# Impacts of Trust Measurements on the Reputation-Based Mining Paradigm

Pouya Pourtahmasbi and Mehrdad Nojoumian
*Department of Computer & Electrical Engineering and Computer Science*
*Florida Atlantic University*
Boca Raton, Florida, USA
*f*ppourtah,mnojoumian*g*@fau.edu

***Abstract*—Trust and reputation play an essential role in the success of the reputation-based mining paradigm, and therefore, it is important to have a mechanism for monitoring, analyzing, and labeling participants' behavior as a measurement of trust. To improve the existing reputation-based mining paradigm, we designed and implemented a trust model that takes into account real-world trust forming habits and incentivizes participants to commit to honest mining strategies in the cryptocurrency system. While detection of dishonest mining strategies can be challenging, this trust model considers past behavior and it is defection sensitive, i.e., making it more difficult to attain a high reputation value, the more one commits to dishonest mining. We also observed that the success of this trust model relies on the performance of attacks' detection, which is a reasonable observation. Our trust model can be used in cryptocurrency simulations to promote cooperation among miners and create a trustworthy environment for all participants.**

***Index Terms*—Blockchain; mining attacks; dishonest mining.**

## I. Introduction

The reputation-based mining paradigm is a cryptocurrency

In this article, we propose a similar trust model that is based on the same premise, but is specifically designed for the evaluation of trust inside the reputation-based mining scheme. The detection of dishonest mining activities could be a challenging effort [2], [11]. Therefore, it is likely that a significant portion of dishonest mining activities remain undetected. To address this issue, we propose a trust modeling procedure that is defection sensitive. This means that the negative impact of the defections outweighs the positive impact of cooperation. This setting will incentivize players to avoid defections if they are willing to stay in the system for a long period of time.

### *B. Our Approach*

Our trust model is designed to maintain the reputation history of player $p$ by updating and saving only a few parameters. In our method, a defection not only decreases the reputation value, but it also decreases the growth rate of the reputation if player $p$ cooperates in the future. In other words, when player $p$ increases the number of times he has defected, he will have to spend exponentially more time cooperating in order to compensate for the reputation loss. After a few defections, as player $p$ cooperates repeatedly and consecutively, the growth rate of his reputation will increase until it is restored to the original value. Even when the growth rate is restored to its original value, if player $p$ defects again, the reputation value as well as the growth rate will drop dramatically and further defections will exponentially cause more negative impact on the reputation of player $p$.

For the reputation-based cryptocurrency mining paradigm, If miner $m$ has a low hash power in conjunction with a negative reputation, miner $m$ will have a much lower chance of making a profit in the cryptocurrency system.

## II. Trust Model Protocols

Our trust model includes a set of step-wise procedures that are used for calculating the level of reputability for the player $p$. First the trust variable is calculated for player $p$, and then the reputation value is derived through a Sigmoid function that is bounded between $-1$ and 1, where $-1$ is the lowest and 1 is

## III. Effectiveness of Our Trust Model

We utilize the following strategies that are used by malicious players to disrupt a reputation system or decode its behavior.

1) Cooperate for some time to build a high reputation value and then defect on costly transactions.
2) Cooperate for some time and then take a mixed strategy of cooperation-&-defection to decode the behavior.
3) Take a mix strategy of cooperation-&-defection and then cooperate for some time to build a high reputation value.

The reputation value starts at zero then, the value changes after each round of game based on the behavior of the player. An increase in reputation value between round $i$ and $i + 1$ is the result of a cooperation and a decrease in reputation value between round $i$ and $i + 1$ is the result of a defection.

Figure 1 represents a player who cooperated for 15 rounds then defected three times in row. The decline of the reputation started slowly since the player was considered trustworthy. When the player defected for three consecutive iterations, a dramatic decline in the trust value occurred. After the decline of the reputation, the player continued by cooperating two times in row, however the reputation value did not significantly change. To compensate for the loss of trust, the player is required to cooperate for a significantly greater number of times than the number of times he cooperated in the beginning of the game. This is due to the fact that the player has shown to be untrustworthy after round 15.

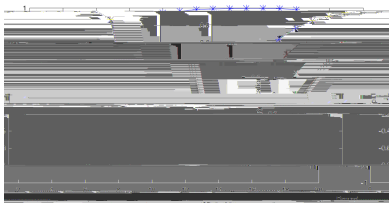

Fig. 1: Effectiveness of our model against the 1st strategy.

Another player, whose reputation progression is shown in Figure 2, also cooperated for a number of times in the beginning of the game but then eventually non-consecutively defected. The graph n

of rounds for simplicity. However, in our implementation, the value of $i$ represents the detection cycle that can contain any number of rounds within the probability distribution range. Therefore, the reputation value for all miners from pool $p$ will be updated by the pool manager once a new detection cycle takes place. The pool manager simply compares the number of actual proof-of-work (POW) against the number of expected POW for all member miners and then the reputation value for each miner is updated accordingly.

In other words, the pool manager compares the actual POW and the expected POW for all member miners for the period between the detection cycle $i - 1$ and $i$. If $x_j < E[x_j]$ and $x_i \geq CI$ for miner $M_j$, where $x_j$ denotes the actual POW, $E[x_j]$ denotes the expected POW since the last detection cycle for miner $M_j$, and $CI$ is the confidence interval for the attack detection, then the pool manager identifies miner $M_j$ as an attacker and updates the reputation of $M_j$ considering $M_j$ has defected in round $i$. Likewise, if the condition is false for miner $M_j$, then the pool manager updates the reputation of $M_j$ considering $M_j$ has cooperated in round $i$.

### *B. Performance in Simulation and Technical Discussion*

To observe the performance of our trust model in the reputation-based mining paradigm, we perform our simulation for 250,000 rounds of mining. The scatter plot in Figure 4 shows a summary of the performance of our trust model after round 250,000. In this plot, $x$ axis represents the reputation value and $y$ axis represents the total number of block withholding attacks. As we explained in the previous section, the detection method in our simulation is based on the confidence interval statistical test with the confidence of 98%. Therefore a percentage of attacks is expected to remain undetected.