Intuition

Blue process must distinguish all 9 possible executions by the other two



One round later the path around the blue is 3 times longer



The impossibility

Main result

Theorem 7. *Every k-round implementation of a full-information protocol for three processes has executions where A, B send in a round messages (together) of $\Omega(k)$ bits.*

END

of Part II

Part III

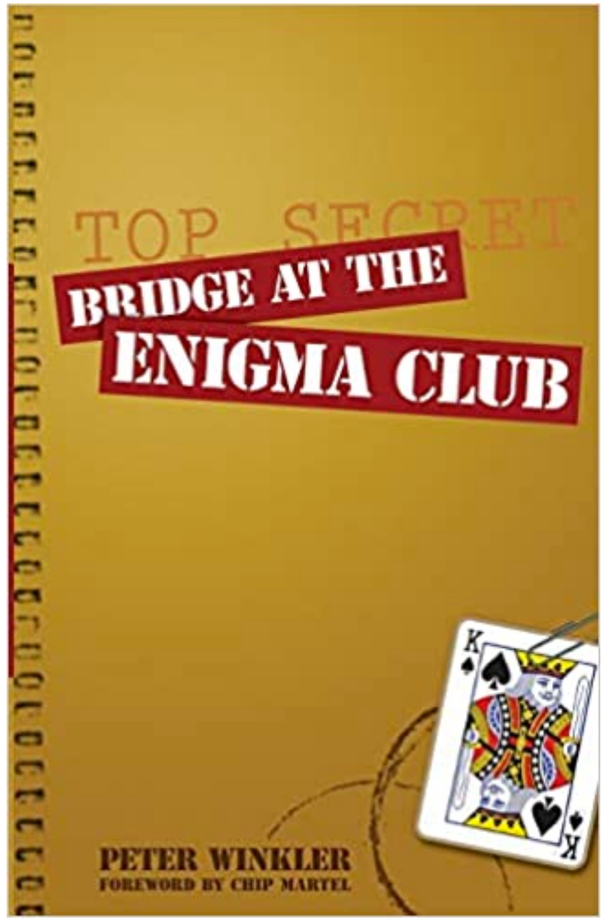
Privacy: Russian cards perspective

A Distributed Computing Perspective of Unconditionally Secure Information Transmission in Russian Cards Problems

Sergio Rajsbaum
UNAM, Mexico

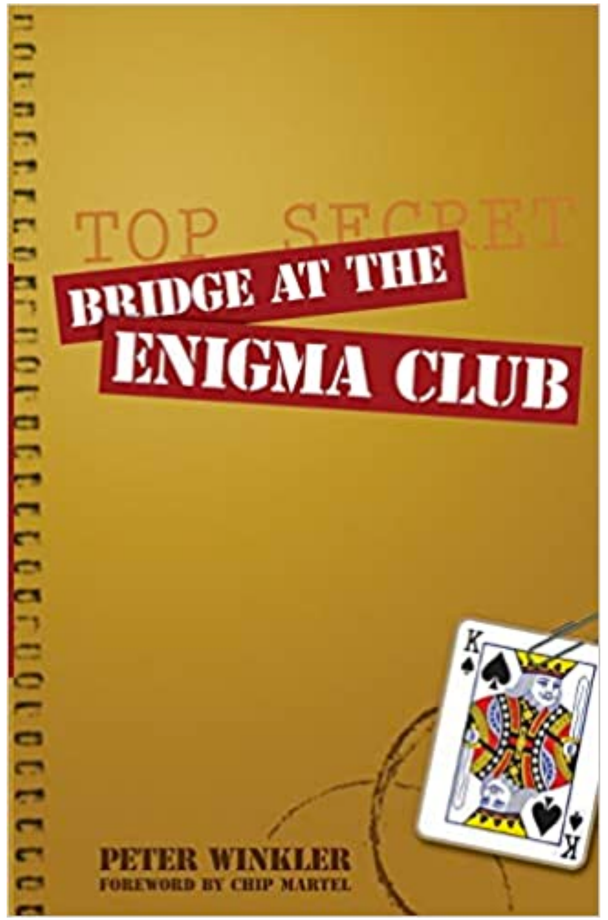
SIROCCO 2021

Card games


TOP SECRET
BRIDGE AT THE
ENIGMA CLUB
PETER WINKLER
FOREWORD BY CHIP MARTEL

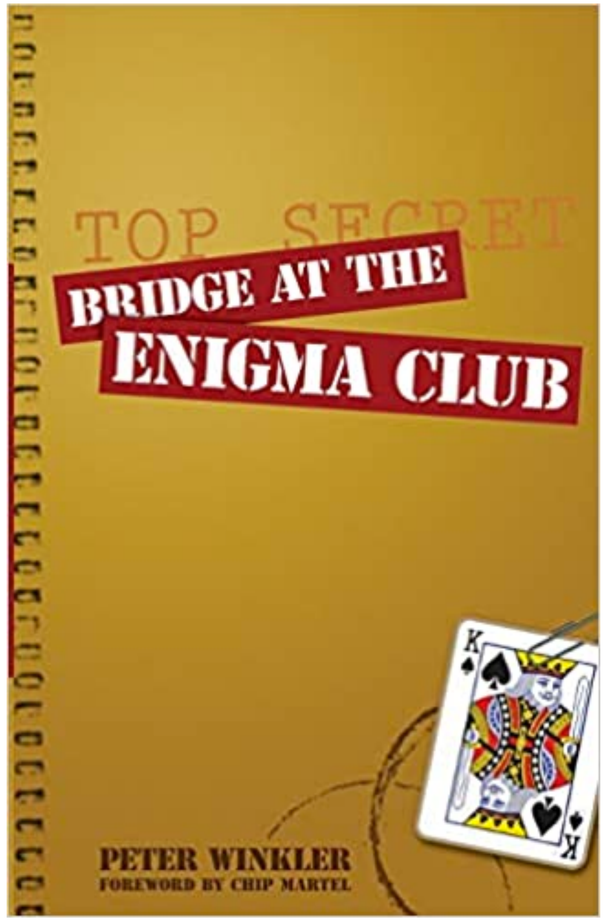
Card games

- Peter Winkler in Bridge Magazine 1981 showed how one player could send her partner information about her hand,



