Polynomial-time targeted attacks on coin tossing for any number of corruptions

Omid Etesami **Ji Gao** Saeed Mahloujifar Mohammad Mahmoody

IPM
Institute for Research in Fundamental Sciences

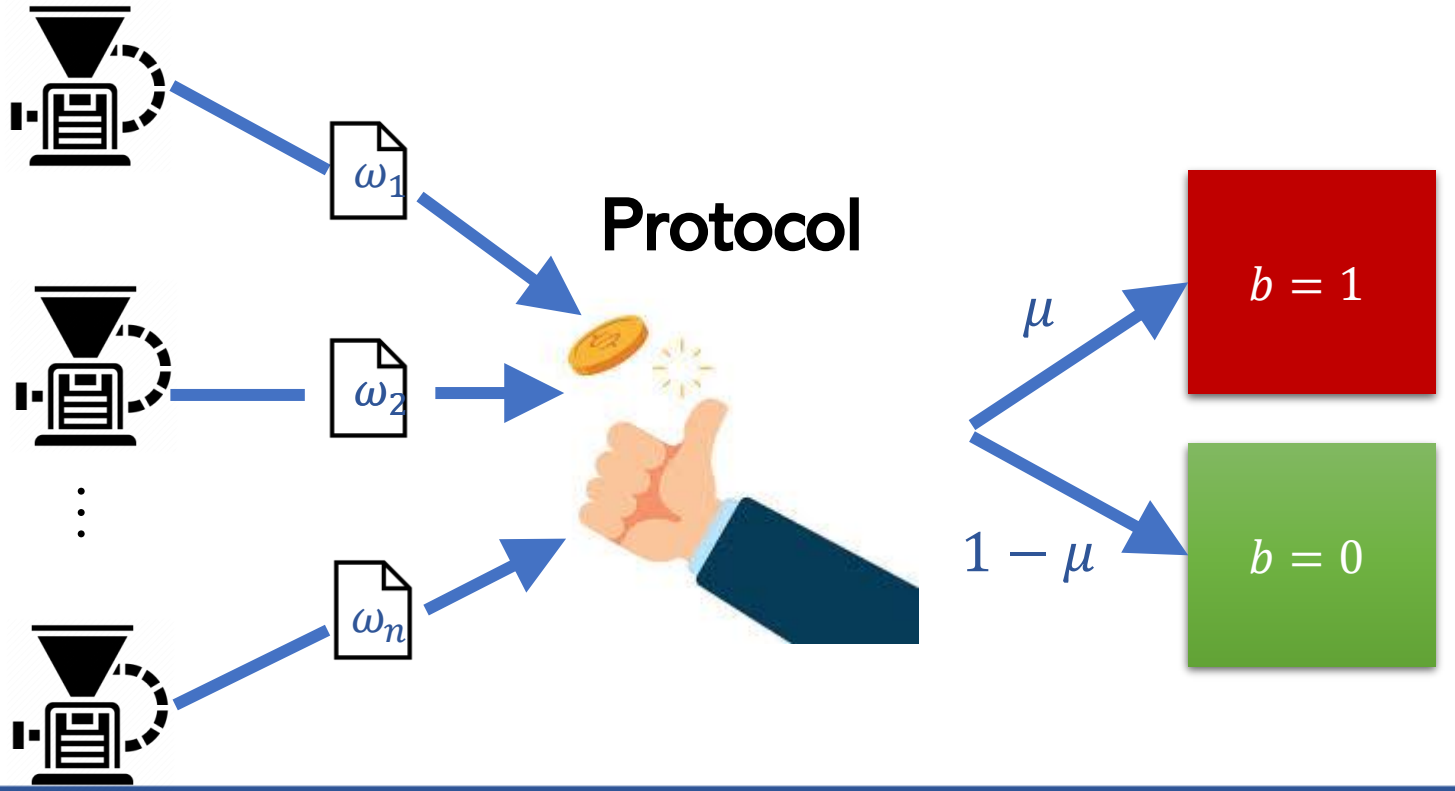
UNIVERSITY of VIRGINIA

PRINCETON UNIVERSITY

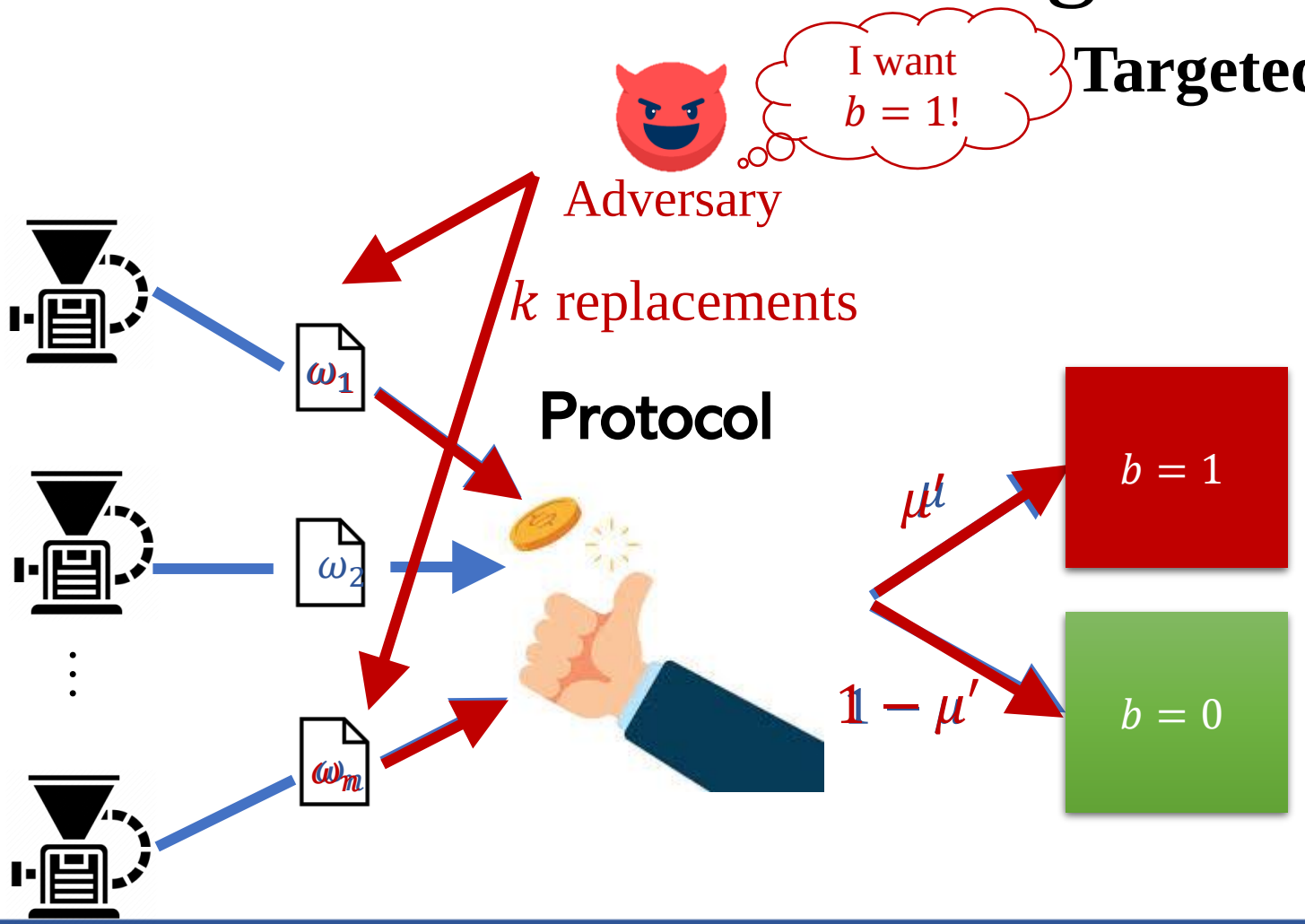
UNIVERSITY of VIRGINIA

