

Game Theoretical Analysis of Resource Allocation in the InterPlanetary File System

David Grisham

2 May 2018

Background

IPFS (InterPlanetary File System)

- P2P hypermedia distribution protocol
 - **Goal:** Replace HTTP, decentralize Internet
- Content-addressed, versioned filesystem
- Git repo in a torrent

IPFS Stack



Figure 1: The IPFS Stack

- IPFS's block exchange protocol
- Inspired by BitTorrent
- *Given a set of peers who want data, how to allocate resources?*

Given a set of peers who want data, how to allocate resources?

- Every user maintains reputation for each peer
 - Very complex dynamics
- **Reciprocation function**

Objectives

- Discover Bitswap reciprocation function(s) that gives desired behavior
 - Will depend on conditions
 - Break down Bitswap dynamics
- Analytical and empirical analyses
- Implementation

System Model

IPFS Network as Graph

- *Nodes*: Users $\in \mathcal{N}$
- *Edges*: Peerings; unweighted, undirected
 - *i's neighborhood*: $\mathcal{N}_i \subseteq \mathcal{N}$

User i distributes B_i bits among peers in each round

- b_{ji}^t : Total bits sent from user j to peer i from round 0 to $t - 1$
- d_{ji}^t : *debt ratio* of j as viewed by i in round t
 - Used as peer-wise reputation

$$d_{ji}^t = \frac{b_{ji}^t}{b_{ij}^t}$$

Reciprocation Function

- *Inputs*: Peer debt ratio, rest of peers' debt ratios
- *Output*: Peer weight
- $S_j(d_{ji}^t, \mathbf{d}_j^{-i,t}) \in [0, 1]$
 - e.g. $S_j(d_{ji}^t, \mathbf{d}_j^{-i,t}) = \frac{d_{ji}^t}{d_{ji}^t + \sum_{d_{jk}^t \in \mathbf{d}_j^{-i,t}} d_{jk}^t}$

B_j bits distributed among peers via weighted round-robin

$$b_{ji}^{t+1} = b_{ji}^t + S_j(d_{ji}^t, \mathbf{d}_j^{-i,t}) \times B_j$$

Game Formulation

- *Players*: Users/nodes
- *Strategy*: Reciprocation function
- *Utility*: $U_i = \sum_{j \in \mathcal{N}_i} b_{ji}^\infty = \sum_{t=0}^{\infty} u_i^t$

$$u_i^t = \sum_{t=0}^{\infty} \sum_{j \in \mathcal{N}_i} (b_{ji}^t - b_{ji}^{t-1})$$

- *Complexity vs. accuracy*
- Attempted formulations
 - Evolutionary game theory
 - Statistical mechanics
 - **Repeated games**

Game Characteristics

- *Infinitely repeated*
 - Discrete rounds, denoted by t
- *Incomplete information*

Objectives

- Classify Bitswap reciprocation functions
 - Conditions where useful
- **Analytical work:** Repeated game model
- **Empirical work:** Simulations
- **Implementation:** `go-ipfs`, IPTB

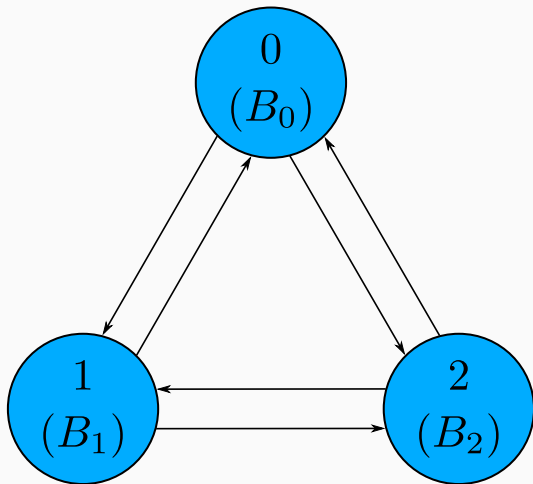
Preliminary Results

- 3 node network
- Parameters
 - Resource distribution
 - Initial peer-wise reputations
- Tests whether given reciprocation function is a Nash equilibrium (NE)

User 0:

1. Follows reciprocation function
2. Deviates from reciprocation function

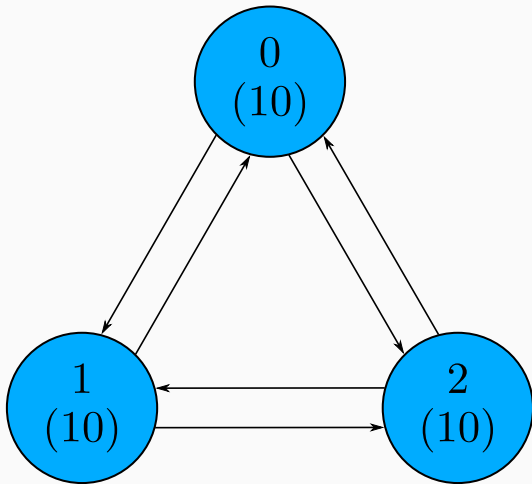
3-Node Network



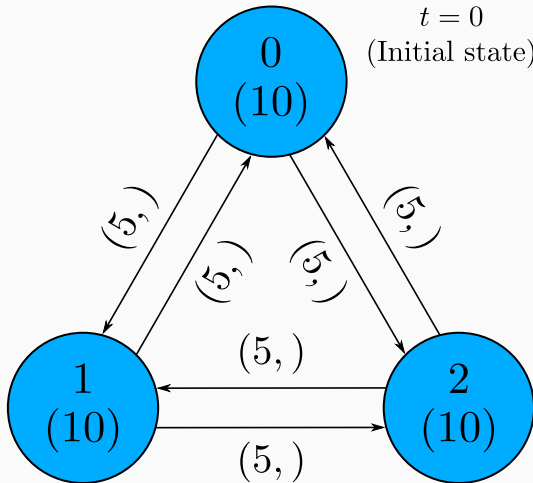
Example 1

- **Reciprocation function:** Linear
- **Initial ledgers:** Split
- **Resource distribution:** $[10, 10, 10]$