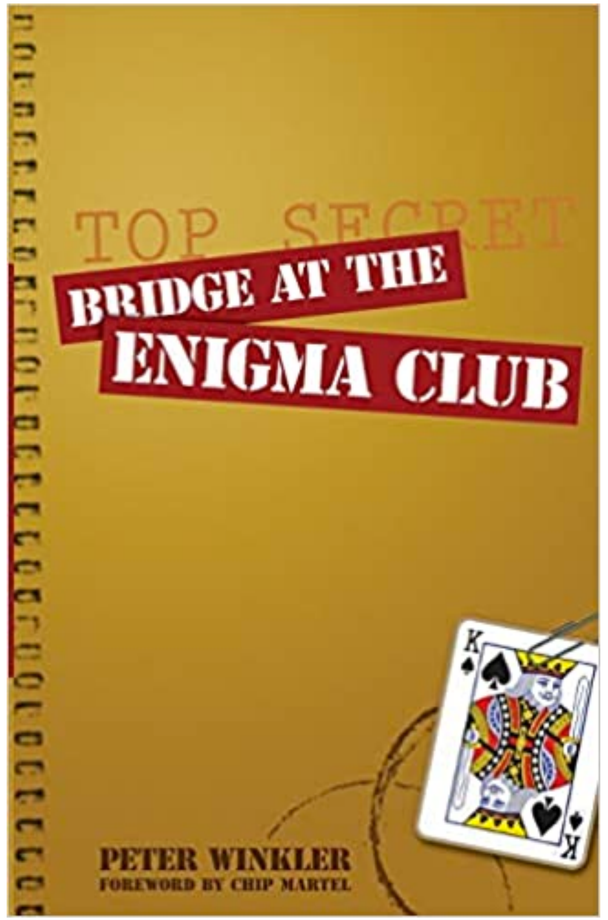
Card games

- Peter Winkler in Bridge Magazine 1981 showed how one player could send her partner information about her hand,
- by public announcements,



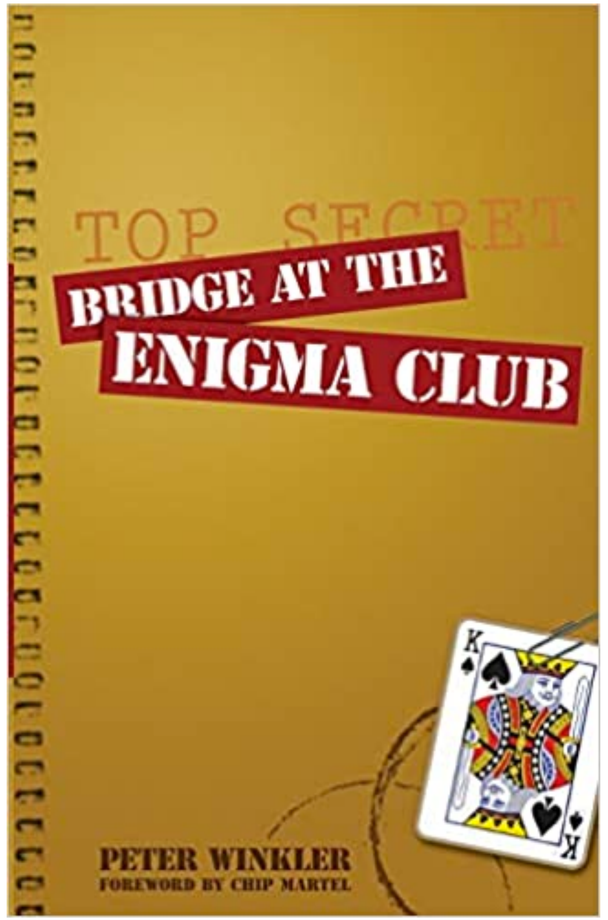
Card games

- Peter Winkler in Bridge Magazine 1981 showed how one player could send her partner information about her hand,
- by public announcements,
- undecipherable to the opponents,



Card games

- Peter Winkler in Bridge Magazine 1981 showed how one player could send her partner information about her hand,
- by public announcements,
- undecipherable to the opponents,
- even though the protocol was known to them



Card games

- Fischer, Wright, etc extended the ideas for secret bit and key exchange, eg

J. Cryptology (1996) 9: 71–99

Journal of
CRYPTOLOGY

Bounds on Secret Key Exchange Using a Random Deal of Cards*

Michael J. Fischer
Computer Science Department, Yale University,
New Haven, CT 06520-8285. U.S.A.

Rebecca N. Wright
AT&T Bell Laboratories, 600 Mountain Avenue, Room 2T-314,
Murray Hill, NJ 07974-0636, U.S.A.

Communicated by Joan Feigenbaum

Received 24 September 1993 and revised 18 December 1994

Abstract. We present a general model for communication among a "team" of players overheard by a passive eavesdropper, Eve, in which all players including Eve are given private inputs that may be correlated. We define and explore secret key exchange in this model. Our secrecy requirements are information-theoretic and hold even if Eve is computationally unlimited. In particular, we consider the situation in which the team players are dealt hands of cards of prespecified sizes from a known deck of distinct cards. We explore when the team players can use the information contained in their hands to determine a value that each team player knows exactly but Eve cannot guess.

Key words. Multiparty protocols, Correlated random variables, Key exchange, Perfect secrecy.

1. Introduction

1.1. *An Example*

Consider a scenario in which Alice, Bob, and a computationally unlimited eavesdropper, Eve, are playing a game of cards with a deck of four cards, J, Q, K, and A. Alice is given two cards, Bob is given one card, and the remaining card may or may not be given to Eve. Can Alice and Bob communicate publicly to agree on a bit that is secret from Eve? The answer to this question depends on what is meant by "secret," whether Eve is allowed to look at the remaining card, and whether Alice and Bob are allowed to use randomization.

