Rational Proofs with Multiple Provers

Jing Chen, Samuel McCauley, Shikha Singh
Department of Computer Science
What are *Proofs*?

- String of symbols that *certify* that a theorem is true
Interactive Proofs

- An all-powerful Merlin (Prover) interacts with a polynomial-time, probabilistic Arthur (Verifier)
- Merlin has to prove that a string $x$ is in a language $L$
Interactive Proofs

- If $x \in L$, then Arthur should accept with probability $1$
- If $x \not\in L$, Arthur should reject with probability at least $\frac{2}{3}$ (or $\frac{3}{4}$, 99%)
Interactive Proofs

- They exchange polynomial number of messages
- Language $L$ is in $\text{IP}$ if it has an interactive proof

Proof that $x \in L$

Is it true?
Interactive Proof for Graph Non-Isomorphism

Arthur

Merlin

Transcript

$G_1 \not\cong G_2$
Interactive Proof for Graph Non-Isomorphism

$G' \leftarrow G_1, G_2$

$G_1 \not\cong G_2$

$\pi(G')$

Arthur

Merlin

Transcript
**Interactive Proof for Graph Non-Isomorphism**

Arthur

$G' \leftarrow G_1, G_2$

$G_1 \neq G_2$

$\pi (G')$

$G_1$ or $G_2$

Merlin

Transcript
Interactive Proof for Graph Non-Isomorphism

If $G_1 \not\cong G_2$: Merlin is correct with probability 1
If $G_1 \cong G_2$: Merlin is caught with probability at least $\frac{1}{2}$
Interactive Proofs

- Merlin can be arbitrary: dishonest or malicious
- Arthur *verifies* Merlin's claim in polynomial time
- Protocols can be computationally intensive
Rational Interactive Proofs

• Arthur promises Merlin a reward for proving the theorem correctly
• Merlin is rational: he wants to *maximize* this reward
Rational Interactive Proofs

- Arthur computes the reward based on the transcript and his randomness
- Correctness is ensured by Merlin’s rationality!

How to pay to incentivize truthfulness?
Delegation of Computation

- Paying for services is a central concept in economics
- Computation is becoming a commodity
- Rational proofs are useful for computation outsourcing, cloud computing, etc.
Rational Interactive Proofs

- Simple and efficient alternative to interactive proofs
- Introduced by Azar and Micali (2012)
Rational Interactive Proofs

Arthur

Transcript

$ f(x) = y 

z$
Rational Interactive Proofs

Expected Reward based on transcript and coin flips

Arthur

Merlin
Power of Rational Proofs

- Constant-round RIP is more powerful than constant-round IP
- With polynomial rounds, \( \text{RIP} = \text{IP} \)
Rational Interactive Proofs

- Efficient rational proofs for delegation of computation:
  - Azar and Micali (2013)
  - Guo et al. (2014)
- All existing work involves a single rational prover
what if?

Arthur has **two** Merlins
Arthur has two Merlins
He can crosscheck their answers!
Arthur has two Merlins
He can crosscheck their answers!

In classical interactive proofs, two provers increase the power of the system

\[
\text{Multi-prover IP} = \text{NEXP} \quad \text{Babai et al. (1991)},
\]
\[
\text{IP} = \text{PSPACE} \quad \text{Shamir, Lund et al. (1992)}
\]
Arthur has two Merlins
He can crosscheck their answers!

“Are multiple Merlins more powerful than one in rational proofs?”

- Azar and Micali (2012)
We introduce: MRIP

*Multi-Prover Rational Interactive Proofs*
Multi-Prover Rational Interactive Proofs

• A way to delegate computation to multiple service providers

• A natural extension of RIP and MIP
MRIP: The Model

- Provers can pre-agree on a joint strategy
- They cannot communicate once the protocol begins
- At the end, the verifier computes a total reward
- Any strategy of the provers that maximizes the total reward leads the verifier to the correct answer
Warm Up: Naïve MRIP for \textbf{NEXP}

(Recall that MIP = NEXP)
Warm Up: Naïve MRIP for \( \text{NEXP} \)

\[ x \in L \text{ or } x \not\in L \]
Warm Up: Naïve MRIP for \( \text{NEXP} \)

\[ x \in L \text{ or } x \not\in L \]

If claim \( x \in L \)
Warm Up: Naïve MRIP for $\text{NEXP}$

If claim $x \in L$ then $Y$ leads to Accept.