Collective coin tossing

Collective coin tossing: n parties aim to jointly produce a random bit b.



Targeted attack on coin tossing



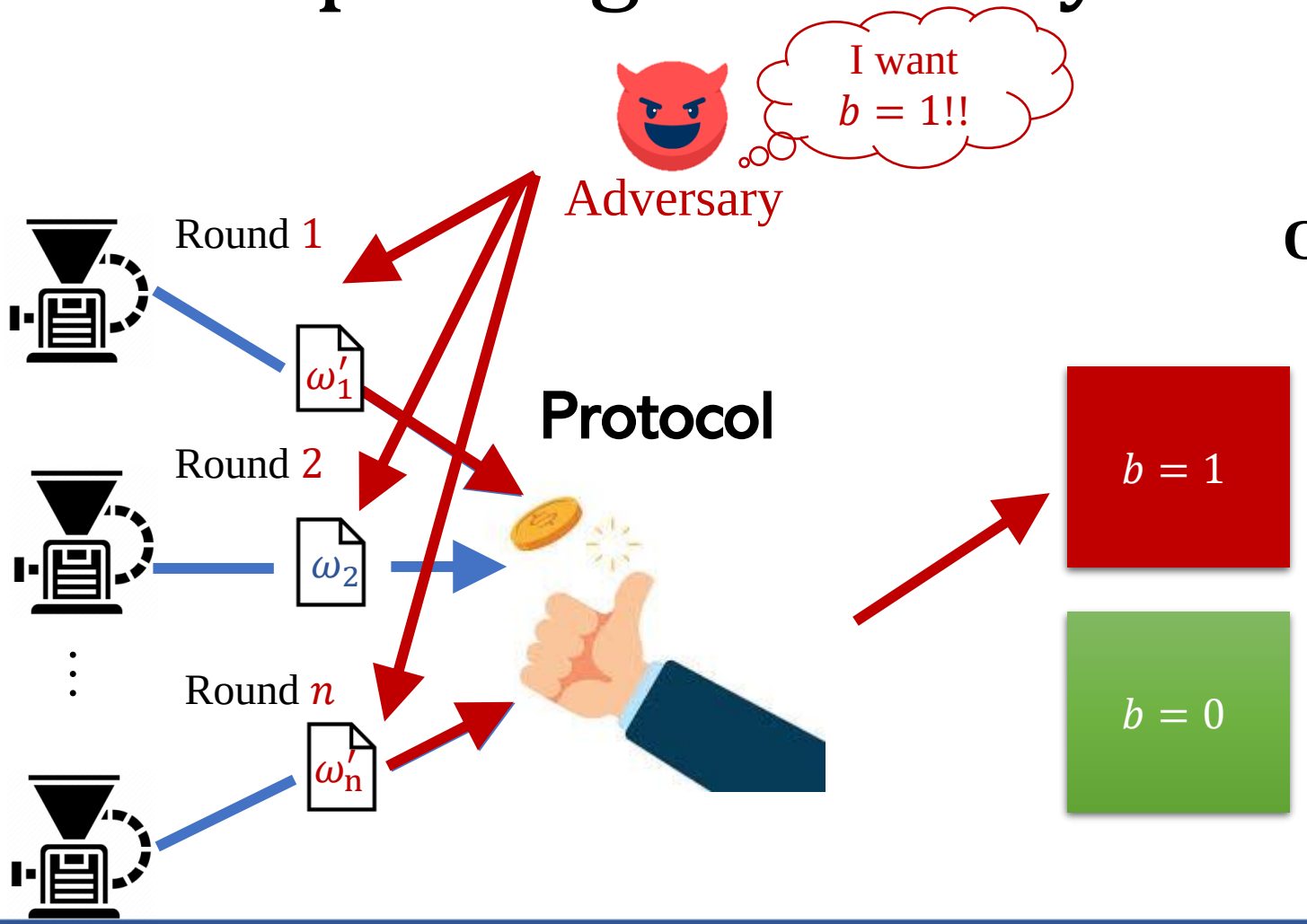
Formal definitions

- Protocol Π:
  - n parties $P_1 \dots P_n$. In round i, P_i sends ω_i (can depend on previous messages).
  - Protocol's outputs bit $b = f(\omega_1 \dots \omega_n)$.