If Eve is not allowed to look at the remaining card, the following deterministic protocol

Card games

Security in the presence of computationally unbounded adversaries

- Correlated inputs are needed for security in the presence of computationally unbounded adversaries

Card games

Security in the presence of computationally unbounded adversaries

- Correlated inputs are needed for security in the presence of computationally unbounded adversaries

- cards represent correlated initial local variables for the players

Russian Cards problem

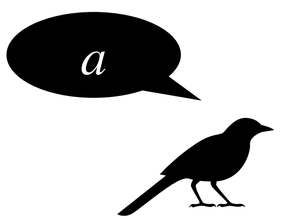
Moscow Mathematics Olympiad in 2000

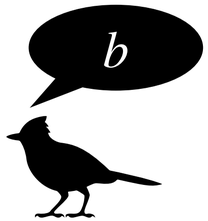
- A deck of 7 cards {0,1,2,3,4,5,6}

Russian Cards problem

Moscow Mathematics Olympiad in 2000

- A deck of 7 cards {0,1,2,3,4,5,6}
- Each of Alice and Bob have 3 cards, eg $a = \{012\}$ and $b = \{345\}$

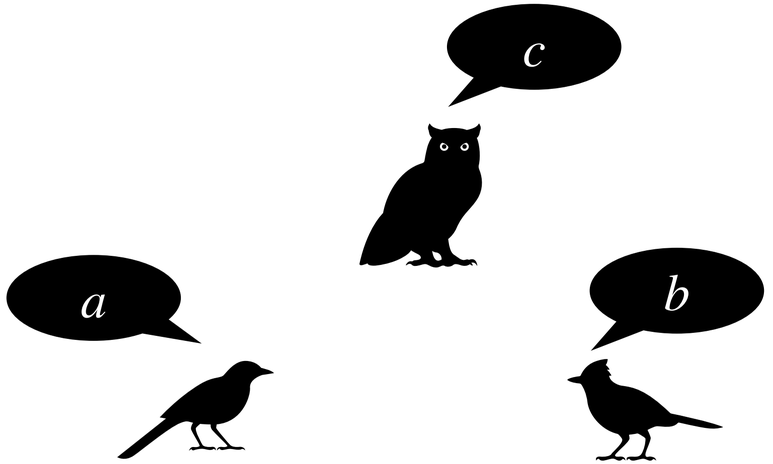




Russian Cards problem

Moscow Mathematics Olympiad in 2000

- A deck of 7 cards {0,1,2,3,4,5,6}
- Each of Alice and Bob have 3 cards, eg, $a = \{012\}$ and $b = \{345\}$
- Cath has the remaining card out of 7 eg, $c = \{6\}$



Russian Cards problem

Moscow Mathematics Olympiad in 2000

Goal

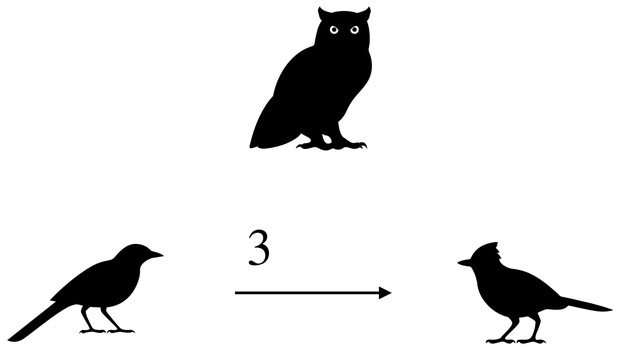
- Alice and Bob learn each other cards using public announcements (no encryption)
- While Cath should not learn the location of any single card (in spite of knowing the protocol of Alice and Bob)

Russian Card problem

Moscow Mathematics Olympiad in 2000

Classic two-step solution

- Alice announces sum mod 7 of $a = \{012\}$ = 3

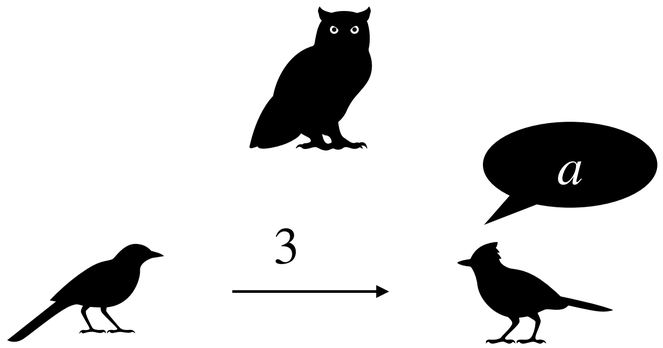


Russian Card problem

Security in the presence of computationally unbounded adversaries

Two-step solution

- Alice announces sum of a mod 7,
- Bob deduces $a = \{012\}$ from his own b and a mod 7

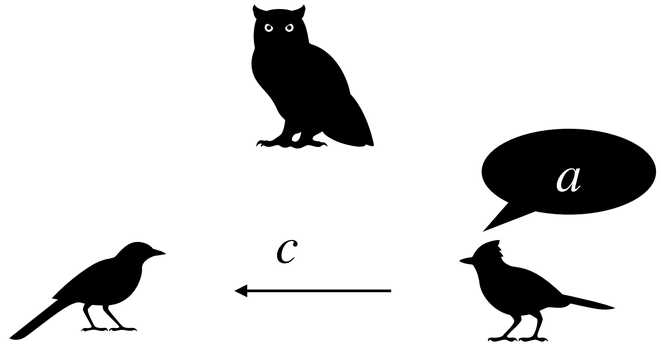


Russian Card problem

Security in the presence of computationally unbounded adversaries

Two-step solution

- Alice announces sum of a mod 7,
- Bob deduces $a = \{012\}$ from his own b and a mod 7
- Bob responds with $c = \{6\}$

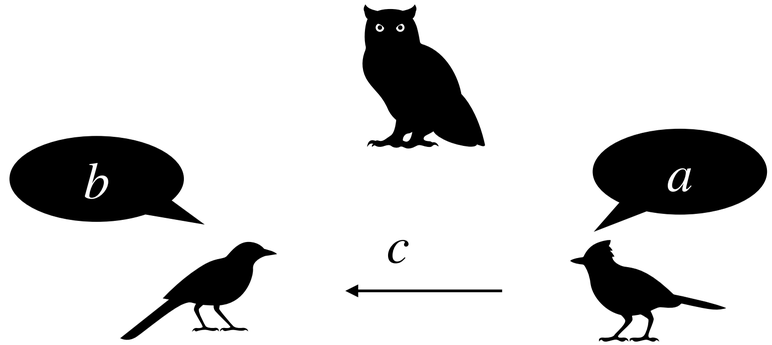


Russian Card problem

Security in the presence of computationally unbounded adversaries

Two-step solution

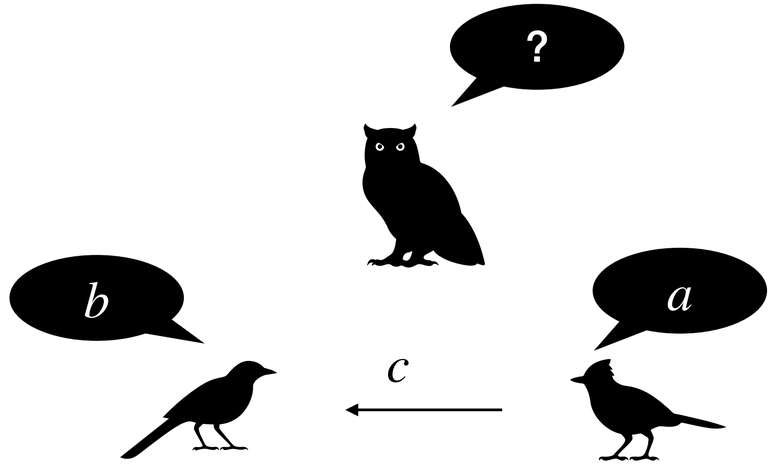
- Alice announces sum of a mod 7,
- Bob deduces $a = \{012\}$ from his own b and a mod 7
- Bob responds with $c = \{6\}$
- from which Alice learns b



