# Tool-sharing system model

Anna Sokolova

26.02.2025

## 1 Introduction

This is a model of a neighborhood tool-sharing program made for the session 3 of the "Social Networks in Social Science Research" course at the University of Mannheim.

## 2 Model Design

I model a neighborhood tool-sharing network where participants can borrow and return tools. The network consists of nodes representing households connected by ties, signifying friendly relationships. Actors face a repeated decision over the number of rounds:

- **Cooperate (C)**: Return borrowed tools, maintain reputation, and keep access to the tool-sharing program.
- **Defect (D)**: Keep the tool without returning it, risking punishment and eventual exclusion.

Before the game starts, all actors contribute to acquiring the tools. By using the community tools, they receive benefit. Defection can result in punishment from direct neighbors (other actors in the network the defector is connected to). Repeated defection reduces the probability of future borrowing.

## 3 Model Parameters

- **Initial investment:** $C_{\text{init}} = 50$ (cost to join the program).
- **Benefit per round:** $B = 4$ (fixed gain from tool access).
- **Punishment per neighbor:** $P = 2$, applied with probability 0.6.
- **Future interaction penalty:** $F = 3$, applied when access is lost.
- **Borrowing probability:**

  $$P_{\text{borrow},i} = \max(0, 1 - 0.1 \times \text{past defections})$$

  Each defection reduces borrowing probability by 10%.

# 4 Utility Functions

## 4.1 Cooperation utility

If a participant always cooperates, their utility function is:

$$U_{\text{cooperate},i}(T) = -C_{\text{init}} + T \cdot B \tag{1}$$

$$U_{\text{cooperate},i}(T) = -50 + 4T$$

## 4.2 Defection utility

If a participant defects, their utility per round is:

$$U_{\text{defect},i}(T) = -C_{\text{init}} + \sum_{t=1}^{T} (B - E[P_i] - (1 - P_{\text{borrow},i})F) \tag{2}$$

where the expected punishment per round is:

$$E[P_i] = d_i \times (P \times 0.6) \tag{3}$$

and $d_i$ is the degree (number of neighbors).

# 5 Main Strategies

## 5.1 Full cooperation

- Always return tools.
- No punishment is applied.
- Utility over $T$ rounds:

$$U_{\text{cooperate},i}(T) = -50 + 4T$$

## 5.2 Full defection

- Always defect (keep tools).
- Faces punishment from neighbors.
- Borrowing probability decreases over time.
- Utility over $T$ rounds:

$$U_{\text{defect},i}(T) = -50 + \sum_{t=1}^{T} (4 - E[P_i] - (1 - P_{\text{borrow},i})F)$$

# 6 Network Structure

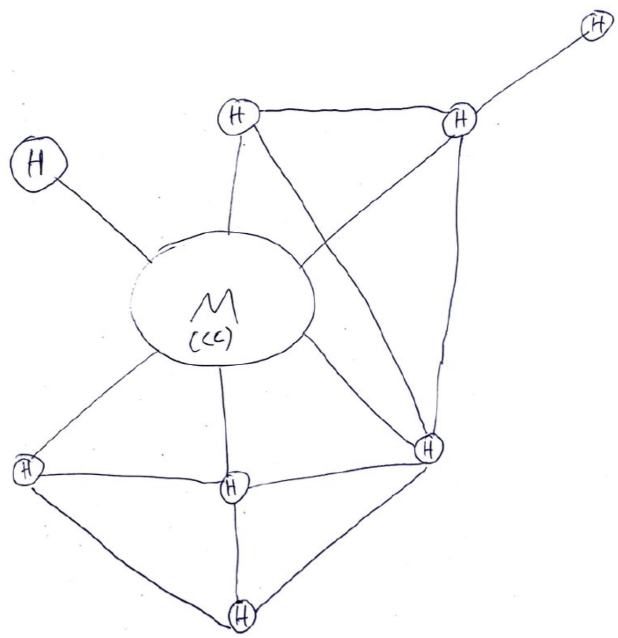


Figure 1: Network Structure for Tool-Sharing Program

# 7 Utility Calculations

Let's calculate some utilities for two different actors in the networks.

- **Actor A**: Rather isolated, only has one tie.
- **Actor B**: Connected to others, linked to fours neighbors.

I will calculate the utilities for the 3 above-mentioned strategies:

- **Full cooperation**: See 5.1 above.
- **Full defection**: See 5.2 above.
- **30% defection**: Defects randomly with 30% probability per round. Punishment and borrowing probability reductions are stochastic.

