Agents' Social Environment Rewarding System Algorithm - ASERSA

Budd McCrackn

Introduction

ASERSA is a concept that integrates socio-economic principles into an agent-based system where agents interact and develop based on variables such as wealth, influence, responsibility, self-esteem, willpower, ambition, competence, inspiration, and action level. The aim is to create a dynamic environment where agents not only optimize their individual goals but also contribute to the community.

This document develops mathematical equations that formally describe the ASERSA model, integrating foundational elements from the Dynamic Force Index Algorithm (DFIA). While DFIA provides the environment for calculating key attributes like volume, influence, and force, ASERSA builds upon these to model complex agent dynamics. We introduce non-linear relationships between variables to create more realistic and intricate interactions. Concrete examples and descriptions of visual elements are included to make the concept more tangible.

1. Definition of Agent Variables

For each agent X at time t, we define the following variables within the ASERSA framework, taking into account the integration with the Dynamic Force Index Algorithm (DFIA):

- XF_t: Initially represents the **Wealth** of the agent, calculated as the sum of tokens held by the agent, which acts as the agent's initial 'force' within the DFIA context. After the DFIA computation, this is referred to as AF_t, reflecting the **Agent Force** or updated wealth.
- I_t: **Influence**, calculated based on the dynamic interactions and the agent's impact within the network.
- AS_t: **Agent Status**, derived from DFIA computations, indicating the agent's relative status within the collective environment.
- R_t: **Responsibility**, which scales with the agent's force relative to the societal force.
- S_t: **Self-Esteem**, influenced by the agent's status and societal interactions.
- V_t: **Willpower**, a psychological measure influenced by self-esteem and inspiration.
- A_t: **Ambition**, driven by willpower and the agent's current inspiration level.

- C_t: **Competence**, representing the agent's capability and effectiveness within the environment.
- IN_t: **Inspiration**, reflecting motivational factors derived from the agent's influence and societal interactions.
- AL_t: **Action Level**, quantifying the agent's activity within a time step, affected by ambition and competence.

Initialization and Example: At initialization, each agent's variables are set based on the system's initial conditions and the agent's assigned attributes. For instance, assume an agent X starts with the following initial values at time $t = 0$:

- $XF_0 = 50$ — representing average initial wealth.
- Subsequent variables $(I_t, AS_t, R_t, S_t, V_t, A_t, C_t, IN_t, AL_t)$ are initialized based on functions that consider XF_t and other dynamic factors specific to the agent's role and environment.

These variables are dynamically updated as per the DFIA and subsequent computational models within ASERSA, ensuring that each agent's behavior reflects their interactions, status changes, and the broader system dynamics. This dynamic updating simulates complex behaviors and outcomes based on evolving agent attributes and their interactions within a socio-economic system.

2. Dynamic Force Index Algorithm (DFIA)

The DFIA is integral to the ASERSA framework, designed to analyze and quantify the dynamic influence of agents within a collective environment, represented as a zone of influence that totals 100%. The algorithm tracks the interactions and relative impact of each agent over time, providing insights into patterns of influence and status among the agents.

Algorithm Overview

DFIA normalizes and quantifies the influence of agents by calculating a 'force' for each agent based on their attributes and then determines each agent's relative influence and status within the total zone.

Mathematical Definitions and Model

- **Zone (z)**: Total theoretical volume of the dataset, constant at 100%.
- **Number of Agents (Xn)**: Total number of agents within the system.
- **Theoretical Volume per Agent (Xz)**:

$$Xz = \frac{z}{Xn}$$

  This provides an initial volume allocation for each agent within the zone.

