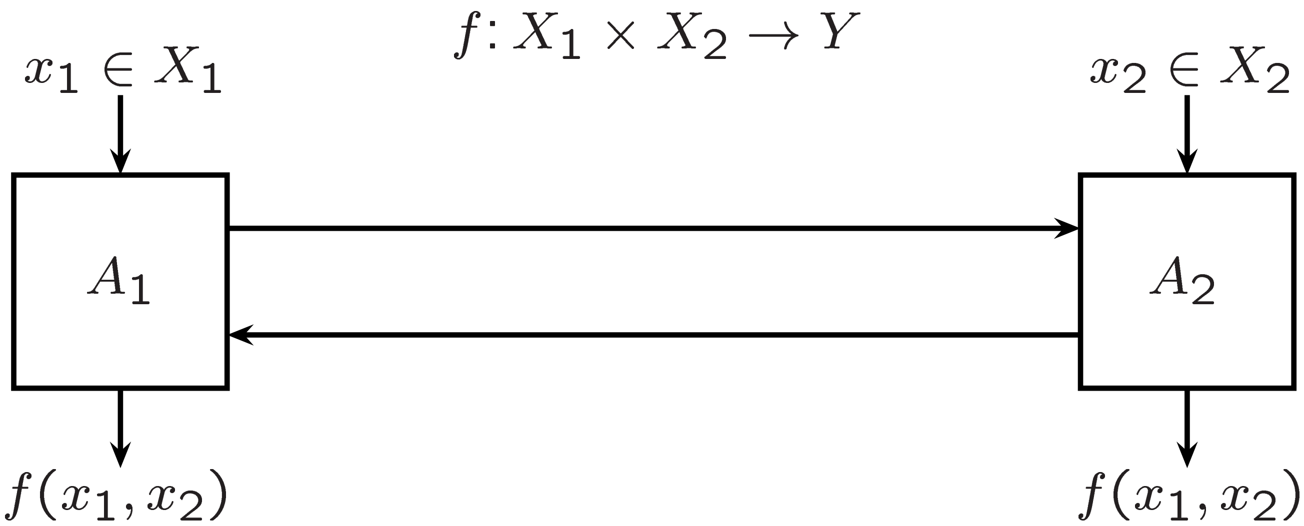
Models of (security) protocols

Marcin Peczarski

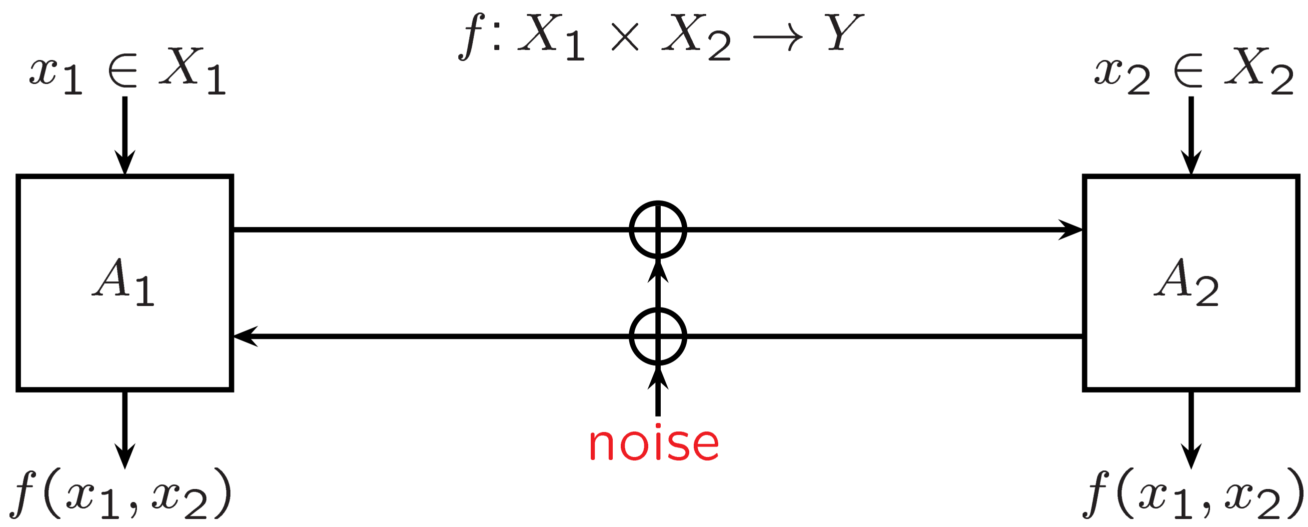
November 3, 2003

Communication Model [A. C. Yao, 1979]