Russian Card problem

Security in the presence of computationally unbounded adversaries

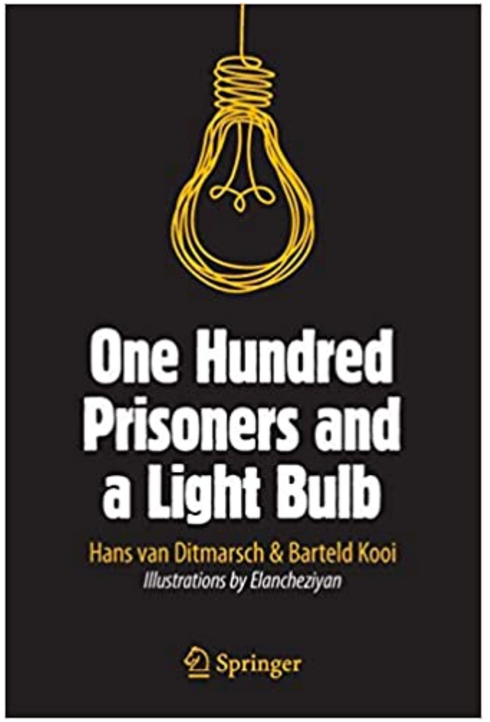
- two-step solution:
- Alice announces sum of a mod 7,
- Bob deduces $a = \{012\}$ from his own b and a mod 7
- Bob responds with $c = \{6\}$
- from which Alice learns b
- and Cath learns nothing



Generalized Russian Cards

- Hans van Ditmarsch started a research line on the *generalized Russian cards* problem:
- deck of *n* cards, Alice gets **a**, Bob gets **b**, and Cath gets **c**

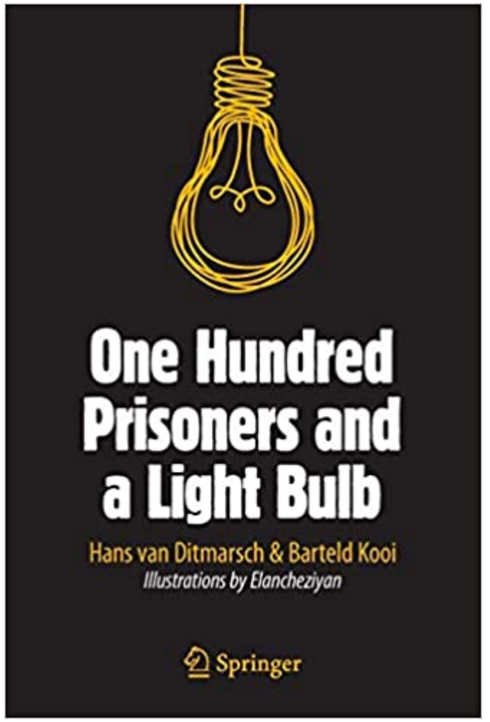
Signature
(a,b,c),
***n* =a+b+c**



Generalized Russian Cards

- Hans van Ditmarsch started a research line on the *generalized Russian cards* problem:
- deck of *n* cards, Alice gets **a**, Bob gets **b**, and Cath gets the remaining **c**
- **Same goal:** Alice and Bob learn each other cards using public announcements, while Cath should not learn the location of any single card
- mostly *two-step solutions*

Signature (a,b,c), *n* =a+b+c



Generalized Russian Cards

- Numerous papers by van Ditmarsch, Fernández-Duque, Swanson, Stinson, etc
- Using either combinatorial designs or epistemic logic
- including variants such as stronger security requirements or more rounds

For a signature $(\mathbf{a}, \mathbf{b}, \mathbf{c}),\ n = \mathbf{a} + \mathbf{b} + \mathbf{c}$ **the question has been**

For what signatures there exists a 2-step protocol that is

- **Informative**: Bob learns the cards of Alice, and
- **Safe**: Cath does not learn the location of any single card

Known

- When **c** = 1, solutions exist where Alice announces the cards sum modulo an appropriate prime number greater or equal to *n*
- Only special cases are known when **c** >1

Contributions

We extend these results

- For the general case when **c** = 1, solutions exist that announce the cards sum modulo an appropriate prime number greater or equal to *n*
- Only special cases are known when **c** >1

Contributions

and new

For what signatures there exists a 2-step protocol?

Contributions

and new

For what signatures there exists a 2-step protocol?

1. How many bits of communication are needed?

Contributions

and new

For what signatures there exists a 2-step protocol?

1. How many bits of communication are needed?

2. Introduce the notion of *Minimally informative* protocol, where Alice wants to communicate *something* to Bob?

Contributions

and new

For what signatures there exists a 2-step protocol?

1. How many bits of communication are needed?
2. Introduce the notion of *Minimally informative* protocol, where Alice wants to communicate *something* to Bob?
3. What if there are **r** cards that are dealt to no one, so

Contributions

and new

For what signatures there exists a 2-step protocol?

