A Process Algebraic Approach to Modeling and Performance Evaluation of Blockchain Fraud Reversal Protocols

Andrea Tirelli
joint work with Andrea Marin and Sabina Rossi

Ca' Foscari University of Venice, NiRvAna Final Meeting

May 30, 2025

Agenda

Context

- Public and private blockchains (digital Euro)
- Fraudulent transactions and smart contracts
- Fraud Reversal Protocols (FRP)
- Can we reverse a (possibly) fraudulent transaction?
- How to model and evaluate the performance of FRPs?

Relevant Questions

- Is there a **optimal time window** to ask for a refund?
- Is there a **cost** associated with the reversal?
- Is there a **trade-off** between the *productivity* of the blockchain and *fairness*, i.e. the possibility of reversing a transaction?

Components

Four main components:

- *Block states*: blockchain evolution throughout the relevant time window
- *User*: a user of the blockchain that is involved in a potentially fraudulent transaction
- *Judges*: a set of judges that can be involved in the transaction reversal process, deciding whether on the reversal (refund) request
- *Hacker*: a malicious user that tries to exploit the blockchain by creating fraudulent transactions

PEPA Specification - Block States

- $\boldsymbol{B_1} = (c_1, \gamma).B_2$
- $\boldsymbol{B_i} = (req, \top).W_i + (c_i, \gamma).B_{i+1}$ for $i = 2, \ldots, n-1$
- $\boldsymbol{B_n} = (req, \top).W_n + (c_n, \gamma).B_1$
- $W_i = (ign, \gamma).B_i + (reset, \delta).B_1$ for $i = 2, \ldots, n$

PEPA Specification - User, Hacker, Judges

- $\boldsymbol{User} = (c_1, \top).Victim + (c_n, \top).User$
- $Victim = (req, r).Req + (c_n, \top).User$
- $Req =$
  $(ign, \top).User + (refund, \top).Granted + (punish, \top).Resume$
- $Granted =$
  $(full, \top).Resume + (partial, \top).Resume + (none, \top).Resume$
- $Resume = (reset, \top).User$
- $\boldsymbol{Hacker} = (full, p\sigma).Hacker + (partial, (1-p)t\sigma).Hacker + (none, (1-p)(1-t)\sigma).Hacker$
- $\boldsymbol{Judges} = (ign, vs).Judges + (refund, (1-v)ws).Judges + (punish, (1-v)(1-w)s).Judges$

PEPA Specification - System Cooperation Equation

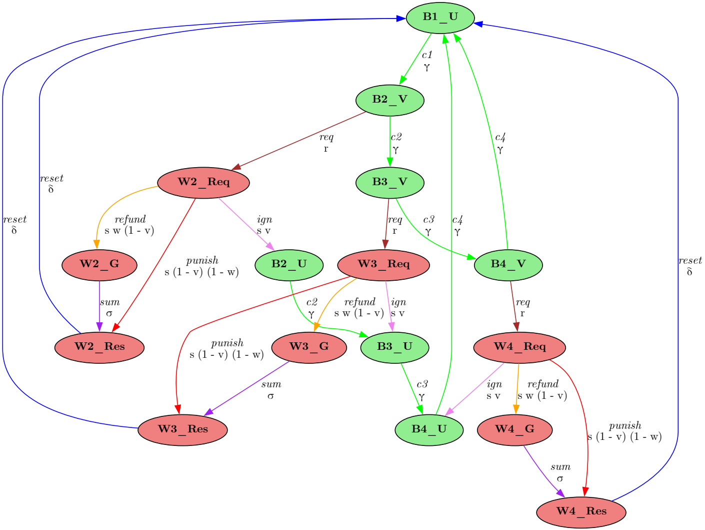
Denoting with

- $L_1 = \{c_1, c_n, reset, ign, req\}$
- $L_2 = \{ign, refund, punish\}$
- $L_3 = \{full, partial, none\}$

the system cooperation equation is:

$$\boldsymbol{B_1} \underset{\boldsymbol{L_1}}{\bowtie} \left(\boldsymbol{User} \underset{\boldsymbol{L_3}}{\bowtie} \boldsymbol{Hacker} \right) \underset{\boldsymbol{L_2}}{\bowtie} \boldsymbol{Judges}$$

Derivation Graph - $n = 4$



Utilization and Refund Probability

The network is *productive* when new blocks are created, i.e., when the counter is not in a waiting state. If $\mathcal{U} = \{User, Victim\}$ and $B_i^* = B_i \underset{L_1}{\bowtie} (* \underset{L_3}{\bowtie} Hacker) \underset{L_2}{\bowtie} Judges$

$$\boldsymbol{U_n} = \sum_{\boldsymbol{i}=1}^{\boldsymbol{n}} \sum_{*\in\mathcal{U}} \pi_{\boldsymbol{B_i^*}}$$