- Communication complexity is the number of bits that must be exchanged by processors, in order to compute $f(x_1, x_2)$, measured on a "worst" input.

Communication Model on Noisy Channel



- Probabilistic channel
  Bits are incorrectly transmitted with a given probability ε.

- Adversarial channel
  The fraction of incorrectly transmitted bits is at most ε.

Coding Theorems [L. J. Schulman, 1992–1996]

Theorem 1 *For every noiseless communication protocol exchanging T bits there exists a protocol computing the same function with error probability $e^{-\Omega(T)}$ on the probabilistic channel and exchanging $\Theta(T/C)$ bits, where $C = 1+\varepsilon \lg \varepsilon + (1-\varepsilon)\lg(1-\varepsilon)$ is channel capacity for transmission error probability ε.*

Theorem 2 *For every noiseless communication protocol exchanging T bits there exists a protocol computing the same function successfully on the adversarial channel and exchanging $\Theta(T)$ bits provided the fraction ε of incorrectly transmitted bits is at most $\frac{1}{240}$.*

Proofs are not constructive.

Ulam's Game $\mathcal{U}\langle n, q, l\rangle$

- Two players Carole and Paul.
- Carole thinks of a integer number x between 1 and n.
- Paul asks at most q questions "Does $x \in F$?", $F \subset \{1, 2, \ldots, n\}$.
- Carole answers yes or no and she can lie at most l times.
- Paul wins if at the end of the game he knows x.

Three Thresholds Theorem [Spencer and Winkler, 1991]

Theorem 3 *Paul and Carole play game* $\mathcal{U}\langle n, q, \lfloor \varepsilon q \rfloor \rangle$.

- *Carole answers before Paul asks next question. The number of lies never exceeds a fraction* ε *of already given answers.*
  If $\varepsilon < \frac{1}{2}$ *then Paul has a winning strategy with* $\Theta(\log n)$ *questions.*
  If $\varepsilon \geq \frac{1}{2}$ *then Carole has a winning strategy.*
- *Carole answers before Paul asks next question.*
  If $\varepsilon < \frac{1}{3}$ *then Paul has a winning strategy with* $\Theta(\log n)$ *questions.*
  If $\varepsilon \geq \frac{1}{3}$ *then Carole has a winning strategy.*
- *Carole answers after Paul asked all his questions.*
  If $\varepsilon < \frac{1}{4}$ *then Paul has a winning strategy with* $\Theta(\log n)$ *questions.*
  If $\varepsilon = \frac{1}{4}$ *then Paul has a winning strategy with* $\Theta(n)$ *questions.*
  If $\varepsilon > \frac{1}{4}$ *then Carole has a winning strategy.*

Generalized Ulam's Game $\mathcal{G}\langle f, q, l\rangle$

- Two players Carole and Paul.
- Given function $f\colon X_1 \times X_2 \to Y$.
- Carole thinks of a pair $(x_1, x_2) \in X_1 \times X_2$.
- Paul asks at most q questions "Does $x_i \in F_i$?", where $F_i \subset X_i$.
- Carole answers yes or no and she can lie at most l times.
- Paul wins if at the end of the game he knows $f(x_1, x_2)$.

Game Determinization

- We allow Carole to play an adversary strategy.

- Carole doesn't actually pick a pair (x_1, x_2) but answers all questions so that there is always at least one pair that satisfies all answers, except for at most l of them.

- In this game either Paul or Carole has a winning strategy.

3D Game $\mathcal{G}\langle f, q, l\rangle$

- Game board is 3-dimensional chess-board. Fields are labeled with (x_1, x_2, k), where $x_1 \in X_1$, $x_2 \in X_2$, $k \in \{0, 1, 2, \ldots, l\}$.