1. How many bits of communication are needed?

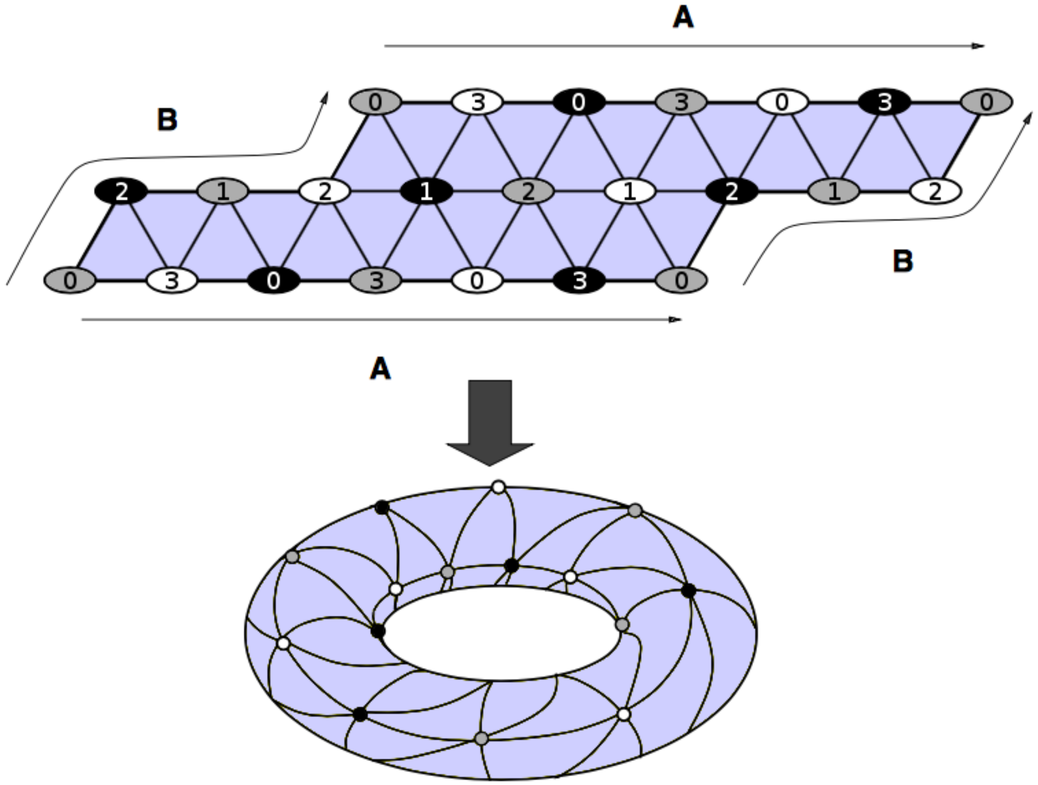
2. Introduce the notion of *Minimally informative* protocol, where Alice wants to communicate *something* to Bob?