The *refund probability* is the probability of being in a state where the user is granted a refund. If $B_i^G = B_i \underset{L_1}{\bowtie} (Granted \underset{L_3}{\bowtie} Hacker) \underset{L_2}{\bowtie} Judges$

$$\boldsymbol{R_n} = \sum_{\boldsymbol{i}=1}^{\boldsymbol{n}} \pi_{\boldsymbol{B_i^G}}$$

Refund Cost

Accepting a refund request has a cost, which is directly proportional to the length of the time window between the fraudolent transaction and the refund request:

$$C_n = \frac{\alpha \sum_{i=1}^{n} i^e \pi_{B_i^G}}{\sum_{i=1}^{n} \pi_{B_i^G}}$$

One possible choice for α is the average number of transactions per block - this value can be estimated from empirical blockchain data or set according to the specific scenario being modeled.

Time Window Optimization

Optimal scenario:

- *maximum* utilization: blockain is as productvie as possible
- *maximum* refund probability: fairness condition in which fraudolent transaction get *reversed*
- *minimal* refund cost: minimal disruption to the blockain

We define the *Blockchain Efficiency-Fairness Index (BEFI)* as

$$\boldsymbol{BEFI_n = \frac{U_n R_n}{C_n}}$$

U_n and R_n have *opposite* trends $\rightarrow$ $BEFI_n$ captures the **trade-off** between these two indices and the cost of the refund mechanism.

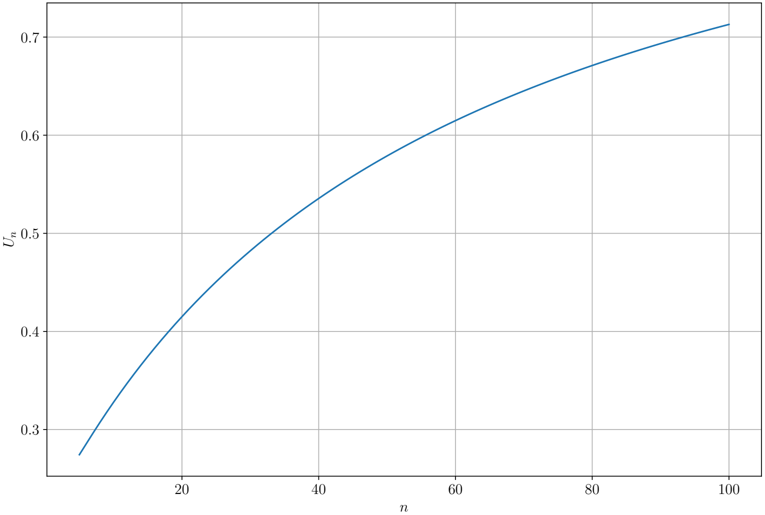
Experimental Setting

Scenario: introduce refund mechanism in Ethereum blockchain

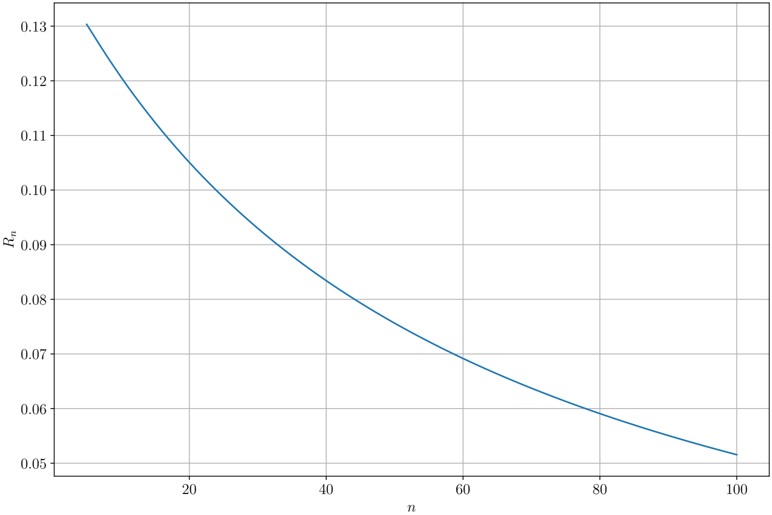
Parameters:

- block production rate $\gamma = \frac{1}{12s}$
- refund request rate $r \geq \gamma$
- request assessment rate (can lead to refund, punishment, dismissal with no processing) $s = \gamma$
- granted refund rate $\sigma = \gamma$ (need to write a block where the transaction is reversed)
- system reset rate $\beta = m\gamma$ for $1 < m < n$ (need to re-create all blocks after the fraudolent transaction)