- At beginning each field $(x_1, x_2, 0)$ is occupied by pebble of color $f(x_1, x_2)$. Other fields are empty.

- In each round Paul chooses $i \in \{1, 2\}$ and $F_i \subset X_i$ ($x_i \in F_i$?). Next Carole chooses $G_i = F_i$ (no) or $G_i = X_i \setminus F_i$ (yes). Let $G_{3-i} = X_{3-i}$. If $x_1 \in G_1$, $x_2 \in G_2$ and $k < l$ then pebble from (x_1, x_2, k) is moved to $(x_1, x_2, k+1)$. If $x_1 \in G_1$, $x_2 \in G_2$ and $k = l$ then pebble (x_1, x_2, k) is taken off from the board.

- Paul wins if after q rounds staying pebbles are in the same color or there is not pebble on the board.

Why we want to play this game?

Proposition 1 *If there exists a communication protocol* $\mathcal{P}$ *computing the function* f*, exchanging at most* q *bits and proof against maximum* l *incorrectly transmitted bits, then Paul has a winning strategy in the game* $\mathcal{G}\langle f, q, l\rangle$*.*

Opposite implication isn't true in general. Open problem is finding conditions for winning strategy under which equivalence holds.

Proposition 2 *If Paul has a winning strategy in the game* $\mathcal{G}\langle f, q, l\rangle$ *with condition* C *then there exists a communication protocol* $\mathcal{P}$ *computing the function* f*, exchanging at most* $c_1 q$ *bits and proof against maximum* $c_2 l$ *incorrectly transmitted bits.*

True for $C =$ "Paul plays non-adaptive strategy" and $c_1 = c_2 = 1$.

Thresholds for Generalized Ulam's Game

Proposition 3 *Paul and Carole play game* $\mathcal{G}\langle f, q, \lfloor \varepsilon q \rfloor \rangle$ *with condition* C. *There exists constant* c *such that:*

- *If* $\varepsilon < c$ *then for every function* f *of communication complexity* n *Paul has a winning strategy with* $\Theta(n)$ *questions.*
- *If* $\varepsilon > c$ *then there exists function* f *such that Carole has a winning strategy.*

For $C =$ "no condition" we know that $\frac{1}{240} \leq c \leq \frac{1}{3}$. What is exact value of c in this case?

For $C =$ "Paul plays non-adaptive strategy" we know that $c \leq \frac{1}{4}$. Is c positive in this case?