3. What if there are **r** cards that are dealt to no one, so

$$n = \mathbf{a}+\mathbf{b}+\mathbf{c}+\mathbf{r} \ ?$$

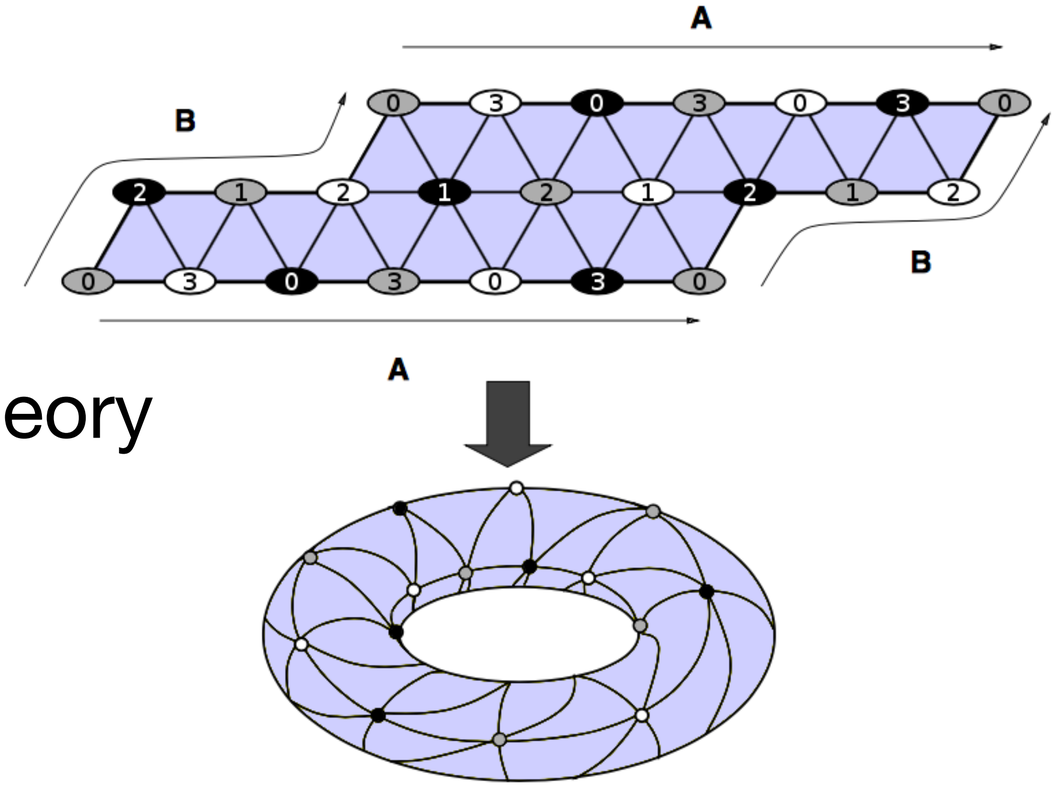
A new perspective

- Inspired by distributed computing



A new perspective

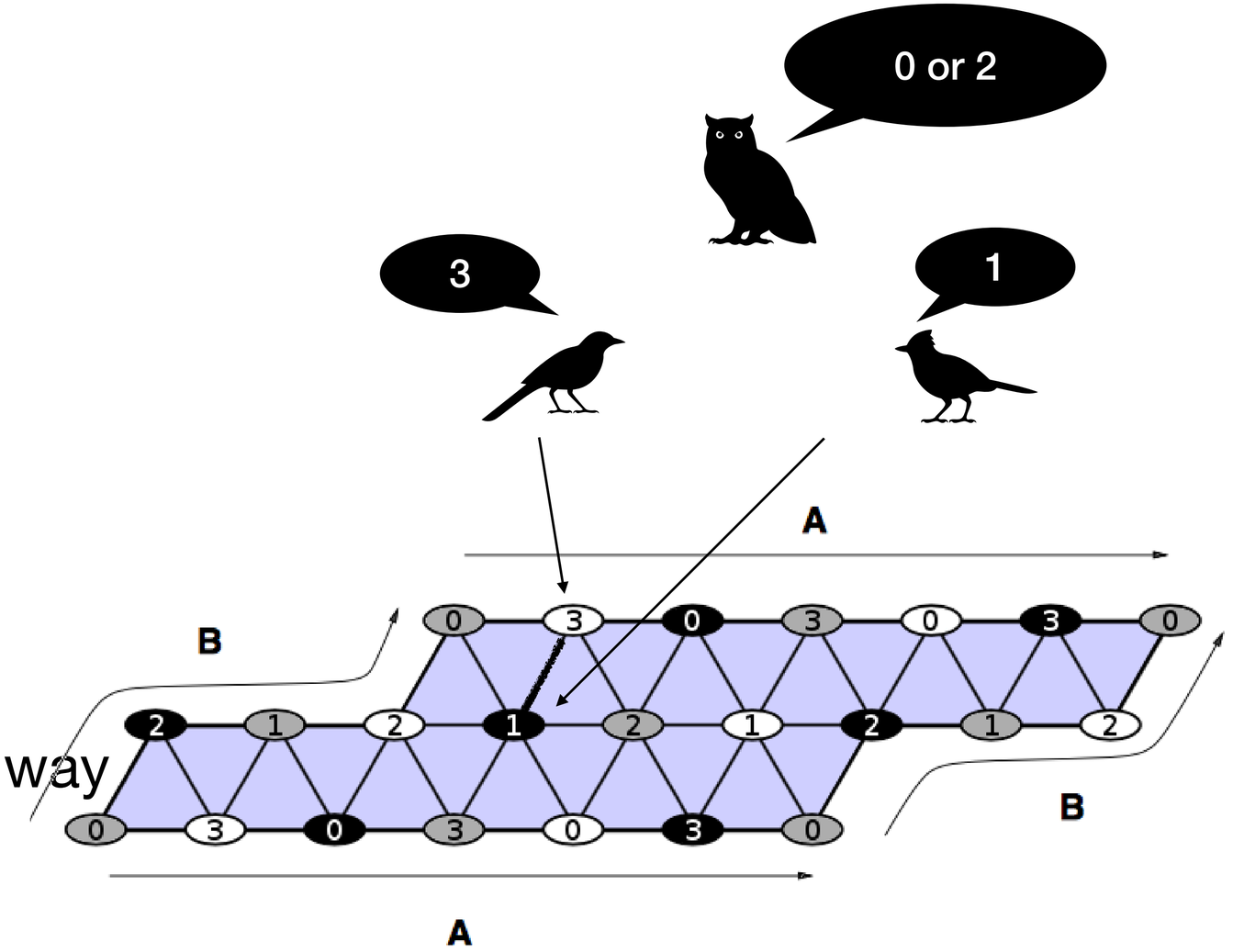
- Inspired by distributed computing
- based on Johnson graphs
- using coding and additive number theory techniques



A new perspective

- Correlated inputs

- n = 4, **a** =1, **b**=1, **c**=1
- Each triangle represents a possible way of dealing the cards (appears in renaming)
- As opposed the the standard model used in *communication complexity*



The indistinguishability, characteristic graphs of Bob and Cath are $J^d(n, \mathbf{a})$, *d-distance Johnson* Graphs!

- vertices are $\mathscr{P}_{\mathbf{a}}(D)$
-

The indistinguishability graphs of Bob and Cath are $J^d(n, \mathbf{a})$, *d-distance Johnson* graphs!

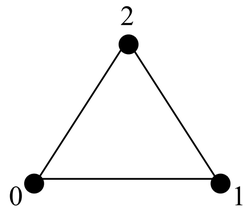
- vertices are $\mathscr{P}_{\mathbf{a}}(D)$
- edges of $J^d(n, \mathbf{a})$ are (a, a') when $\mathbf{a} - d \leq |a \cap a'|$
-

The indistinguishability graphs of Bob and Cath are $J^d(n, \mathbf{a})$, *d-distance Johnson* graphs!

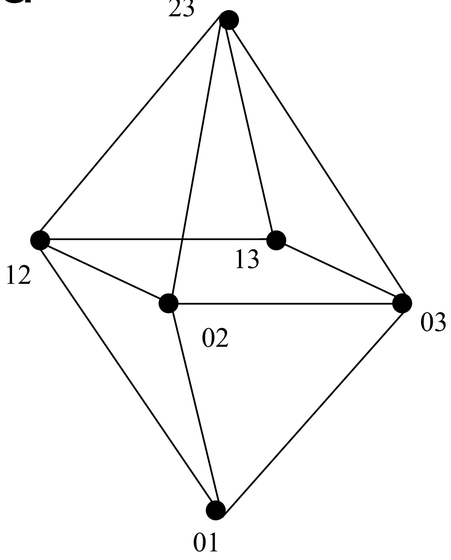
- vertices are $\mathscr{P}_{\mathbf{a}}(D)$
- edges of $J^d(n, \mathbf{a})$ are (a, a') when $\mathbf{a} - d \leq |a \cap a'|$
- When $d = 1$ we get the thoroughly studied

