EDA, An Empathy-Driven Computational Architecture

Xinmiao Yu, Riccardo Morri, Dr. Fernanda Eliott
ELBICA Lab, Grinnell College

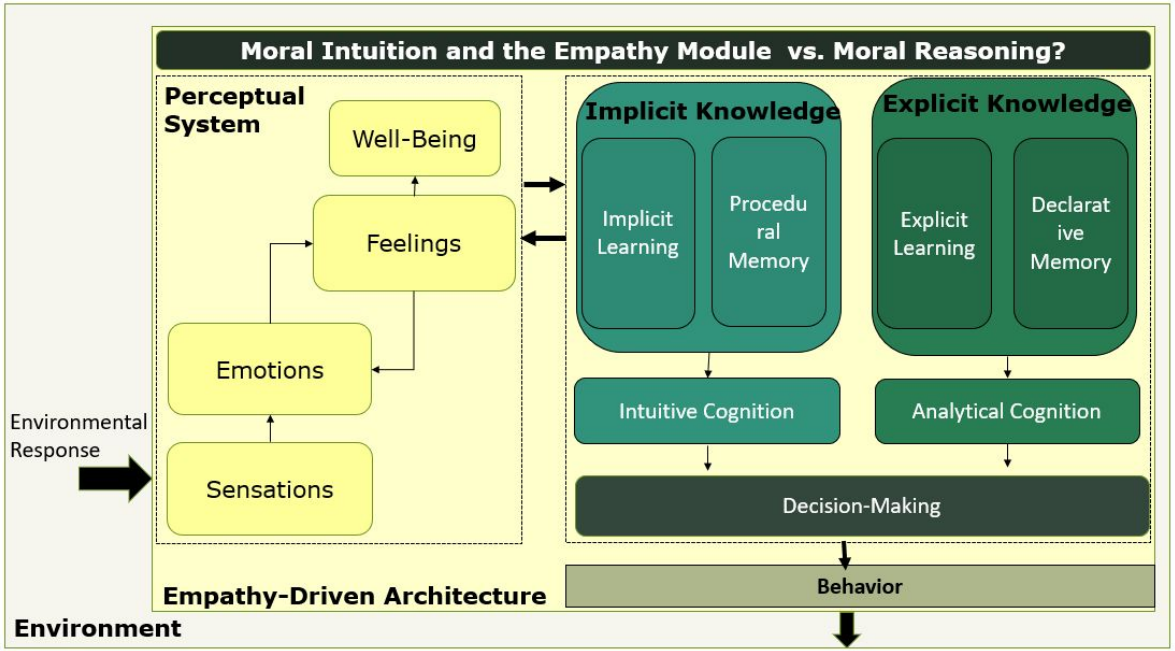
Goals

- Design a biologically-inspired computational architecture, *EDA*, which utilizes artificial emotions, feelings, empathy, and moral cognition
- Agents should naturally select cooperative behavior over selfish behavior to accomplish tasks in multi-agent systems
- Drive insights on how human empathy and morality works, and how it affects human decision-making
    - Not claiming to be an exact theory of these ideas, however

Important Concepts

- Emotions and feelings are necessary for rational decision-making
- Empathy is only possible with emotions and feelings, and is a catalyst for moral intuition
- Moral behavior is a form of cooperation

EDA's Design


Moral Intuition and the Empathy Module vs. Moral Reasoning?
Perceptual System
Well-Being
Feelings
Emotions
Sensations
Environmental Response
Implicit Knowledge
Implicit Learning
Procedu ral Memory
Explicit Knowledge
Explicit Learning
Declarat ive Memory
Intuitive Cognition
Analytical Cognition
Decision-Making
Behavior
Empathy-Driven Architecture
Environment

EDA's Empathy

a. Estimates the situation of another agent using its own emotions
b. Decides how sensitive it should be towards the other agent's situation by analyzing their interaction history
c. Calculates an **empathy coefficient** using this information

EDA's Emotions

Fear: Response to imminently dangerous stimuli.

Happiness: Response to situations beneficial to the agent's survival and well-being.

Anger: Response to loss of control over the environment.

Sadness: Response to stimuli harmful to long-term success.

Disgust: Response to revolting or distasteful stimuli.

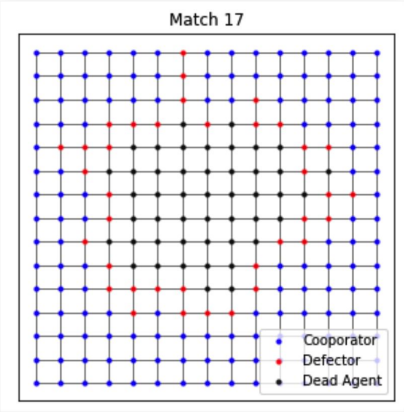
Surprise: Response to unpredictability in the environment.

Experiments

Experimenting Emotion Equations with PDG

PDG

- PDG stands for **Prisoner's Dilemma Game**
- In our simulation experiment, we use an undirected graph to represent agents and their interactions. Interactions are mutual if both agents are alive
  - **Nodes** in the graph represent different **agents**
    - Surviving agents can be either defectors or cooperators
  - **Edges** represent **interactions** between agents
    - Only interactions with **cooperators** renders a positive reinforcement
- To simplify the problem, we are only using a **lattice network**
  - Lattice network: each agent has 4 neighbors to interact with.
  - Agents' policies are hand designed without learning.