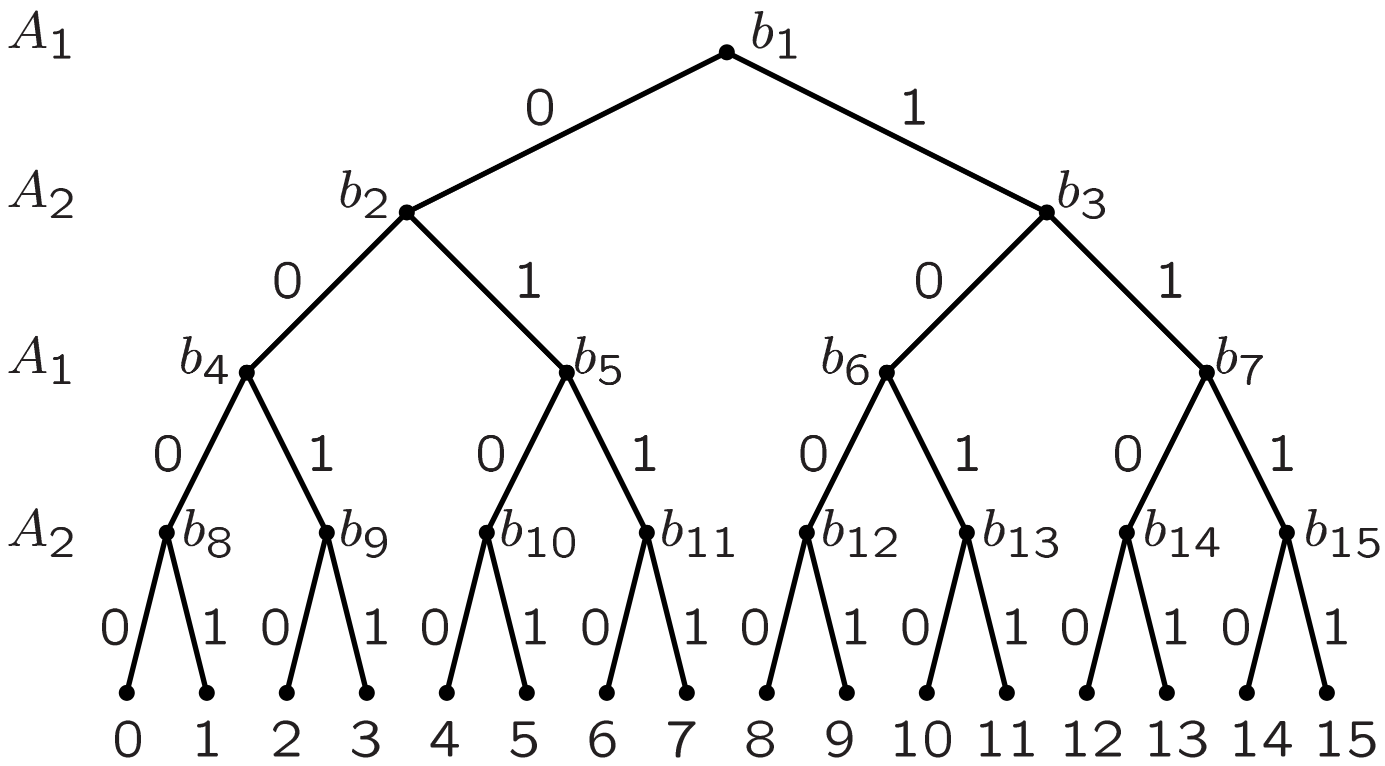
Universal Function (Protocol)

$$\begin{tikzcd}[column sep=huge] x_1, x_2 \arrow[r, "f"] \arrow[d, shift right=2, "g_1"'] \arrow[d, shift left=2, "g_2"] & f(x_1, x_2) \\ w_1, w_2 \arrow[r, "u"] & u(w_1, w_2) \arrow[u, "h"'] \end{tikzcd}$$

Computing function $f\colon X_1 \times X_2 \to Y$ of communication complexity n:

- preprocessing $g_i\colon X_i \to W_i$;
- computing universal function $u\colon W_1 \times W_2 \to U$ of communication complexity at most $2n$ (using universal protocol);
- postprocessing $h\colon U \to Y$.

An Example of Universal Function (Protocol)



$n = 2$, $W_1 = \{0,1\}^5$, $W_2 = \{0,1\}^{11}$, $U = \{0,1,2,\ldots,15\}$,
$w_1 = b_1b_4b_5b_6b_7$, $w_2 = b_2b_3b_8b_9b_{10}b_{11}b_{12}b_{13}b_{14}b_{15}$

Function *EQ*

For $|X| = n$ we define function $EQ_n\colon X \times X \to \{0, 1\}$ as

$$EQ_n(x_1, x_2) = \begin{cases} 1, & \text{if } x_1 = x_2; \\ 0, & \text{otherwise.} \end{cases}$$

Proposition 4 *If Paul has a winning strategy in the game* $\mathcal{G}\langle EQ_n, q, l\rangle$ *then he has a winning strategy in the game* $\mathcal{U}\langle n, q, l\rangle$.

This means that lower bound on the number of questions for Ulam's game holds for generalized game with *EQ* function.

Proposition 5 *If Paul has a winning strategy in the game* $\mathcal{U}\langle n, q, l\rangle$ *then he has a winning strategy in the game* $\mathcal{G}\langle EQ_n, q + 2l + 1, l\rangle$.

Stronger result with $\mathcal{G}\langle EQ_n, q + l + 1, l\rangle$ is expected.