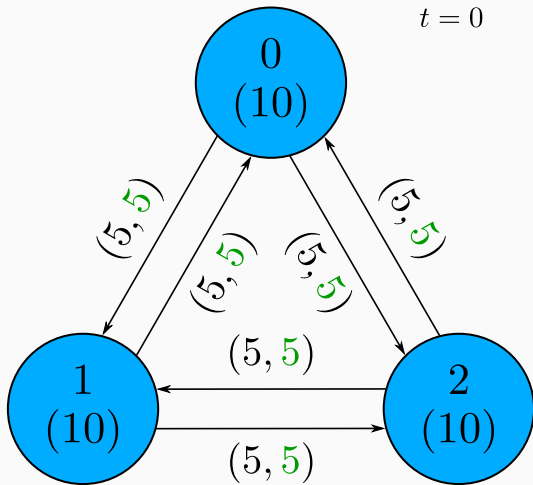
Example 1 – Non-Deviating



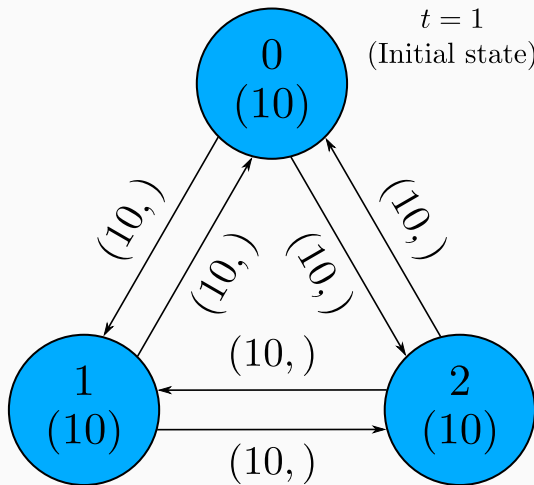
Example 1 – Non-Deviating



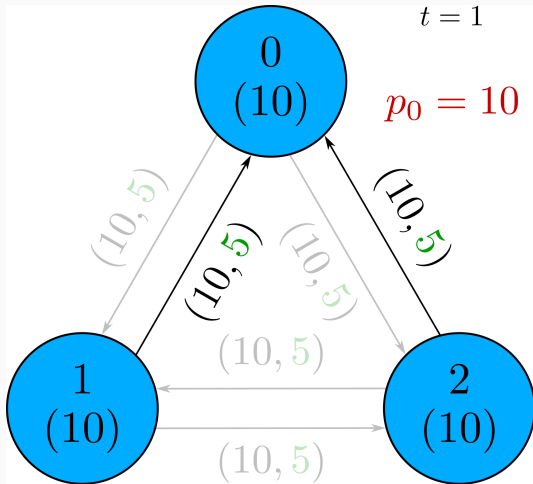
Example 1 – Non-Deviating



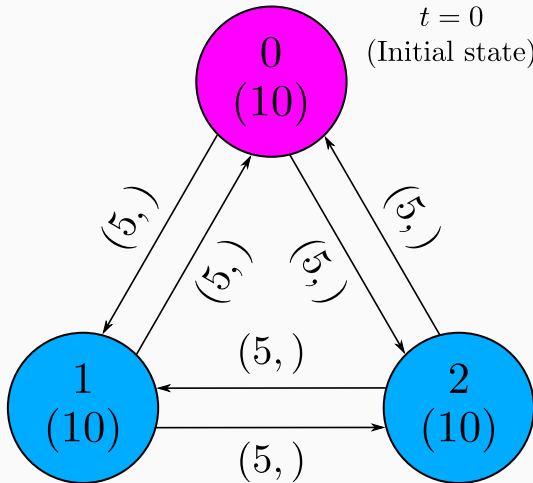
Example 1 – Non-Deviating



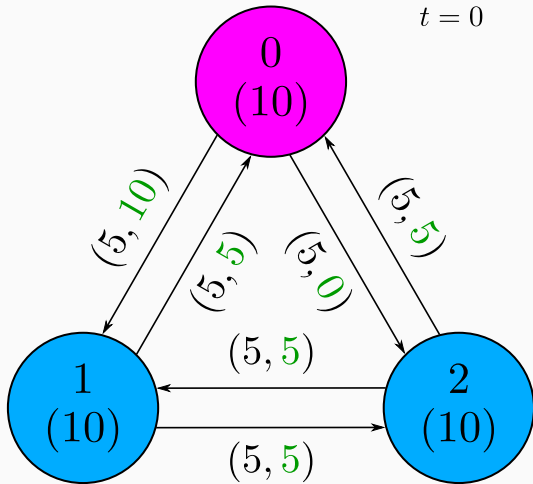
Example 1 – Non-Deviating