- **Force-value (XF_t)**: Sum of tokens held by an agent at time t, representing its initial 'force' or wealth.
- **Total Force-value (XnF_t)**: Cumulative force of all agents at time t.

$$XnF_t = \sum_{\text{all agents}} XF_t$$

- **Remaining Force-value ($XrnF_t$)**:

$$XrnF_t = XnF_t - XF_t$$

  Force-value excluding the agent under consideration.
- **Force-Index (ΣXi_t)**:

$$\Sigma Xi_t = \frac{XnF_t \times (Xn - 1)}{XrnF_t \times Xn}$$

  Adjustment factor reflecting relative influence.
- **Actual Volume (Xz_t)**:

$$Xz_t = Xz \times \Sigma Xi_t$$

  Represents the actual 'space' occupied by an agent.
- **Relative Influence (Xzo_t)**:

$$Xzo_t = Xz_t - Xz$$

  Deviation from the theoretical volume, indicating the influence level.

Variable Renaming for ASERSA Integration

In the context of ASERSA, the variables are renamed to better align with the algorithm's focus on social dynamics:

- **Relative Society Force (SF_t)**: $SF_t = XrnF_t$
- **Relative Agent Force (AF_t)**: $AF_t = XF_t$ (after DFIA computations)
- **Relative Society Status (SS_t)**: $SS_t = z - Xz_t$
- **Relative Agent Status (AS_t)**: $AS_t = Xz_t$
- **Relative Society Influence (SI_t)**: $SI_t = \Sigma Xi_t$
- **Relative Agent Influence (AI_t)**: $AI_t = Xzo_t$

Computational Steps

The algorithm operates through a series of steps to compute the influence of each agent dynamically:

1. Initialize total zone and calculate the initial theoretical volume per agent.
2. Compute the force-value XF_t for each agent based on the current tokens held.
3. Calculate the total and remaining force-values XnF_t and $XrnF_t$.
4. Determine ΣXi_t and compute the actual and relative volumes Xz_t and Xzo_t.
5. Update the agent attributes in ASERSA by using the renamed variables for societal and individual dynamics analysis.

Integrating DFIA into ASERSA

DFIA calculations provide the groundwork for further computations in ASERSA, linking the dynamic social status and influence of agents to their actions and interactions within the model. By accurately modeling these initial values, we can simulate a variety of complex behaviors and outcomes based on agent interactions and their evolving attributes.

3. Non-Linear Causal Relationships between Variables

The ASERSA framework models complex interactions among agents through non-linear dynamics based on the DFIA outputs. These interactions encompass various aspects such as responsibility, self-esteem, inspiration, willpower, ambition, competence, and action levels, all of which are dynamically influenced by the agents' status and societal metrics.

1. **Responsibility Calculation:**
$$R_t = \frac{AF_t}{SF_t} \cdot R_{\text{opt}},$$
where AF_t is the agent's force (wealth after DFIA), SF_t is the society's force, and R_{opt} is the predefined optimal responsibility level.

2. **Self-Esteem Dynamics:**
$$S_t = \left(\frac{AS_t}{SS_t + AS_t}\right)^2 \cdot S_{\text{opt}},$$
with AS_t and SS_t denoting the agent's and society's statuses respectively, and S_{opt} as the optimal self-esteem level.

3. **Inspiration Influences:**
$$IN_t = \sqrt{\frac{AI_t}{SI_t} \cdot I_{\text{opt}}},$$
reflecting the inspiration based on the agent's influence AI_t and the societal influence SI_t, modulated by I_{opt}.

4. **Willpower Formulation:**
$$V_t = V_{\text{opt}} \cdot \left(1 - e^{-S_t \cdot IN_t}\right),$$
where V_{opt} is the maximum willpower achievable.

5. **Ambition Driven by Inspiration and Responsibility:**
$$A_t = k_6 \cdot \left(1 - e^{-\frac{IN_t}{R_t}}\right),$$
with k_6 being a scaling constant.

6. **Competence Development:**
$$C_{t+1} = C_t + k_7 \cdot A_t \cdot (C_{\max} - C_t),$$
where k_7 is the learning rate, and $C_{\max}$ represents the maximum possible competence.

7. **Action Level Determination:**
$$AL_t = \psi \cdot \left((C_t \cdot V_t \cdot A_t)^{\frac{1}{3}}\right) \cdot \left(1 - e^{-\left((C_t \cdot V_t \cdot A_t)^{\frac{1}{3}}\right)}\right),$$
with ψ being a proportionality constant.