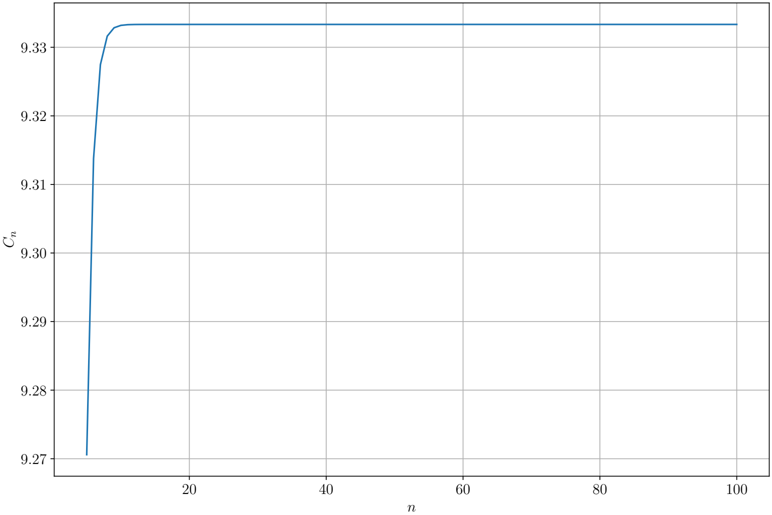
Utilization



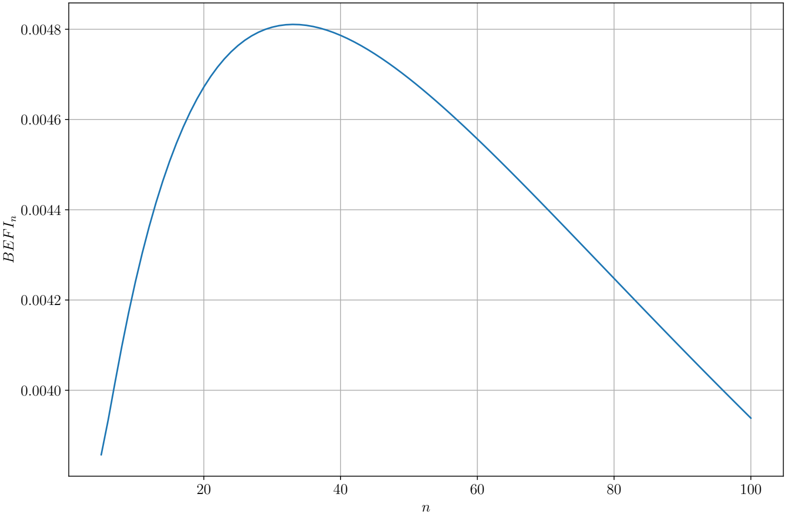
Refund Probability



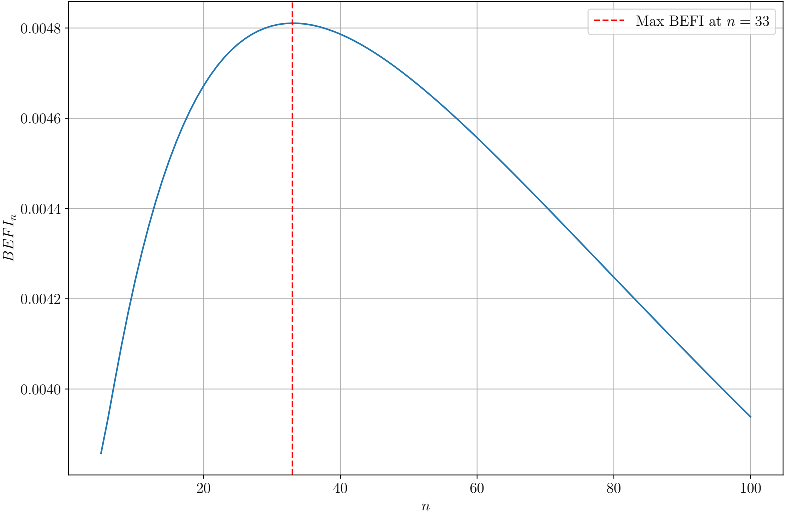
Total Refund Cost



BEFI



Trade-off Analysis



Summary

Assumptions:

- Simplified user behavior (e.g., single user, single hacker)
- Possibility to establish a pool of *Judges*.

Strengths:

- Agnostic model that can be applied to different blockchain scenarios (potentially even private blockchain)
- Model takes into account both productivity of the chain and the push for fairness and fraud remedy
- We acknlowledge the possibility that a refund request may come in too late (hacker has already spent all/part of the hacked sum)

Future Work

- Investigate protocol applicability in concrete scenarios, such as private blockchains
- Analyse empirical data to understand transaction dependence and estimate spending speed
- Extend the model to capture fraudolent spending speed more precisely