- Targeted ***k*-replacing (strong adaptive[1])** adversary A:
  - Aims to increase $\Pr[b = 1]$.
  - Can override k of the messages $\omega_1 \dots \omega_n$ at its round:
    - In round i, given $\omega_1 \dots \omega_{i-1}$ **and $\boldsymbol{\omega_i}$**, the adversary can replace ω_i with ω_i'.

1. Shafi Goldwasser, Yael Tauman Kalai, and Sunoo Park. Adaptively secure coin-flipping, revisited. In International Colloquium on Automata, Languages, and Programming, pages 663–674. Springer, 2015. 2, 5, 11

Targeted k-replacing adversary



Online Adversary

Main question

- Recall: b' is the output under the attack and b is without attack.

$$\mu = \Pr[b = 1]$$

$$\mu' = \Pr[b' = 1]$$

Adversary's gain: $\mu' - \mu$

- Main question: assume we have a (n, μ)-protocol Π.
  With a fixed budget k, **how much gain** can the adversary achieve?
  - Messages are uniform binary ($\Pr[\omega_i = 0] = \Pr[\omega_i = 1] = 0.5$)
  - Messages are arbitrarily long.

Example: threshold protocol

- Threshold protocol: output follows threshold majority function

  $f(\omega_1, \dots, \omega_n) = 1$ iff $\sum_i \omega_i > t$.

  - It only uses uniform binary messages (random bits ω_i)!
  - All inputs that $\sum_i \omega_i > t$ shape a Hamming ball.
  - **Robustness :** With budget k, adversary is limited to $|\sum_i \omega'_i - \sum_i \omega_i| \leq k$.
    So, adversary can succeed if $\sum_i \omega_i > t - k$ (a bigger Hamming ball).
    (This holds even if adversary is "offline" and can do the changes at the end.)

- **Main question**: is this protocol **optimal**?
  What if it runs in **polynomial time**?

Our result

Recall: Threshold majority function: $f(\omega_1, \ldots, \omega_n) = 1$ iff $\sum_i \omega_i > t$.
Let $\beta_{\mathrm{n}}^{(t)} = \Pr[f(\omega_1, \ldots, \omega_n) = 1]$ over uniformly random bits.

- **Uniform binary messages**: Threshold protocol is optimal.
  - For $\mu = \beta_{\mathrm{n}}^{(t)}$ **poly-time online** attacks can achieve $\mu' = \beta_n^{(t-k)}$ **on any protocol**, which matches the majority's upper bound even for offline attacks

- **Any message length**: Threshold is optimal up to a constant for $\mu = \Omega(1)$
  - We have a **poly-time** attack that achieves $\mu' = \mu + \Omega(\mu k/\sqrt{n})$.

Outline

1. Related work and applications
2. Attack on any message length
3. Attack on uniform binary messages
4. Conclusion

Related work: Uniform binary messages (targeted attacks)

- [Lichtenstein et al. 1989][1] shows threshold functions are optimal under a **weaker adversary** model that cannot see the messages before making changes; for such attacks adversary can achieve $\mu' = \beta_{n-k}^{(t-k)} \ll \beta_n^{(t-k)}$.

- [Kalai et al. 2018][2] proposes a polynomial-version attack that works for $k = \Omega(\sqrt{n})$ which is optimal up to a constant when $\mu = \Omega(1)$.

1. David Lichtenstein, Nathan Linial, and Michael Saks. Some extremal problems arising fromdiscrete control processes.Combinatorica, 9(3):269–287, 1989
2. Yael Tauman Kalai, Ilan Komargodski, and Ran Raz. A lower bound for adaptively-securecollective coin-flipping protocols. In32nd International Symposium on Distributed Comput-ing, 2018.

Related work: Arbitrary message length (targeted attacks)

- [Mahloujifar-Mahmoody-ALT19][1] [Etesami et al.-SODA'2020][2] poly-time adversary with budget $k = \Omega(\sqrt{n})$ can increase the probability to approximately 1.

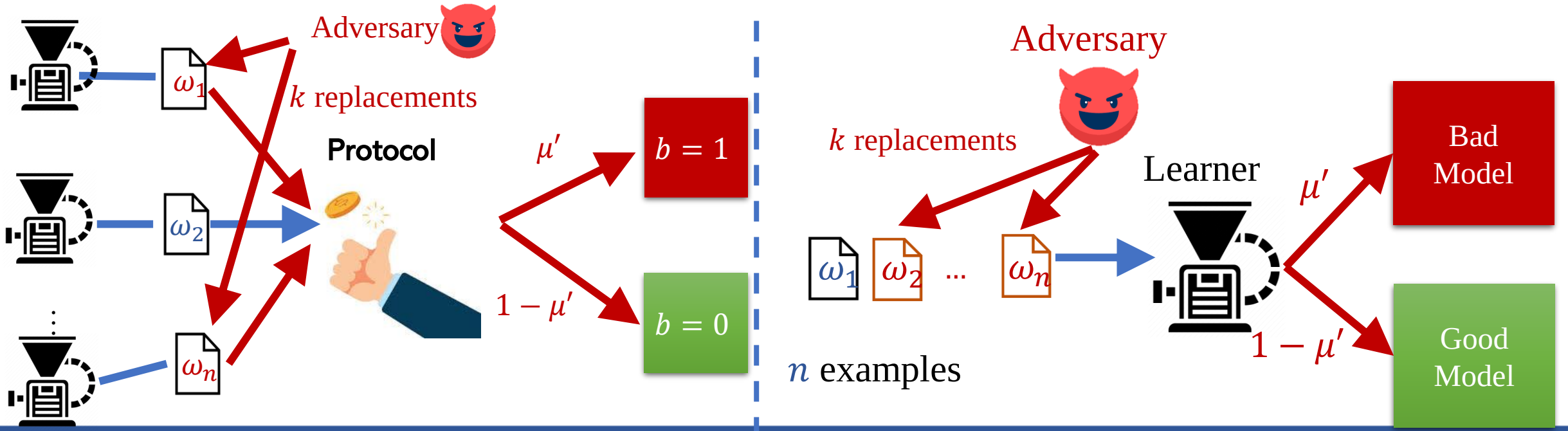
- We want to have a universal solution on every k!