These equations model the agents' dynamics within ASERSA, where each agent's state and behavior are continuously updated based on their interactions and internal states, influenced by the broader societal context and individual attributes.

4. Tax Calculations and Policy Implementation

ASERSA employs a dynamic taxation system where taxes are computed based on agent attributes and societal configurations. The model supports different taxation strategies including flat, progressive, and universal basic income (UBI) approaches, which influence agent behaviors and resource allocation within the simulation.

4.1. Tax Rate Calculation

Each agent's tax rate is computed based on their wealth (post-DFIA, represented by AF_t), agent status (AS_t), and economic stability:

$$\tau_X(t) = \tau_{\max} \cdot \left(\omega_{AF} \cdot \frac{AF_t}{AF_{\text{total}}(t)} + \omega_{AS} \cdot \frac{AS_t}{AS_{\text{opt}}} + \omega_E \cdot E(t)\right),$$

where:

- $\tau_{\max}$: Maximum tax rate.
- $\omega_{AF}, \omega_{AS}, \omega_E$: Weights for agent force (wealth), agent status, and economic stability, summing to 1.
- $AF_{\text{total}}(t) = \sum_X AF_t$: Total wealth (agent force) of all agents at time t.
- AS_{opt}: Optimal agent status level.
- $E(t)$: Economic stability factor at time t (normalized between 0 and 1).

4.2. Tax Collection and Redistribution

The tax collected from each agent is:

$$\text{Tax}_X(t) = \tau_X(t) \cdot AF_t.$$

After tax collection, redistribution among agents is carried out using mechanisms like UBI, progressive redistribution, or custom strategies based on relative deprivation indices.

5. Adaptive Reward Function in Reinforcement Learning

The adaptive reward function dynamically adjusts based on agent performance, considering individual utility from wealth changes, community contributions via taxes, and self-improvement reflected by changes in agent status. The reward calculation integrates these factors using a weighted sum approach, where weights are adaptively updated to optimize agent performance over time.

5.1. Reward Function Definition

The reward function $r_X(t)$ is defined as:

$$r_X(t) = \alpha(t) \cdot \Delta AF_t + \beta(t) \cdot C_X^{\text{comm}}(t) + \gamma(t) \cdot \Delta AS_t,$$

where:

- $\alpha(t), \beta(t), \gamma(t)$: Weights for the reward components that dynamically adjust and always sum to one.
- ΔAF_t: Change in wealth (agent force) due to token transactions.
- $C_X^{\text{comm}}(t)$: Contribution to the community, determined by taxes paid.
- ΔAS_t: Change in agent status from time $t-1$ to t.

5.2. Dynamic Adjustment of Weights

Weights are updated using a gradient ascent method aimed at maximizing a performance measure $P_X(t)$:

$$P_X(t) = (1-\lambda) \cdot r_X(t) + \lambda \cdot P_X(t-1),$$

where λ is a smoothing factor between 0 and 1.

5.3. Gradient Calculation and Weight Update

The temporal difference error is calculated as:

$$\delta(t) = r_X(t) + \lambda \cdot P_X(t) - P_X(t-1).$$

Gradients for the weights are:

$$\frac{\partial r_X(t)}{\partial \alpha(t)} = \Delta AF_t, \quad \frac{\partial r_X(t)}{\partial \beta(t)} = C_X^{\text{comm}}(t), \quad \frac{\partial r_X(t)}{\partial \gamma(t)} = \Delta AS_t.$$

Weights are updated as:

$$\theta(t+1) = \theta(t) + \eta \cdot \delta(t) \cdot \frac{\partial r_X(t)}{\partial \theta},$$

where $\theta(t) \in \{\alpha(t), \beta(t), \gamma(t)\}$ and η is the learning rate.