$x \in L$ or $x \not\in L$
Warm Up: Naïve MRIP for NEXP

If claim $x \in L$  

$X \in L$ or $x \not\in L$  

Accept

MIP for NEXP
Warm Up: Naïve MRIP for **NEXP**

- If claim $x \in L$
  - Accept

$x \in L$ or $x \not\in L$

MIP for NEXP

- Acc
- Rej

End

$\$$
Warm Up: Naïve MRIP for NEXP

If claim $x \in L$

Y

Accept

N

Reject

$\$$

End

$x \in L$ or $x \not\in L$

MIP for NEXP

Acc

$\$$

End

Rej
Warm Up: Naïve MRIP for NEXP

Truth: $x \in L$

If claim $x \in L$

- Y: Accept
- N: Reject

MIP for NEXP

- Acc
- Rej

End

N

$$

End

$$
Warm Up: Naïve MRIP for NEXP

Truth: $x \in L$

If claim $x \in L$

Accept

Reject

MIP for NEXP

Prob = 1

End

End

$\$
Warm Up: Naïve MRIP for \textbf{NEXP}

Truth: \( x \not\in L \)

\( x \in L \) → Accept

\( x \not\in L \) → Reject

\( \text{Prob} \leq 1/3 \)

\( \text{Acc} \rightarrow \text{End} \)

\( \text{Rej} \rightarrow \text{End} \)

If claim \( x \in L \) → Accept

\( \text{Prob} \leq 1/3 \)
Warm Up: Naïve MRIP for NEXP

Truth: $x \not\in L$

If claim $x \in L$

Y → Accept

N → Reject

End

MIP for NEXP

Acc

Rej

End

$\$
Efficient MRIP for NEXP

- We have a naïve protocol using an MIP as a black box
- However, MIP protocols are often complicated, or computation and communication intensive
- We construct an efficient (i.e. linear) and simple MRIP protocol for NEXP
Efficient MRIP for NEXP

- Uses *Brier’s Scoring Rule* to solve an NEXP-complete problem.
Efficient MRIP for NEXP

• Uses *Brier’s Scoring Rule* to solve an NEXP-complete problem.

• Idea: ask one prover for a distribution, use second prover to help us sample from it
Is MRIP strictly more powerful?

- Recall:
  - MRIP contains MIP
  - However, with a single prover: RIP = IP
MRIP is Closed under Complement

- Consider a protocol for a language L
- A rational Merlin correctly reports $x \in L$ or $x \notin L$
- This protocol also works for $\overline{L}$
- MRIP contains NEXP, so MRIP also contains coNEXP
MRIP vs RIP and MIP

- If $\text{NEXP} \neq \text{coNEXP}$:
  - MRIP is more powerful than RIP
  - MRIP is more powerful than MIP

\[ \text{NEXP} = \text{MIP} \]
\[ \text{EXP} = \text{RIP} = \text{IP} = \text{PSPACE} \]
\[ \text{MRIP} \]
\[ \text{coNEXP} \]
Exactly How Powerful is MRIP?

Theorem: MRIP = EXP||NP

Exponential-time Turing Machine with non-adaptive access to an NP oracle
MRIP = $\text{EXP}^{\text{NP}}$ (proof sketch)

- Show MRIP = $\text{EXP}^{\text{\text{poly-NEXP}}}$ (equivalent class)
- Already know protocol for NEXP; another gets us EXP
MRIP = $\text{EXP}^{\text{NP}}$ (proof sketch)

- Issue is combining the rewards: need one reward that incentivizes truth in each protocol
- Idea is to scale payments in each round
When paying for (verifiable) computation, we can solve more difficult problems by employing multiple provers and cross-checking their answers!
When paying for (verifiable) computation, we can solve more difficult problems by employing multiple provers and cross-checking their answers!

This idea is already used in internet crowdsourcing applications.
Ask us questions separately and cross-check the results to get better answers
Number of provers and rounds

- In MIP only 2 provers and 1 round of communication is sufficient for any problem – Feige and Lovász (1992)
- Our MRIP for NEXP (=MIP) is 2 prover, 2 rounds
- Our EXP∥NP protocol has 4 provers and 5 rounds
- How many provers and rounds do we need to realize the full power of MRIP?
2 Provers and 3 Rounds are Sufficient

- Any MRIP with polynomial provers and polynomial rounds can be simulated by 2 provers in 3 rounds
- Three rounds of communication is almost optimal
- Is it possible to reduce it to just one round?
Utility Gap

• If provers lie, they receive exponentially small penalty
• What if they only tell the truth for a *substantial* gain?
• We call this gain the *utility gap*

I don’t get out of bed for less than $10,000 a day…
Utility Gap

- Polynomial gap: $P^{\text{||NEXP}}$
- Constant gap: Contains both NEXP and coNEXP
- The power of the system seems to reduce

Compare to EXP^{\text{||NP}} for MRIP with arbitrary gap
Conclusion

• How to use multiple rational provers optimally
• Two rational provers provide more power
• Only small number of provers and rounds required
• Requiring large utility gap seems to reduce power
Remember, you can ask more if you ask both of us separately and cross-check!

Thank You!