1. Saeed Mahloujifar and Mohammad Mahmoody. Can adversarially robust learning leverage computational hardness? In Aurelien Garivier and Satyen Kale, editors, ´ Proceedings of the 30th International Conference on Algorithmic Learning Theory, volume 98 of Proceedings of Machine Learning Research, pages 581–609, Chicago, Illinois, 22–24 Mar 2019. PMLR.
2. Omid Etesami, Saeed Mahloujifar, and Mohammad Mahmoody. Computational concentration of measure: Optimal bounds, reductions, and more. In Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pages 345–363. SIAM, 2020.

Application 1: Targeted poisoning in machine learning

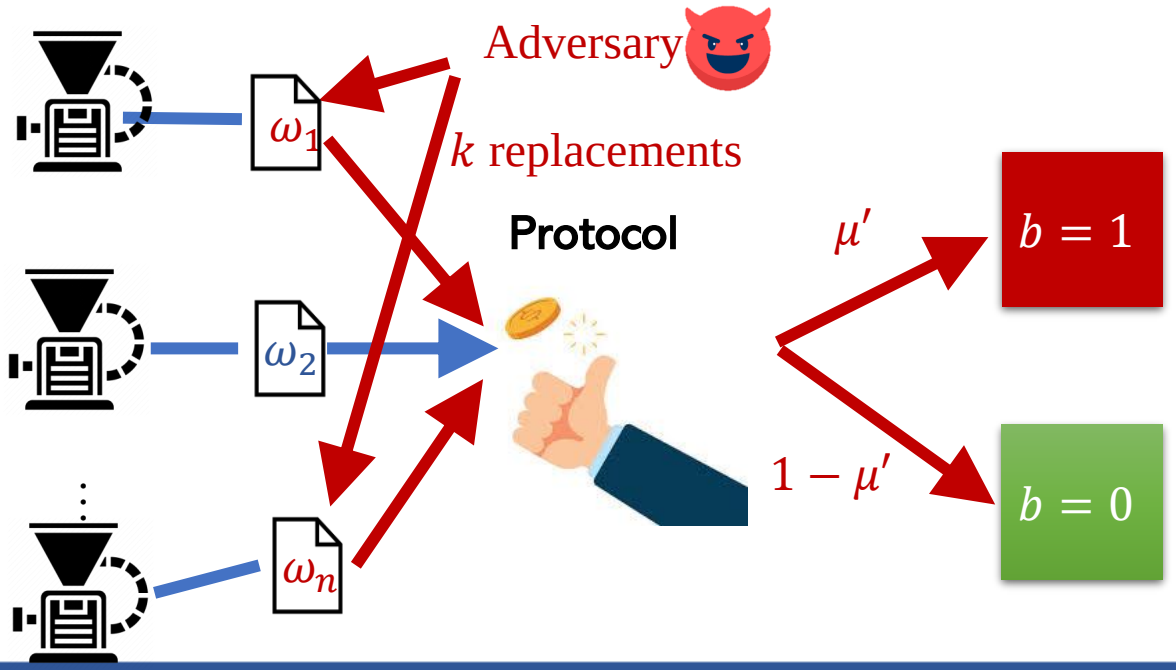
- Connection built by [Mahloujifar and Mahmoody, TCC 2017][1]:

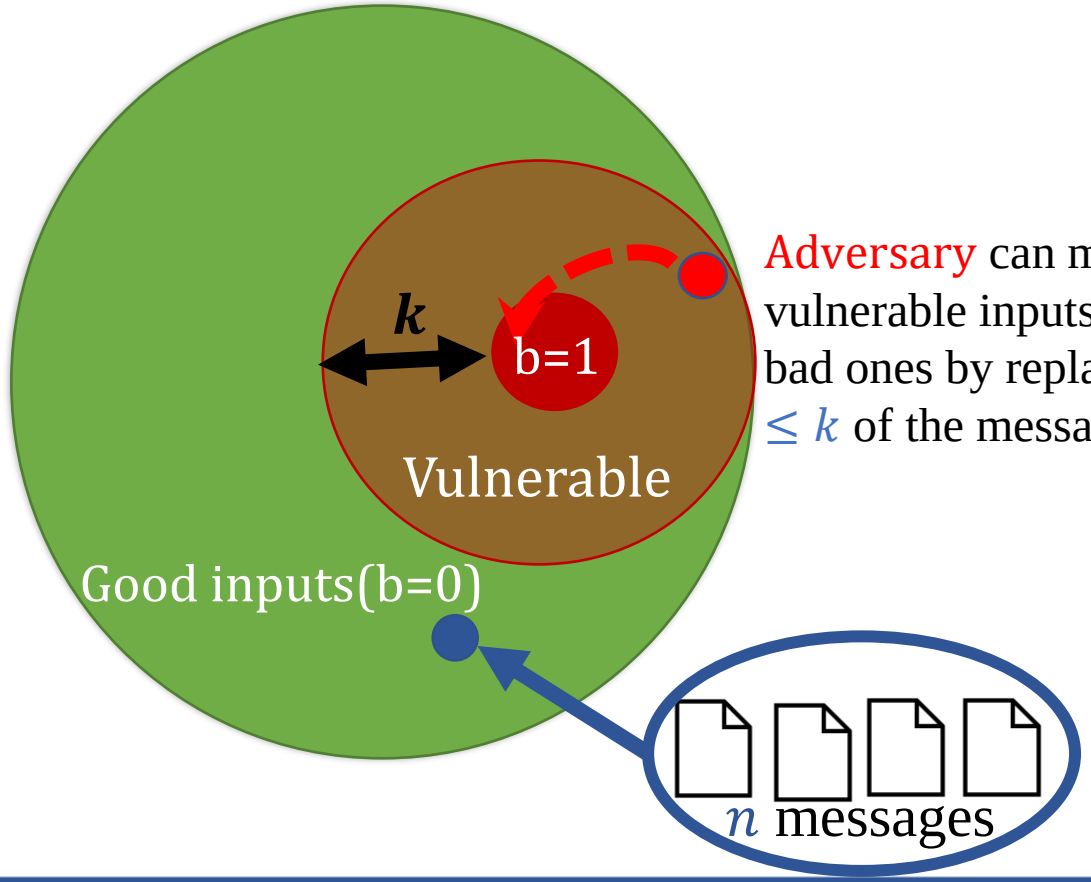
Targeted attack on coin-tossing $\xleftarrow{Reduced}$ Targeted poisoning attacks on machine learning