Multi-party Communication

- We consider communication between k processors $A_1, A_2, \ldots, A_k$.
- Processors compute function $f\colon X_1 \times X_2 \times \ldots \times X_k \to Y$.
- At the beginning of communication processor A_i knows only its input $x_i \in X_i$.
- In each step of communication one processor broadcasts one bit to other processors.
- Communication ends when all processors know the value of function $f(x_1, x_2, \ldots, x_k)$.

Noise Model for Multi-party Communication

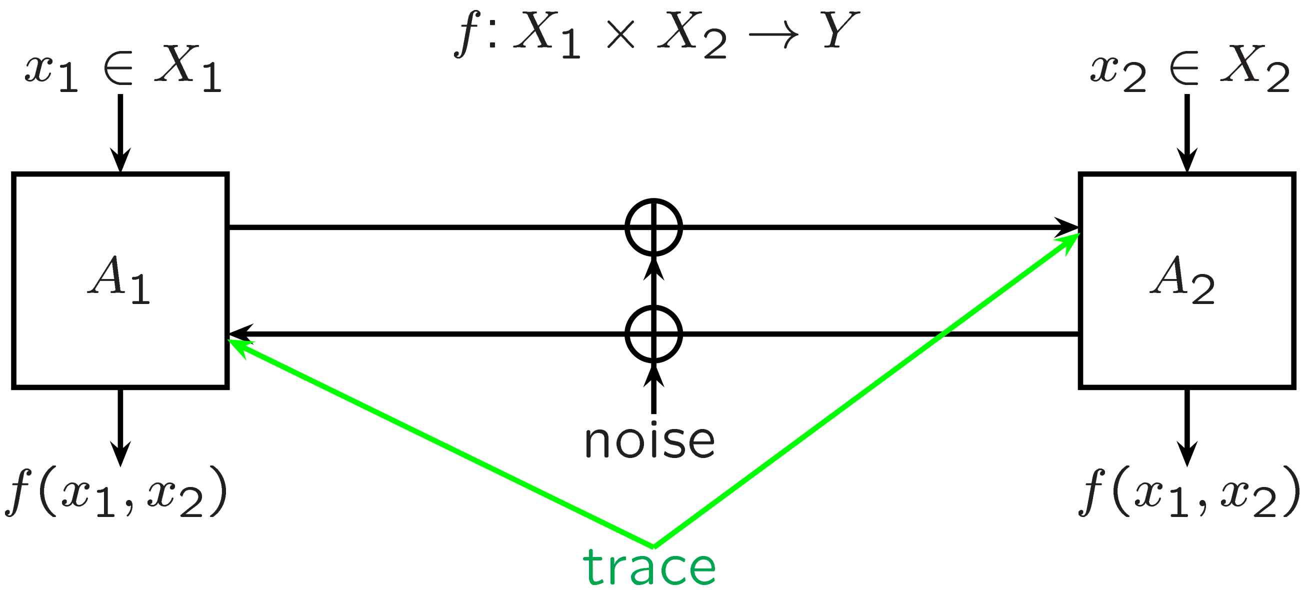
- Each broadcast is completely correct or not. There is at most l incorrectly broadcasts.

- In one broadcast only a part of processors receive message correct. One processor receive at most l incorrectly bits.

- At most l bits in whole communication is received incorrectly.

