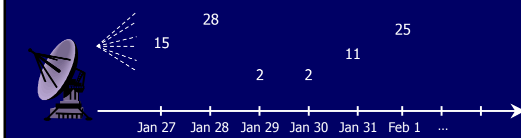
CS 259

Probabilistic Contract Signing

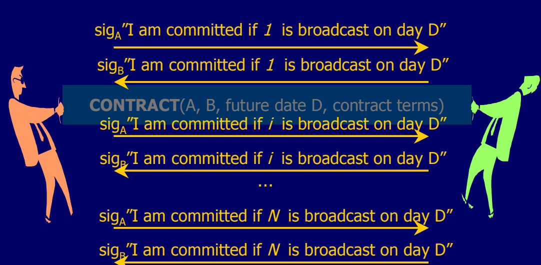
Vitaly Shmatikov

Rabin's Beacon

- A "beacon" is a trusted party that publicly broadcasts a randomly chosen number between 1 and N every day
  - Michael Rabin. "Transaction protection by beacons". Journal of Computer and System Sciences, Dec 1983.



Rabin's Contract Signing Protocol



2N messages are exchanged if both parties are honest

Probabilistic Fairness

- Suppose B stops after receiving A's i[th] message
  - B has sig_A"committed if *1* is broadcast",
    sig_A"committed if *2* is broadcast",
    …
    sig_A"committed if *i* is broadcast"
  - A has sig_B"committed if *1* is broadcast", …
    sig_B"committed if *i-1* is broadcast"
- … and beacon broadcasts number *b* on day D
  - If $b < i$, then both A and B are committed
  - If $b > i$, then neither A, nor B is committed
  - If $b = i$, then only A is committed — This happens only with probability 1/N

Properties of Rabin's Protocol

Fair
- The difference between A's probability to obtain B's commitment and B's probability to obtain A's commitment is at most 1/N
  - But communication overhead is 2N messages

Not optimistic
- Need input from third party in every transaction
  - Same input for all transactions on a given day sent out as a one-way broadcast. Maybe this is not so bad!

Not timely
- If one of the parties stops communicating, the other does not learn the outcome until day D

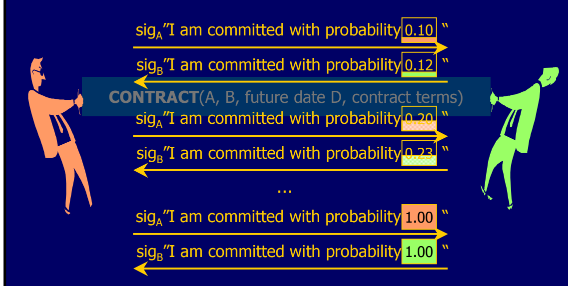
BGMR Probabilistic Contract Signing

[Ben-Or, Goldreich, Micali, Rivest '85-90]

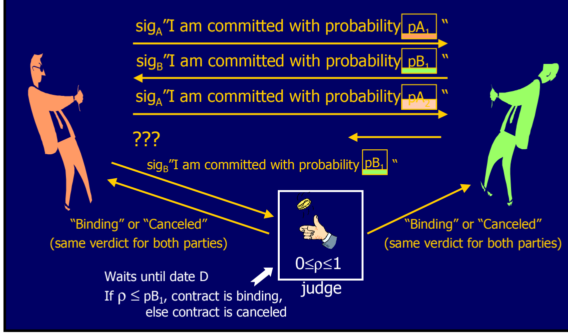
- Doesn't need beacon input in every transaction
- Uses sig_A"I am committed with probability p_A" instead of
  sig_A"I am committed if *i* is broadcast on day D"
- Each party decides how much to increase the probability at each step
  - A receives sig_B"I am committed with probability p_B" from B
  - Sets $p_A=\min(1, p_B \cdot \alpha)$ — α is a parameter chosen by A
  - Sends sig_A"I am committed with probability p_A" to B

… the algorithm for B is symmetric

BGMR Message Flow



Conflict Resolution



Judge

- Waits until date D to decide
- Announces verdict to both parties
- Tosses coin once for each contract
- Remembers previous coin tosses
  - Constant memory: use pseudo-random functions with a secret input to produce repeatable coin tosses for each contract
- Does not remember previous verdicts
  - Same coin toss combined with different evidence (signed message with a different probability value) may result in a different verdict

Privilege and Fairness

Privilege

A party is privileged if it has the evidence to cause the judge to declare contract binding

Intuition: the contract binds either both parties, or neither; what matters is the ability to make the contract binding

Fairness

At any step where Prob(B is privileged) > v, Prob(A is not privileged | B is privileged) < ε

Intuition: at each step, the parties should have comparable probabilities of causing the judge to declare contract binding (privilege must be symmetric)

Properties of BGMR Protocol

Fair
- Privilege is almost symmetric at each step:
  if Prob(B is privileged) > p_{A0}, then
  Prob(A is not privileged | B is privileged) < 1-1/α

Optimistic
- Two honest parties don't need to invoke a judge

Not timely
- Judge waits until day D to toss the coin
- What if the judge tosses the coin and announces the verdict as soon as he is invoked?

Formal Model

- Protocol should ensure fairness given any possible behavior by a dishonest participant
  - Contact judge although communication hasn't stopped
  - Contact judge more than once
  - Delay messages from judge to honest participant
- Need nondeterminism
  - To model dishonest participant's choice of actions
- Need probability
  - To model judge's coin tosses
- The model is a Markov decision process

Constructing the Model

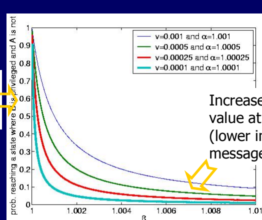
- Discretize probability space of coin tosses
  - The coin takes any of N values with equal probability
- Fix each party's "probability step" (Defines state space)
  - Rate of increases in the probability value contained in the party's messages determines how many messages are exchanged
- A state is unfair if privilege is asymmetric
  - Difference in evidence, <u>not</u> difference in commitments
- Compute probability of reaching an unfair state for different values of the parties' probability steps (Use PRISM)

Attack Strategy

- Dishonest B's probability of driving the protocol to an unfair state is maximized by this strategy:
  1. Contact judge as soon as first message from A arrives
  2. Judge tries to send verdict to A (the verdict is probably negative, since A's message contains a low probability value)
  3. B delays judge's verdicts sent to A
  4. B contacts judge again with each new message from A until a positive verdict is obtained
- This strategy only works in the timely protocol
  - In the original protocol, coin is not tossed and verdict is not announced until day D
- Conflict between optimism and timeliness

Analysis Results

Probability of reaching a state where B is privileged and A is not

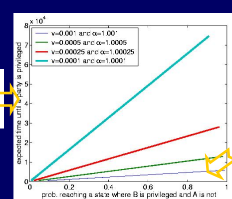


Increase in B's probability value at each step (lower increase means more messages must be exchanged)

For a higher probability of winning, dishonest B must exchange more messages with honest A

Attacker's Tradeoff

Expected number of messages before unfair state is reached



Probability of reaching a state where B is privileged and A is not

- Linear tradeoff for dishonest B between probability of winning and ability to delay judge's messages to A
- Without complete control of the communication network, B may settle for a lower probability of winning

Summary

- Probabilistic contract signing is a good testbed for probabilistic model checking techniques
  - Standard formal analysis techniques not applicable
  - Combination of nondeterminism and probability
  - Good for quantifying tradeoffs
- Probabilistic contract signing is subtle
  - Unfairness as asymmetric privilege
  - Optimism cannot be combined with timeliness, at least not in the obvious way