1. Saeed Mahloujifar and Mohammad Mahmoody. Blockwise p-tampering attacks on cryptographic primitives, extractors, and learners. In Theory of Cryptography Conference, pages 245–279. Springer, 2017.

Application 2: computational isoperimetry under Hamming distance in product spaces



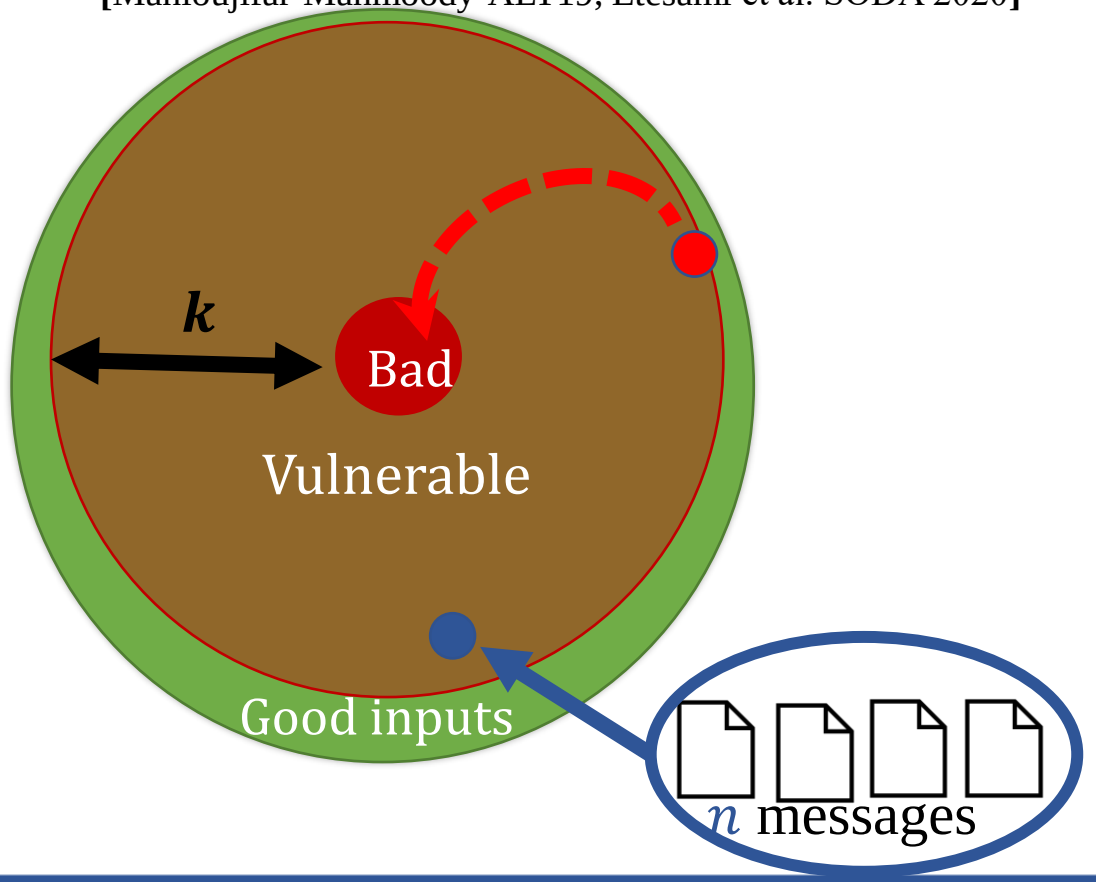


Adversary can map vulnerable inputs to bad ones by replacing $\leq k$ of the messages