Johnson graphs

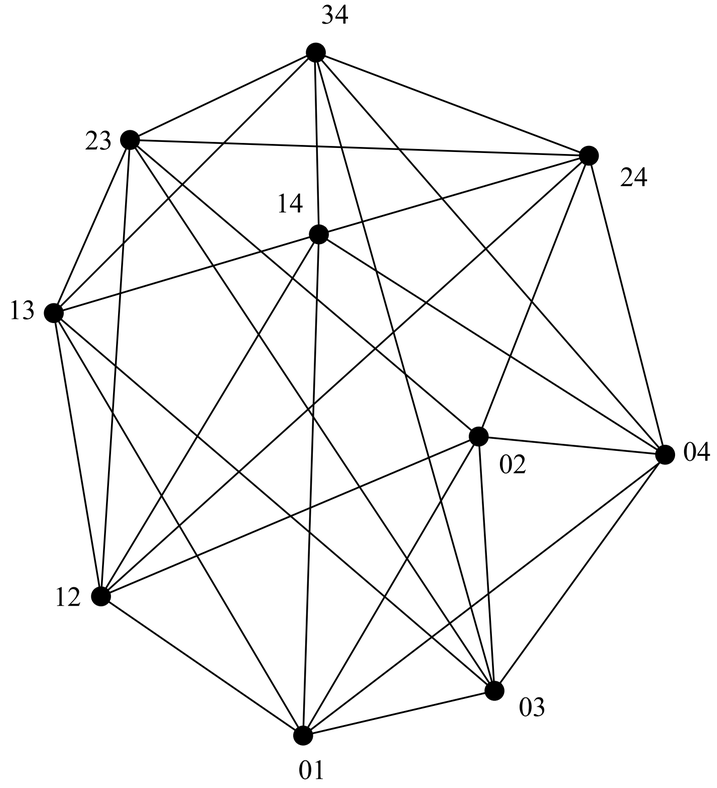
$J(n, \mathbf{a})$



$J(3,1)$



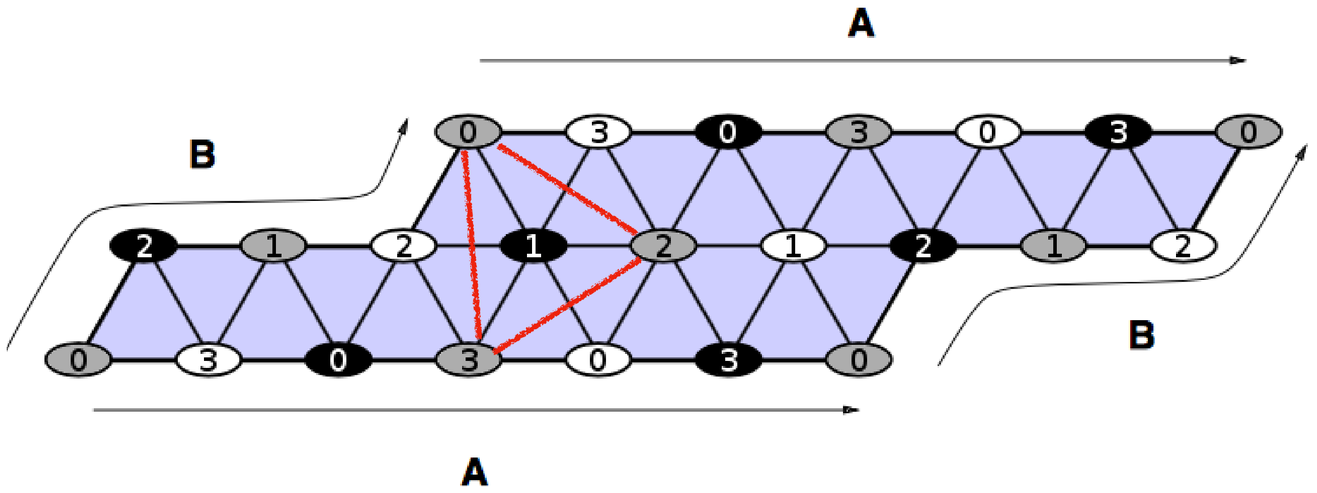
$J(4,2)$



$J(5,2)$

The clicks $K_p(\,\cdot\,)$ of the distance d Johnson graph

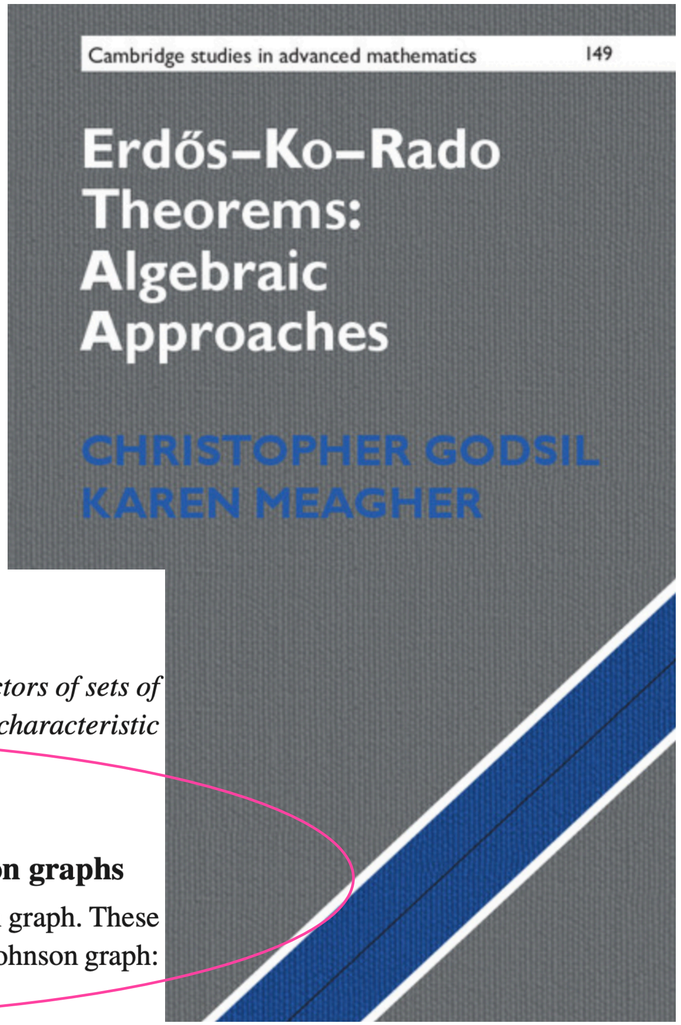
- Given a hand b of Bob, the hands of Alice that he considers possible form a click $K_p(b)$ in $J^{\mathbf{c+r}}(n, \mathbf{a})$



Chromatic number Johnson graphs

Needed for a protocol to be <u>informative</u>

- But even the chromatic number of Johnson graphs, when $d = \mathbf{c} + \mathbf{r} = 1$, is an open problem,
- that has been studied in coding theory, graph theory and combinatorics of intersecting sets



16.5.2 Problem. *Prove that the only balanced characteristic vectors of sets of size q, in the eigenspace belonging to θ of $P(q^2)$, are the balanced characteristic vectors of the canonical cliques.*

16.6 Determine the chromatic number of the Johnson graphs

In Section 6.1, we describe two families of cliques in the Johnson graph. These give the following lower bound on the chromatic number of any Johnson graph:

$$\max\{n - k + 1, k + 1\} \leq \chi(J(n, k)).$$

It is an old result that for all n and k the bound $\chi(J(n, k)) \leq n$ holds (see [89]). Further, it is not hard to determine that

Informative protocols

Main results

We adapt results from coding theory to show that for $J^{\mathbf{c}+\mathbf{r}}(n, \mathbf{a})$

Theorem

For an informative protocol $\Theta((\mathbf{c}+\mathbf{r})\log n)$ bits are needed and sufficient.