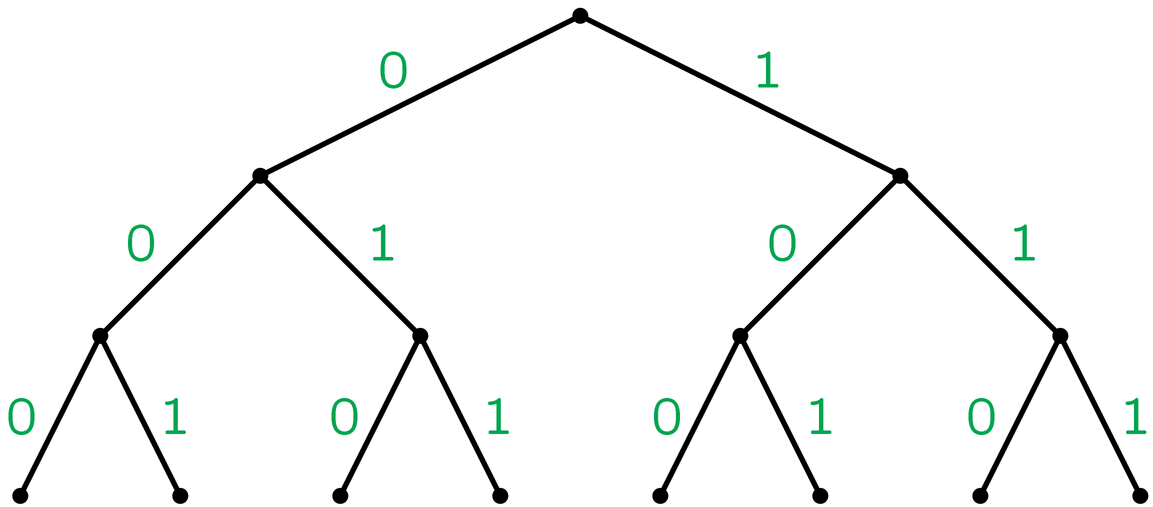
- Broadcasts are always correct. Processors are lying. Each processor can lie at most l times.

Proof of Proposition 1

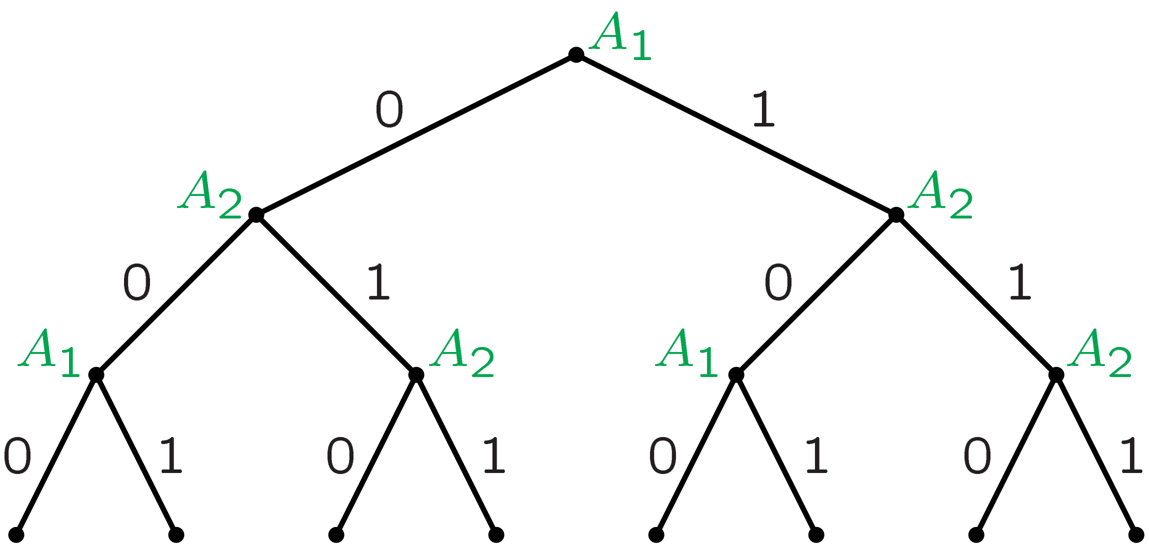


- Trace is a sequence of bits received by processors.