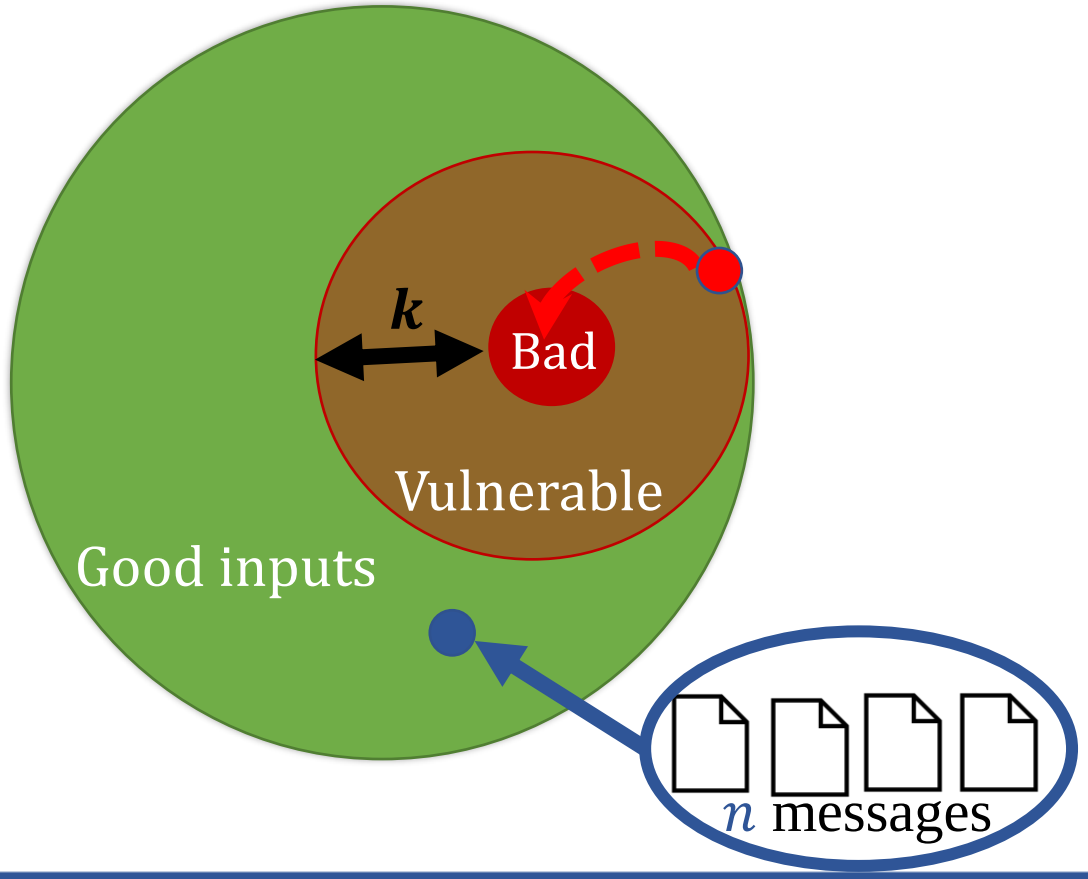
Computational **concentration** vs. computational **isoperimetry**

Computational concentration

[Mahloujifar-Mahmoody-ALT19, Etesami et al. SODA'2020][1][2]



Computational isoperimetry



1. Saeed Mahloujifar and Mohammad Mahmoody. Can adversarially robust learning leverage computational hardness? In Aurelien Garivier and Satyen Kale, editors, ´ Proceedings of the 30th International Conference on Algorithmic Learning Theory, volume 98 of Proceedings of Machine Learning Research, pages 581–609, Chicago, Illinois, 22–24 Mar 2019. PMLR.
2. Omid Etesami, Saeed Mahloujifar, and Mohammad Mahmoody. Computational concentration of measure: Optimal bounds, reductions, and more. In Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pages 345–363. SIAM, 2020.

Outline

1. Related work and applications
2. Attack on any message length
3. Attack on uniform binary messages
4. Conclusion

High level idea of our attack

- Adversary: Parameter λ and budget k
- Adversary's strategy: Make a replacement if (1) not already made k replacements and (2) the gain is at least λ.
- Namely:
  - $\alpha = \Pr[f(\omega_1, \dots, \omega_i, \omega_{i+1}, \dots) = 1]$ (over the randomness of remaining inputs)
  - $\alpha' = \max_{\omega_{i+1}^*}(\Pr[f(\omega_1, \dots, \omega_i, \omega_{i+1}^*, \dots) = 1])$
  - If $\alpha' - \alpha \geq \lambda$ and the adversary has not made k replacements yet, the adversary replaces ω_{i+1} to ω_{i+1}^*.

Connection to previous attacks

- At the high level, the attack is similar to the attack in [Mahloujifar-Mahmoody-ALT19][1], [Etesami et al.-SODA'2020][2] but with key differences:
- **Main difference**: These two papers use an analysis that **only applies to** $k = \Omega(\sqrt{n})$. We want to find a universal solution, especially for small $k = o(\sqrt{n})$
- **Syntactical differences:**
  - [Mahloujifar-Mahmoody-ALT19][1]: If $\bar{\alpha} - \alpha \geq \lambda$ (where $\bar{\alpha} = \Pr[f(\omega_1, \dots, \omega_i, \dots) = 1]$), it resets the input message. Otherwise, if $\alpha' - \alpha \geq \lambda$, replace ω_{i+1} to ω_{i+1}^*.
    Our attack has one fewer case and leads to a sharper bound even for $k = \Omega(\sqrt{n})$
  - [Etesami et al.-SODA'2020][2] uses the ratio α'/α instead of $\alpha' - \alpha$.
    This leads to a sharper bound than [MM19] and ours, **but only for $k = \Omega(\sqrt{n})$.**

1. Saeed Mahloujifar and Mohammad Mahmoody. Can adversarially robust learning leverage computational hardness? In Aurelien Garivier and Satyen Kale, editors, ´ Proceedings of the 30th International Conference on Algorithmic Learning Theory, volume 98 of Proceedings of Machine Learning Research, pages 581–609, Chicago, Illinois, 22–24 Mar 2019. PMLR.
2. Omid Etesami, Saeed Mahloujifar, and Mohammad Mahmoody. Computational concentration of measure: Optimal bounds, reductions, and more. In Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pages 345–363. SIAM, 2020.