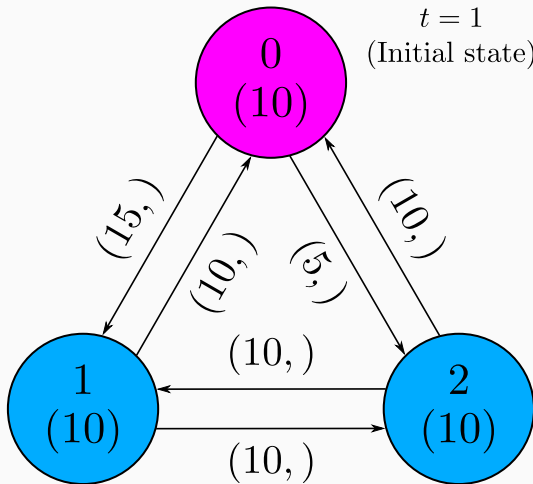
Example 1 – Deviating



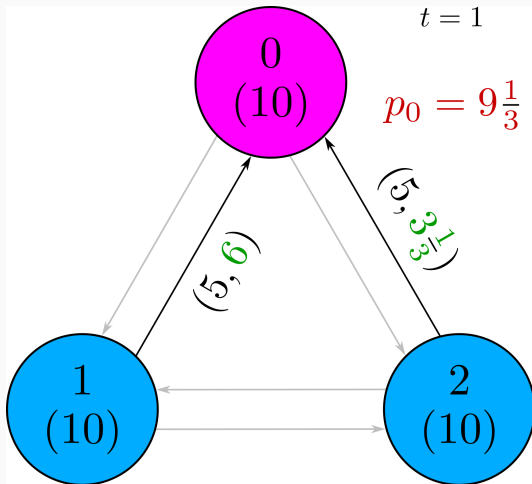
Example 1 – Deviating



Example 1 – Deviating



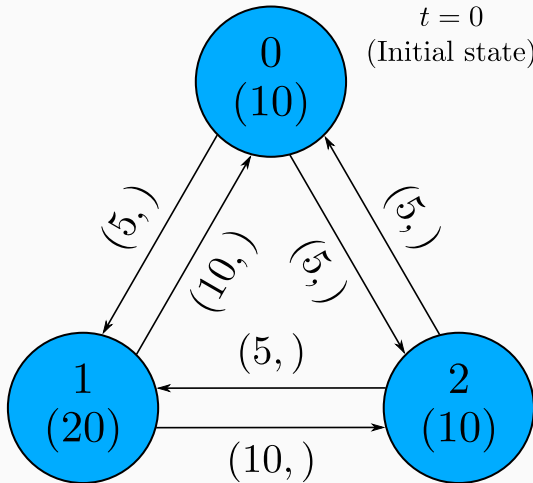
Example 1 – Deviating



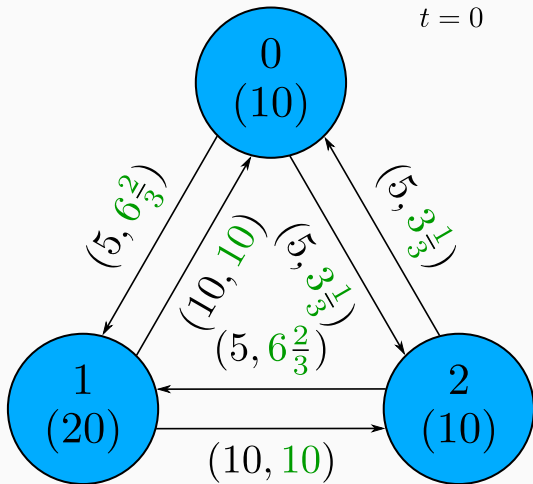
Example 2

- **Reciprocation function:** Linear
- **Initial ledgers:** Split
- **Resource distribution:** $[10, 20, 10]$

