# The Decisive Power of Indecision: Low-Variance Risk-Limiting Audits and Election Contestation via Marginal Mark Recording

Benjamin Fuller, Rashmi Pai, and Alexander Russell,
*University of Connecticut - Voting Technology Research Center*

https://www.usenix.org/conference/usenixsecurity24/presentation/fuller

This artifact appendix is included in the Artifact Appendices to the Proceedings of the 33rd USENIX Security Symposium and appends to the paper of the same name that appears in the Proceedings of the 33rd USENIX Security Symposium.

August 14–16, 2024 • Philadelphia, PA, USA

978-1-939133-44-1

Open access to the Artifact Appendices to the Proceedings of the 33rd USENIX Security Symposium is sponsored by USENIX.

# USENIX Security '24 Artifact Appendix: The Decisive Power of Indecision: Low-Variance Risk-Limiting Audits and Election Contestation via Marginal Mark Recording

Benjamin Fuller, Rashmi Pai, Alexander Russell
University of Connecticut – Voting Technology Research Center
{benjamin.fuller,rashmi.pai,alexander.russell}@uconn.edu

## A Artifact Appendix

### A.1 Abstract

This artifact provides a simulation of the stopping time of ballot comparison risk-limiting audits (RLAs). In particular, the work introduces two new types of RLAs that make marginal marks explicit in a cast vote record (CVR). These are called Bayesian and Conservative RLAs. In addition, to the new methods we compare against the Baseline method. All methods use Kaplan Markov as the statistical test.

### A.2 Description & Requirements

*The simulation is largely CPU bound and does not require multiple cores or have a large memory footprint. It was run on a 2019 Intel Mac. It requires Python≥ 3.9 and NumPy 1.20.3. The main statistic that we report on is the 95% of required ballots. As a result, we have to run a large number of simulations to get an accurate prediction. Our tests used* 5000 *simulations which took several days to run.*

#### A.2.1 Security, privacy, and ethical concerns

*Not applicable. The simulation simple distributes discrepancy to different ballots in an election. It uses town size data from the 2020 Connecticut Election but this data is publicly available.*

#### A.2.2 How to access

Access on Github `https://github.com/rpai0005/Questionable-Simulation-Tools/releases/tag/usenixae`.

#### A.2.3 Hardware dependencies

Evaluation used one CPU on a 2019 Intel Macbook Pro. No large memory requirement.

#### A.2.4 Software dependencies

Python≥ 3.9 and NumPy ≥1.20.3.

#### A.2.5 Benchmarks

Uses 2020 CT Town data which is included in repository.

### A.3 Set-up

git clone and as needed install python3 -m pip install NumPy=1.20.3. Change directory to the cloned repository directory.

#### A.3.1 Installation

#### A.3.2 Basic Test

To run a simple basic test open Questionable_Input.txt and change Simulations per margin=5000 to be Simulations per margin=11. Then run python3 Questionable_Simulation.py. This should take a few minutes to run (6 min on the 2019 Intel MacBook Pro) and will out a line for each set of simulation parameters. Once this completes change Questionable_Input.txt back Simulations per margin=5000.

### A.4 Evaluation workflow

You should now be ready to run the full evaluation. This runs 5000 simulations for each set of parameters. There are 3 margins, 3 types of audits, 11 probabilities of the mark being interpreted as full, and up to five different probabilities for each of these of what is recorded on the CVR. This means that the full run is ≈ 5000*3*3*11*5 ≈ 2.5 million simulations. Each of the 5000 simulations groups will be written on one row of the CVR. You may wish to redirect output by running python3 Questionable_Simulation.py > Simulation_Output.txt. These results were transcribed into Tables 2, 3, and 4. In particular, the qMath column indicates the approach used, 0 is the Baseline approach, 1 is the Bayesian approach, and 2 is the Conservative approach. q_CVR_rate is the rate at which marginal marks are counted as votes on the CVR. q_auditor_rate is the rate at which marginal marks are counted as votes by the audit board. Margin is the diluted margin ($\mu$ in Tables). Table 2 consists of the first nine rows $\mu = .01, .02, .03$ with q_CVR_rate and q_auditor_rate as .5. Table 3 is a summary of Table 4 which is the rest of the printed rows. In Table 3 q_CVR_rate = q_auditor_rate and is denoted as $p_m$. In Table 3, q_CVR_Rate is denoted as $p_{cvr}$ and q_auditor_rate is denoted as $p_{MAaudit}$.

### A.4.1 Major Claims

**(C1):** That both the Bayesian and Conservative methods reduce the 95% of sampled ballots in comparison to Baseline by $\approx 10\%$ using the Kaplan Markov statistical test. This is shown in Table 2 at a margin of $\mu = .01$.

### A.4.2 Experiments

All experiments are run using the call python3 Questionable_Simulation.py > Simulation_Output.txt. Expected run time on CPU equivalent to 2019 Intel MacBook Pro is 45 compute hours. Minimal harddisk or memory requirement. Note that the experiment is randomized so results may differ slightly.

## A.5 Version

Based on the LaTeX template for Artifact Evaluation V20231005. Submission, reviewing and badging methodology followed for the evaluation of this artifact can be found at https://secartifacts.github.io/usenixsec2024/.