Why previous analysis only works for large $k = \Omega(\sqrt{n})$

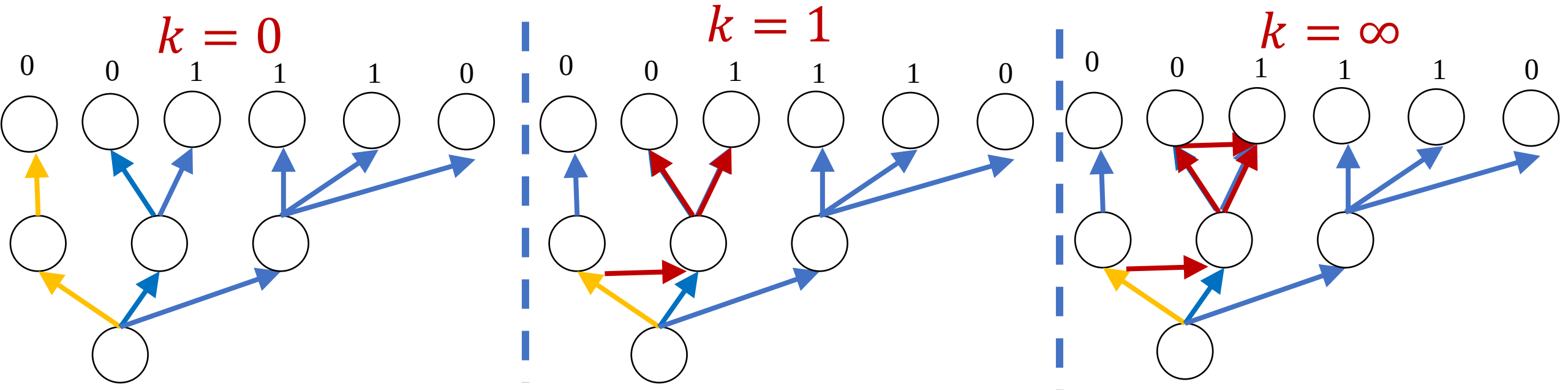
- [Kalai et al.-DISC'2018, Mahloujifar-Mahmoody-ALT19, Etesami et al.-SODA'2020] share a similar core which makes them all rely on $k = \Omega(\sqrt{n})$ budget.

- Their analysis goes through the following two steps:
  1. First, they show an attack with ***unlimited*** budget that can fix output to 1.
  2. Then by relying on (1) they show that this attack's budget is at most $\Theta(\sqrt{n})$.

- It can be shown that fixing the output to 1 could *require* budget $\approx \sqrt{n}$.

High level idea when $k = 1$: Case 1

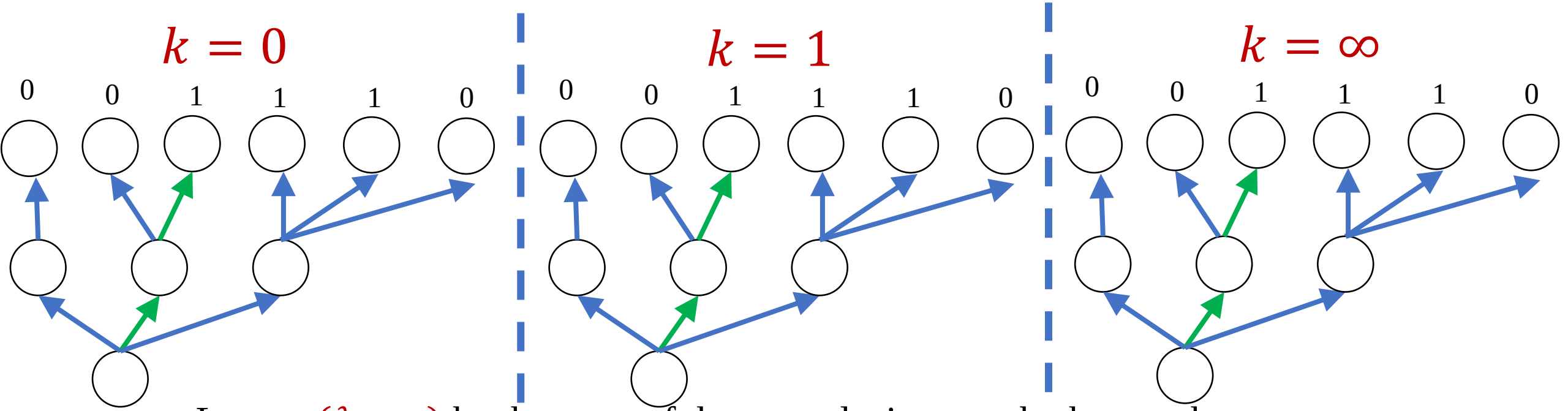
- Note that $k = 1$ is previously an open question!



- Let p_1 be the probability of the 1-replacing attack happens, and $\mu_1 = \mu'$ be the probability of $b = 1$ under the 1-replacing attack. Then we have

$$\mu_1 \geq \mu + p_1\lambda$$

High level idea when $k = 1$: Case 2



- Let $err(\lambda, \mu, n)$ be the error of the ∞-replacing attack, then we have

$$p_1 \geq 1 - \mu - err(\lambda, \mu, n)$$

- Combining with Case 1, we have $\mu_1 \geq \mu + \lambda(1 - \mu - err(\lambda, \mu, n))$
- Finally, we get **gain** $\mu_1 - \mu = \Omega(\mu/\sqrt{n})$ when let $\lambda = \Theta(\mu/\sqrt{n})$.

Extension to any budget

- Approach 1: Recursively apply the 1-replacing attack for k times.
  - Unfortunately, it is only polynomial time when $k = O(1)$.

- Approach 2: Directly analyze the k-replacing attack using induction.
  - Let the attack be the ∞-replacing attack that is cut after k replacements.
  - A generalization of the idea for 1-replacing attack shows:

$$\mu_k \geq \mu_{k-1} + \lambda(1 - \mu_{k-1} - err(\lambda, \mu, n))$$

  - Solving the recursion above gives the desired bound
  - The attack can be made poly-time using the same tricks as in Mahloujifar-Mahmoody-ALT19]