5.4. Normalization of Weights

After updating, weights are normalized:

$$\theta(t+1) = \frac{\theta(t+1)}{\alpha(t+1) + \beta(t+1) + \gamma(t+1)}.$$

5.5. Example

Assuming:

- Initial weights: $\alpha(0) = 0.4, \beta(0) = 0.3, \gamma(0) = 0.3$.
- Learning rate: $\eta = 0.05$.
- Smoothing parameter: $\lambda = 0.9$.
- At time $t = 1$:
  - $\Delta AF_t = 5$.
  - $C_X^{\text{comm}}(1) = 2$.
  - $\Delta AS_t = 1$.
  - $r_X(1) = 0.4 \cdot 5 + 0.3 \cdot 2 + 0.3 \cdot 1 = 2.9$.
  - $P_X(1) = (1 - 0.9) \cdot 2.9 + 0.9 \cdot P_X(0)$ (assuming $P_X(0) = 0$).

Compute the temporal difference error:

$$\delta(1) = r_X(1) + \lambda \cdot P_X(1) - P_X(0) = 2.9 + 0.9 \cdot 2.9 - 0 = 5.51.$$

Update weights:

$$\begin{aligned}
\alpha(1) &= 0.4 + 0.05 \cdot 5.51 \cdot 5 = 0.4 + 1.3775 = 1.7775, \\
\beta(1) &= 0.3 + 0.05 \cdot 5.51 \cdot 2 = 0.3 + 0.551 = 0.851, \\
\gamma(1) &= 0.3 + 0.05 \cdot 5.51 \cdot 1 = 0.3 + 0.2755 = 0.5755.
\end{aligned}$$

Normalize weights:

$$\text{Total} = 1.7775 + 0.851 + 0.5755 = 3.204.$$

$$\alpha(1) = \frac{1.7775}{3.204} \approx 0.555,$$
$$\beta(1) = \frac{0.851}{3.204} \approx 0.266,$$
$$\gamma(1) = \frac{0.5755}{3.204} \approx 0.179.$$

The updated weights reflect a higher emphasis on individual utility due to its higher contribution to the reward.

6. Network Effects in ASERSA

Network effects play a crucial role in ASERSA by influencing agents' competence and behavior through their interactions. The network structure determines how agents learn from each other and how influence propagates within the system.

6.1. Network Structure

- **Agent Network (G)**: A graph $G = (N, E)$ where N is the set of agents and E is the set of edges representing relationships between agents.
- **Edge Weights (w_{ij})**: Each edge e_{ij} has a weight w_{ij} indicating the strength and type of interaction between agent i and agent j.
  - $w_{ij} > 0$: Positive influence (cooperation, support).
  - $w_{ij} < 0$: Negative influence (competition, conflict).

6.2. Updating Competence with Network Effects

Agents update their competence based on their ambition, inspiration, and interactions with their neighbors:

$$C_{t+1} = C_t + k_7 \cdot A_t \cdot (C_{\max} - C_t) + \kappa \cdot \left(\overline{C}_{\text{net}}(t) - C_t\right),$$

where:

- $\overline{C}_{\text{net}}(t)$: Weighted average competence of agent X's neighbors.

$$\overline{C}_{\text{net}}(t) = \frac{\sum_{j \in N_X} w_{Xj} \cdot C_j(t)}{\sum_{j \in N_X} |w_{Xj}|},$$

  with N_X being the set of neighbors of agent X.
- κ: Learning rate from neighbors.

6.3. Impact of Network Topology

The structure of the network significantly influences the dynamics of competence and influence:

- **Scale-Free Networks**: Promote the emergence of highly influential agents (hubs).
- **Small-World Networks**: Facilitate rapid information and influence spread.
- **Random Networks**: Offer uniform interaction opportunities.

7. Concluding Remarks

By integrating socio-economic variables, non-linear relationships, adaptive learning models, and dynamic tax and redistribution policies into the dynamics of the agents, ASERSA creates a system where agents not only focus on their individual goals but also adapt their behaviors based on their experiences and interactions with others.

This enriched model can be used to study various phenomena in socio-economic systems, such as wealth distribution, social mobility, and the impact of policies on individual and collective behaviors.