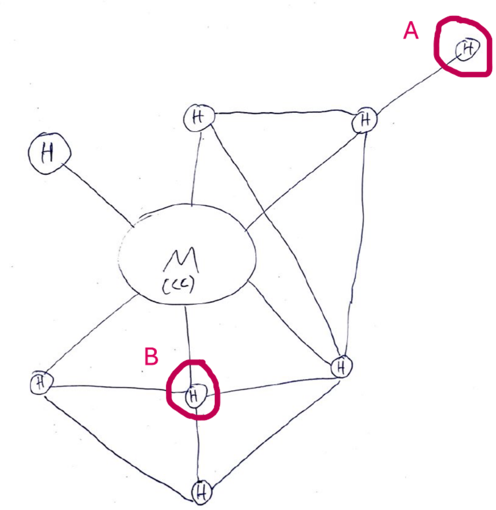


Figure 2: Actors in the Tool-Sharing Program

### 7.1 Utilities for selected actors

For each actor, I compute their total utility over 10, 50, and 100 rounds based on the three strategies:

| Strategy | Actor A (1 tie) | Actor B (4 ties) |
|---|---|---|
| Full Cooperation - 10 rounds | -10.00 | -10.00 |
| Full Cooperation - 50 rounds | 150.00 | 150.00 |
| Full Cooperation - 100 rounds | 350.00 | 350.00 |
| Full Defection - 10 rounds | -35.50 | -59.50 |
| Full Defection - 50 rounds | -43.50 | -163.50 |
| Full Defection - 100 rounds | -112.90 | -256.30 |
| 30% Defection - 10 rounds | -15.00 | -25.50 |
| 30% Defection - 50 rounds | 45.00 | -50.30 |
| 30% Defection - 100 rounds | 110.20 | -90.10 |

Table 1: Utility Comparison for Selected Actors under Different Strategies

## 7.2 Visualizing utilities

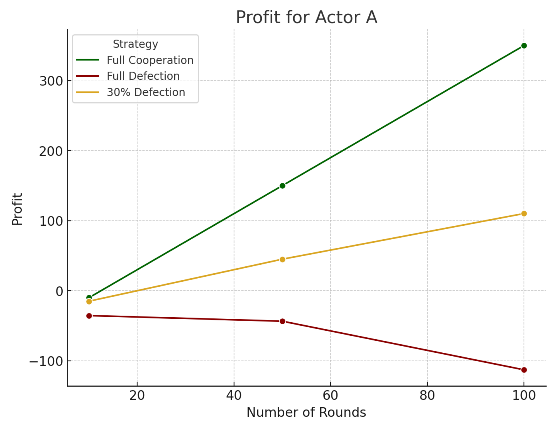


Figure 3: Actor A profits over time with different strategies

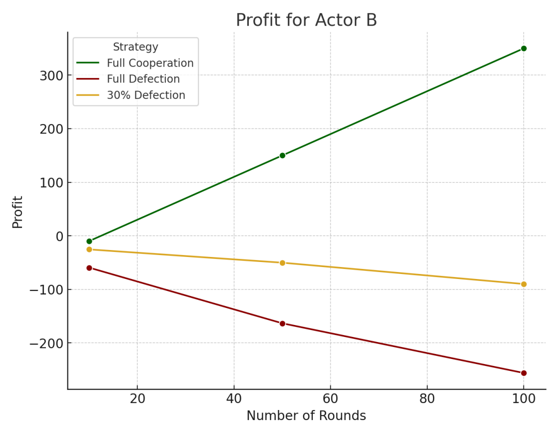


Figure 4: Actor B profits over time with different strategies

Below is the R code used to generate the profit graphs for actors A and B:

```r
# Load packages
library(ggplot2)
library(tidyr)
library(dplyr)

# Create the data frame based on the table above
data <- data.frame(
  Strategy = c("Full Cooperation", "Full Cooperation", "Full Cooperation",
               "Full Defection", "Full Defection", "Full Defection",
               "30% Defection", "30% Defection", "30% Defection"),
  Rounds = c(10, 50, 100, 10, 50, 100, 10, 50, 100),
  Actor_A = c(-10.00, 150.00, 350.00, -35.50, -43.50, -112.90,
              -15.00, 45.00, 110.20),
  Actor_B = c(-10.00, 150.00, 350.00, -59.50, -163.50, -256.30,
              -25.50, -50.30, -90.10)
)

# Reshape the data for plotting
data_long <- data %>%
  pivot_longer(cols = c(Actor_A, Actor_B),
               names_to = "Actor",
               values_to = "Profit")

# Save 1st graph for Actor A
png("actor_a_profit.png", width = 800, height = 600)
ggplot(data_long %>% filter(Actor == "Actor_A"),
       aes(x = Rounds, y = Profit, color = Strategy, group = Strategy)) +
  geom_line(size = 1) +
  geom_point(size = 3) +
  labs(title = "Profit for Actor A",
       x = "Number of Rounds",
       y = "Profit") +
  scale_color_manual(values = c("Full Cooperation" = "darkgreen",
                                "Full Defection" = "darkred",
                                "30% Defection" = "yellow")) +
  theme_minimal() +
  theme(legend.title = element_blank())
dev.off()

# Save 2nd graph for Actor B
png("actor_b_profit.png", width = 800, height = 600)
ggplot(data_long %>% filter(Actor == "Actor_B"),
       aes(x = Rounds, y = Profit, color = Strategy, group = Strategy)) +
  geom_line(size = 1) +
  geom_point(size = 3) +
  labs(title = "Profit for Actor B",
       x = "Number of Rounds",
       y = "Profit") +
  scale_color_manual(values = c("Full Cooperation" = "darkgreen",
                                "Full Defection" = "darkred",
                                "30% Defection" = "yellow")) +
  theme_minimal() +
  theme(legend.title = element_blank())
dev.off()
```