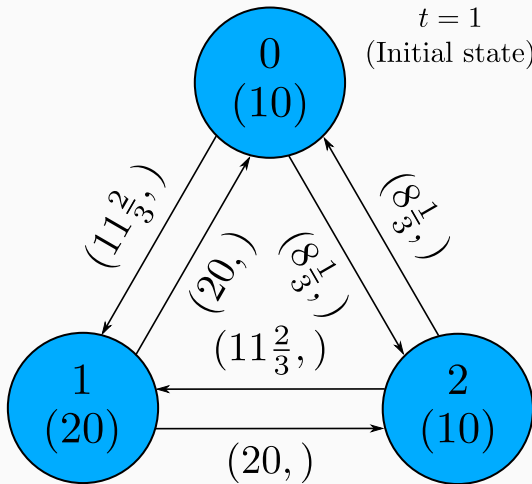
Example 2 – Non-Deviating



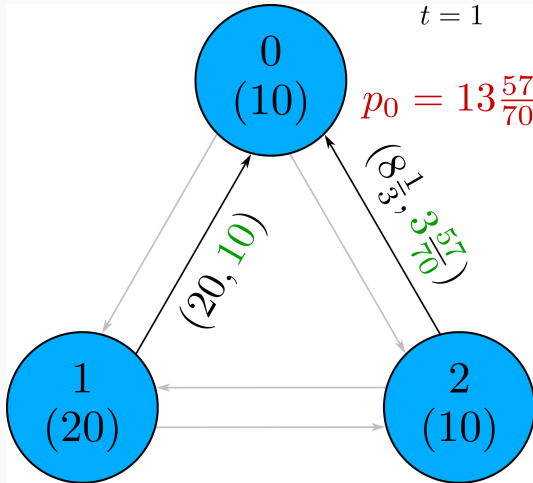
Example 2 – Non-Deviating



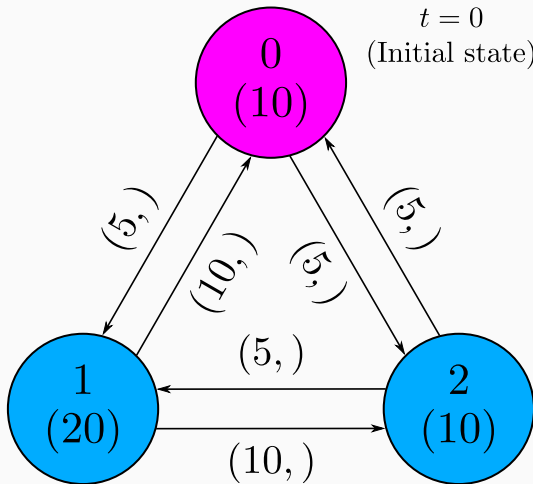
Example 2 – Non-Deviating



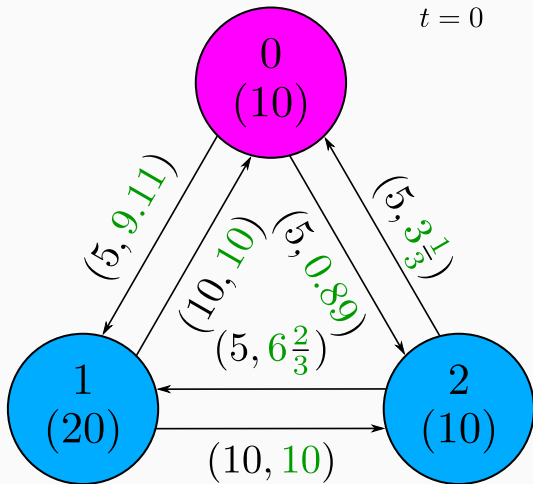
Example 2 – Non-Deviating



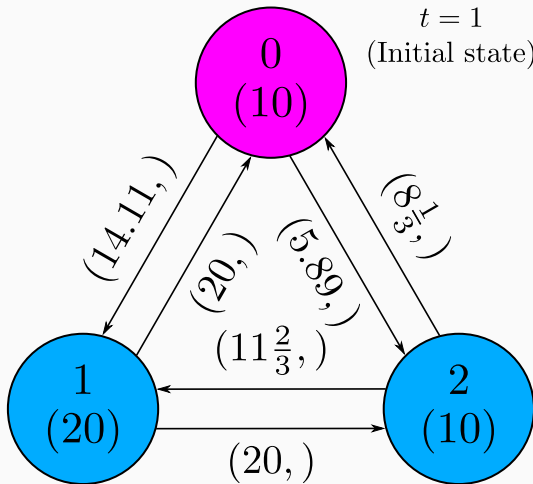
Example 2 – Deviating



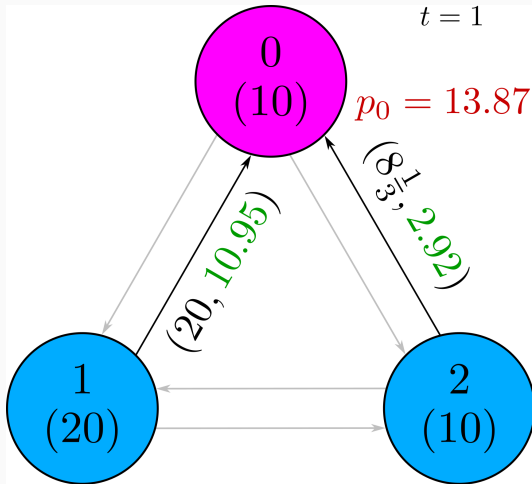
Example 2 – Deviating



Example 2 – Deviating



Example 2 – Deviating



Summarized Results

Case	Payoff
Example 1 (ND)	10
Example 1 (D)	$9\frac{1}{3}$
Example 2 (ND)	13.68
Example 2 (D)	13.87

Conclusions

- Homogeneous resource distributions
 - Any RF (trivially) NE
- Non-homogeneous resource distributions
 - NE not yet found

- Verified results of strategy simulator
- Mathematica notebook
- Intractable for nontrivial reciprocation functions
 - **Next step:** Alternative functions/representations

- Beta strategy-integration into `go-ipfs`
- IPTB: IPFS nodes in Docker containers
- Scripted tests

Plan

1. **Repeated game analysis**
 - Balances model accuracy with complexity
2. **Evolutionary game theory** (if time allows)
 - Good model, but high complexity

1. **Strategy simulator**

- Complements repeated game analysis

2. **Bitswap tests**

- Test actual IPFS nodes

Timeline

- **Thesis**
 - Layout
 - Write intro/background, results so far
- **Implementation:** IPTB simulations
- **Simulation:** Continue evaluating strategies
- **Analytical:** Simplify intractable cases

- **Thesis**
 - Update results as they come
 - Plots and visualizations
 - Check formatting with writing center
- **Implementation:** IPTB simulations
- **Simulation:** Continue evaluating strategies
- **Analytical:** Evaluate results, re-orient

- **Thesis:** Primary focus
- **Implementation:** Finish up lingering work here