Outline

1. Related work and applications
2. Attack on any message length
3. Attack on uniform binary messages
4. Conclusion

Recall: Threshold functions

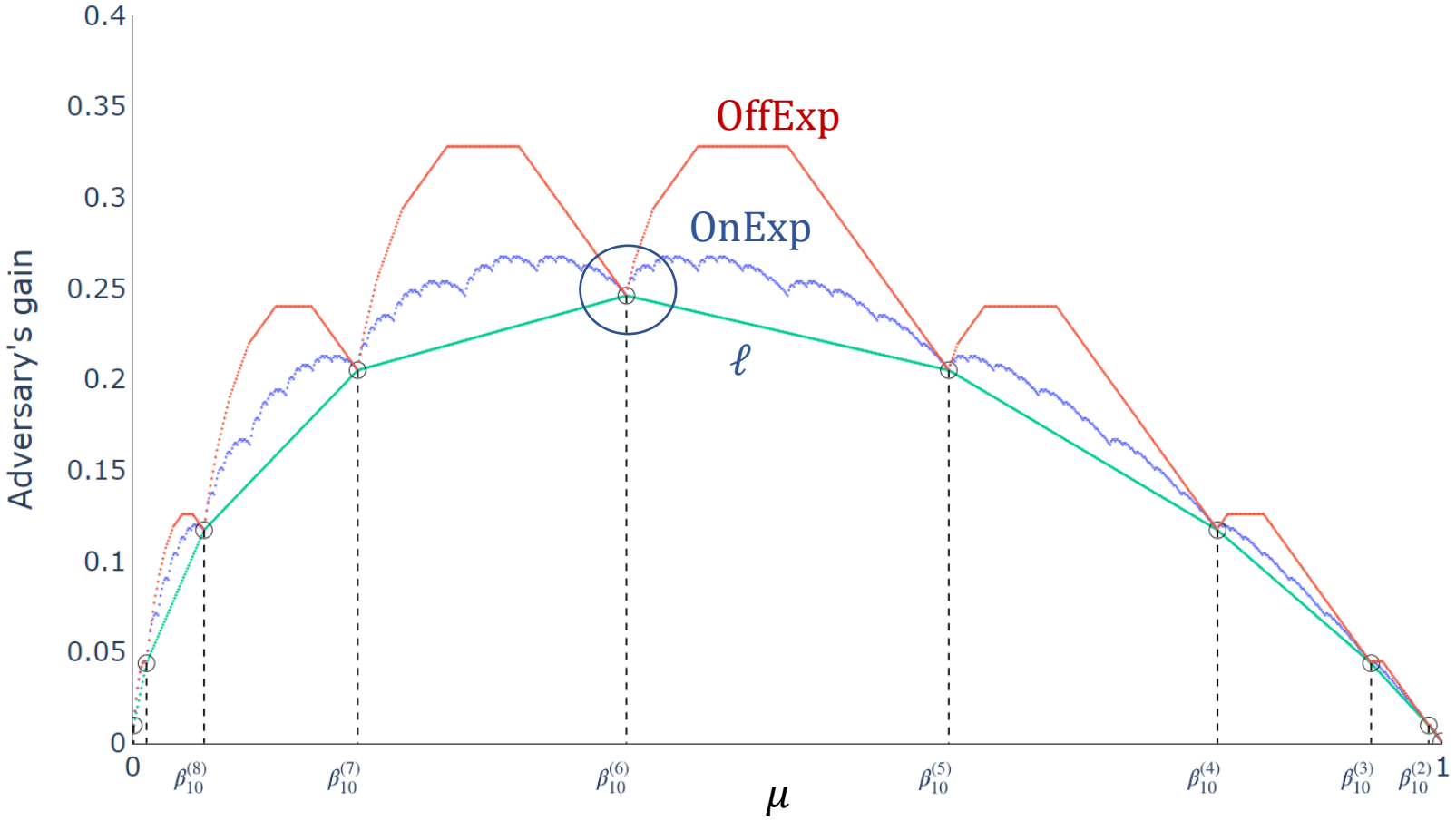
- Assume each ω_i is uniform binary.
- Threshold function $f(\omega_1, \ldots, \omega_n) = 1$ iff $\sum_i \omega_i > t$.
- Let $\beta^t = \Pr[\sum_i \omega_i > t]$.
- A k-replacing **offline** adversary on the t-threshold function can only achieve $\Pr[b = 1] \leq \beta^{t-k}$, which is probability of a hamming ball.
- Main question: How much gain can an **online** adversary achieve?

High level idea of the proof

- What we care is the **online-expansion function** $\mathrm{OnExp}_n^{(k)}(\mu)$ which is the optimal gain under the best possible k-replacing attack.
- $\mathrm{OnExp}_n^{(k)}(\mu)$ can be computed by induction on n.
- The corresponding optimal attack can also be implemented in polynomial time **if** one is given oracle access to the values of $\mathrm{OnExp}_n^{(k)}(\mu)$, but we do **not** have it!
- We prove a **piece-wise linear** (concave) lower bound for $\mathrm{OnExp}_n^{(k)}(\mu)$ as follows:
  - $\forall \beta_n^{(t)},\ \ell_n^{(k)}(\beta^t) = \mathrm{OffExp}_n^{(k)}(\beta^t)$,
  - $\ell_n^{(k)}(x)$ is linearly extended on all other points x.
- This piece-wise linear function is inspired by a similar function from [Khorasgani et al. 2021][1]). But the induction for proving the lower bound is quite different in our setting.

1. Hamidreza Amini Khorasgani, Hemanta K Maji, and Mingyuan Wang. Optimally-secure coin-tossing against a byzantine adversary. In 2021 IEEE International Symposium on Information Theory (ISIT).

High level idea of the proof (Contd.)



Polynomial version of the attack

- **Recall:** The optimal attack can be implemented in polynomial time **if** one is given oracle access to the values of $\mathrm{OnExp}_n^{(k)}(\mu)$. But we do **not** have access to $\mathrm{OnExp}_n^{(k)}(\mu)$ and do not know how to compute or even approximate it!

- To achieve polynomial-time attack, we define an adversary that **approximates and uses** $\ell_n^{(k)}$ instead of $\mathrm{OnExp}_n^{(k)}(\mu)$.

- The inductive proof still shows that using $\ell_n^{(k)}$ instead of $\mathrm{OnExp}_n^{(k)}(\mu)$ works!

Outline

1. Related work and applications
2. Attack on any message length
3. Attack on uniform binary messages
4. Conclusion

Conclusion

- For uniform binary protocol, threshold (majority) protocol is optimal for online and offline k-replacing attacks and for any k.
  - This result can be viewed as a computational version of Harper's isoperimetric inequality!
- For protocol with any message length, the majority protocol is still optimal up to a constant factor.
  - This result can be used to obtain generic targeted poisoning attacks on learners with small budget $k = o(\sqrt{n})$ where n is the size of the training set.

Thank you!

I appreciate any questions and comments from you!