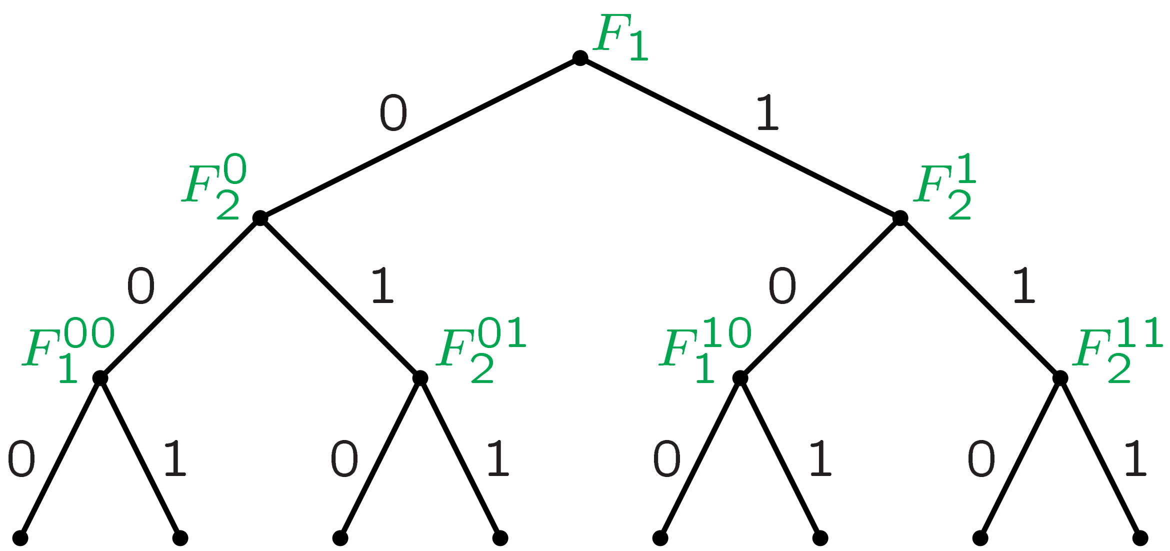
- All traces form binary tree.



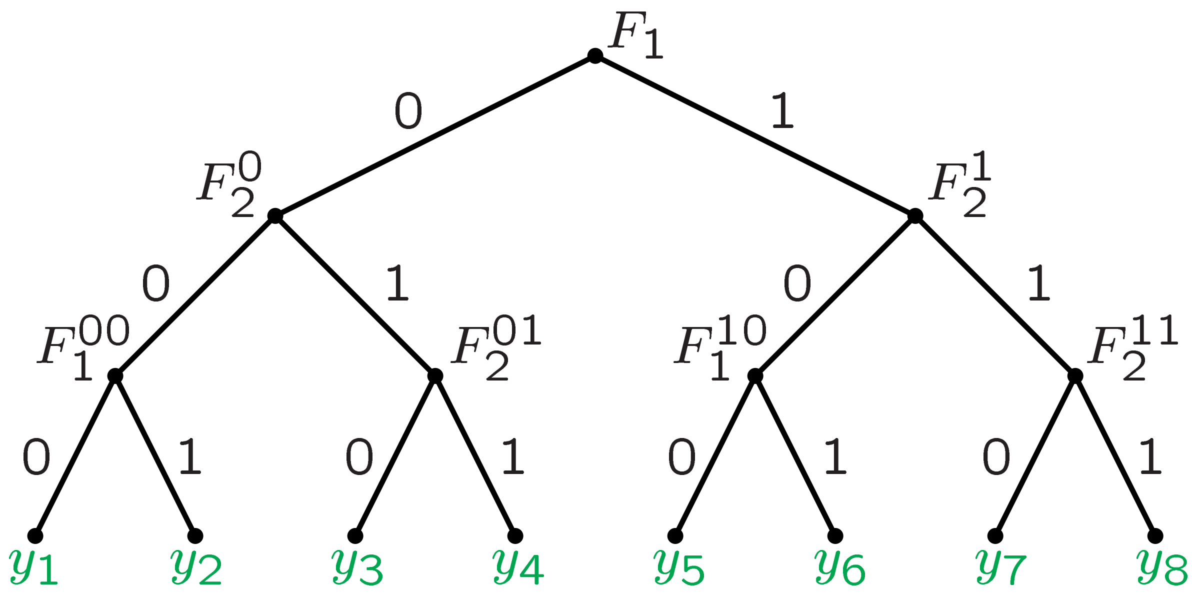
- Intern vertex determines processor which sends next bit.



- Each intern vertex v can be labeled with the set $F_i^v = \{x_i \in X_i :$ in vertex v processor A_i sends $1\}$.



- Each leaf can be labeled with the value of function f.



- Constructed tree describes the winning strategy for Paul.