Informative and safe protocols

Main results

$d = 1$

Simple coding theory plus additive number theory to show that for $J(n, \mathbf{a})$

Theorem

$d = 1$

$\Theta(\log n)$ bits are needed and sufficient for an informative and safe protocol,

$$\mathbf{c} + \mathbf{r} = 1, \quad \mathbf{a}, \mathbf{b} \geq 3, n \geq 7$$

Alice sends the sum of her cards modulo n

Minimally informative

Main results

Minimally informative

Main results

- When $\mathbf{c} + \mathbf{r} = 1$

Minimally informative

Main results

- When $\mathbf{c} + \mathbf{r} = 1$

  Minimally informative is equivalent to Bob learning one card of Alice

Minimally informative

Main results

- When $\mathbf{c} + \mathbf{r} = 1$

  Minimally informative is equivalent to Bob learning one card of Alice

- in general, Bob learns that Alice has one of the cards in a set of size $\mathbf{c} + \mathbf{r} \geq 1$

Minimally informative

Main results

- When $\mathbf{c} + \mathbf{r} = 1$

  Minimally informative is equivalent to Bob learning one card of Alice

- in general, Bob learns that Alice has one of the cards in a set of size $\mathbf{c} + \mathbf{r} \geq 1$

Minimally informative

Main results: partial characterization of existence and reductions from the informative case

Theorem

One bit is sufficient for a minimally informative and safe protocol, $\mathbf{c} + \mathbf{r} = 1$, $\mathbf{a} > \lceil n/2 \rceil - 1$ and $\mathbf{b} < n/2$.

Alice sends the sum of her cards modulo 2.

.

Minimally informative

Main results: partial characterization of existence and reductions from the informative case

Theorem

One bit is sufficient for a minimally informative and safe protocol, $\mathbf{c} + \mathbf{r} = 1$, $\mathbf{a} > \lceil n/2 \rceil - 1$ and $\mathbf{b} < n/2$.

Alice sends the sum of her cards modulo 2.

There is a 1-bit minimally informative and safe protocol for the Russian cards $(3,3,1)$.

Conclusions

Alice and Bob, communication complexity and privacy under correlated inputs

- Studied under different perspectives: information theory, complexity theory, distributed computing, combinatorics, epistemic logic
- Using various techniques from information and coding theory, combinatorics, epistemic logic, topology, with interesting interactions
- Many open problems in the case of Alice and Bob, many more for arbitrary number of participants

Bibliography

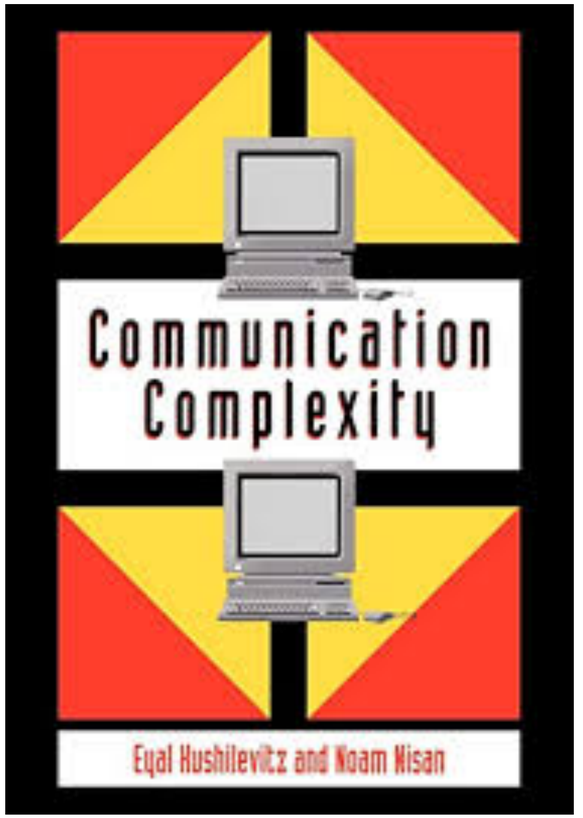
- Zero-error information theory, sequence of papers by Alon Orlitsky et al

  - Failure-free, information theory, coding perspective

2021 Claude E. Shannon Award Winner

Bibliography

- Communication complexity
  - Failure-free, functions instead of relations
  - important relations with complexity theory and other areas

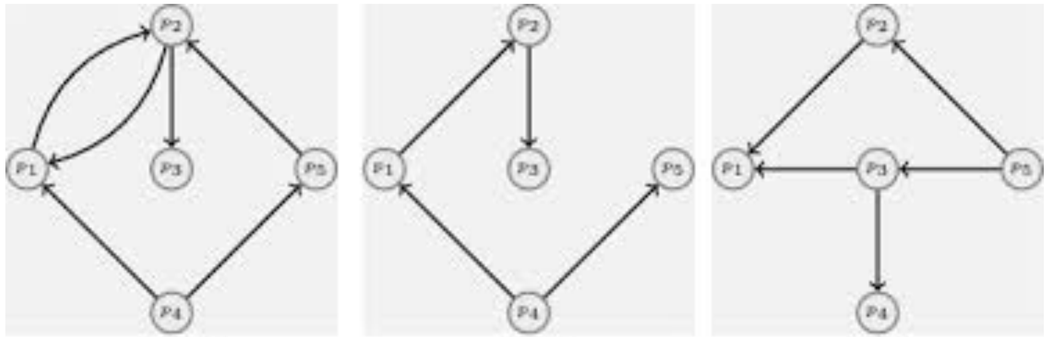


Bibliography

- Communication complexity
- Bit complexity in distributed computing
  - but not wait-free (Most papers on message complexity)

Bibliography

- Communication complexity
- Bit complexity in distributed computing
- Dynamic networks
  - mostly on characterizing the sequences of graphs that allow to solve various problems, often using full-information protocols, E. Goddard. B. Charron-Bost, U. Schmid, etc.



Bibliography

- Communication complexity
- Bit complexity in distributed computing
- Dynamic networks
- Beep models

Question

Can you tell the temperature by listening to the chirping of a cricket?

Answer

Yes!

Life stages of the Mormon cricket: egg, first instar nymph, third instar nymph, and adult female.
Agricultural Research Service, U.S. Department of Agriculture.

The frequency of chirping varies according to temperature. To get a rough estimate of the temperature in degrees fahrenheit, count the number of chirps in 15 seconds and then add 37. The number you get will be an approximation of the outside temperature.

https://www.loc.gov/everyday-mysteries/meteorology-climatology/item/can-you-tell-the-temperature-by-listening-to-the-chirping-of-a-cricket/

Bibliography

With coauthors

- Wait-free: with Carole and Hugues in SIROCCO’20
- Logic and wait-free: with Jorge Armenta and Jeremy Ledent in LANMR’20
- Oneway Russian cards in SIROCCO’21
- Two ways results on russian cards: with Eduardo Pascual and Zoe Leyva in SSS’21

Thanks for your attention