Simulation Procedure

- Each simulation is divided into matches where
  - Agents **interact** with all neighbors, receive reinforcement from interactions, and update **emotional values** based on reinforcement received.
  - Agents will be **eliminated** if they do not receive enough reinforcement from interactions
  - Agents might turn from **cooperators** to **defectors** if its defecting neighbor received higher reinforcement within this match

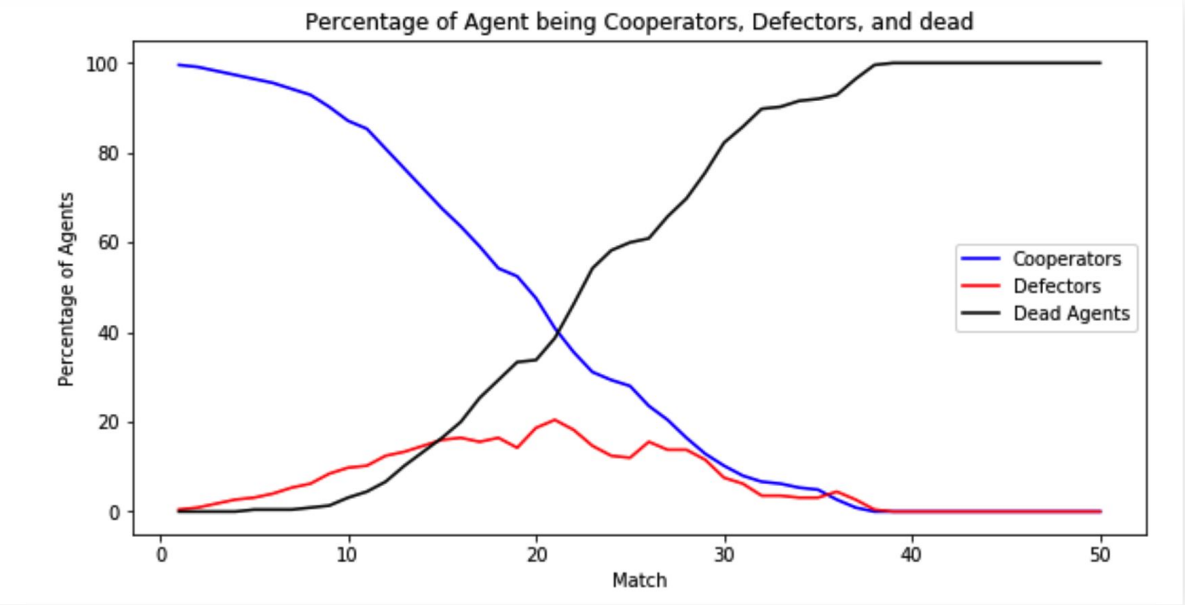
Reinforcement	**Agent Being a Cooperator**	**Agent Being a Defector**
Neighbor is a Cooperator	$1/V_i$	$2/V_i$
Neighbor is a Defector	0	0
Neighbor is Dead	0	0

Experimental Setup

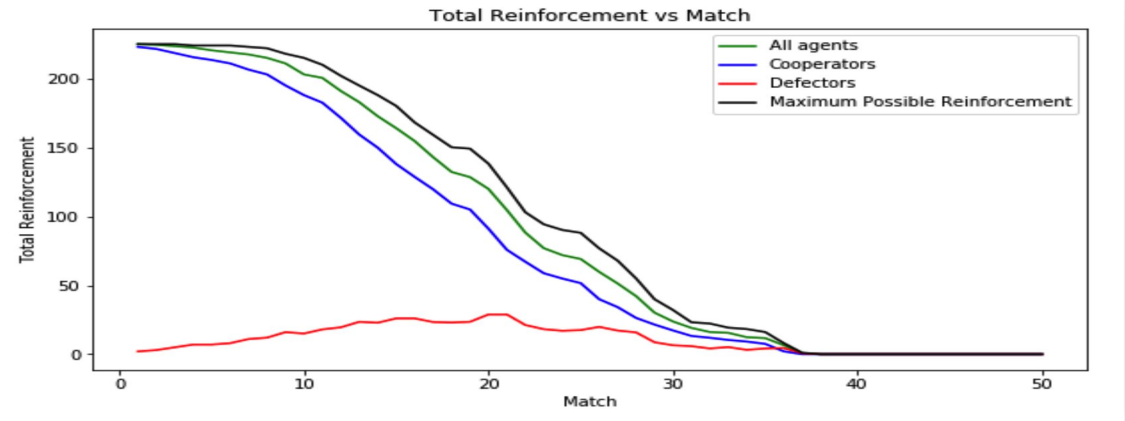
- PDG is played in a lattice network with 225 agents.
  - At the beginning of the simulation, there are 224 corporators and one defector who is at the center
- *Ti* defines the total reinforcement that each agent need to receive in order to survive.
- The probability that an agent will change its strategy from a corporator to a defector is 0.25.
  - If an agent has a defecting neighbor with high total reinforcement, there is a 0.25 chance that the agent will also become a defector
- The table below summarizes reinforcement from one interaction

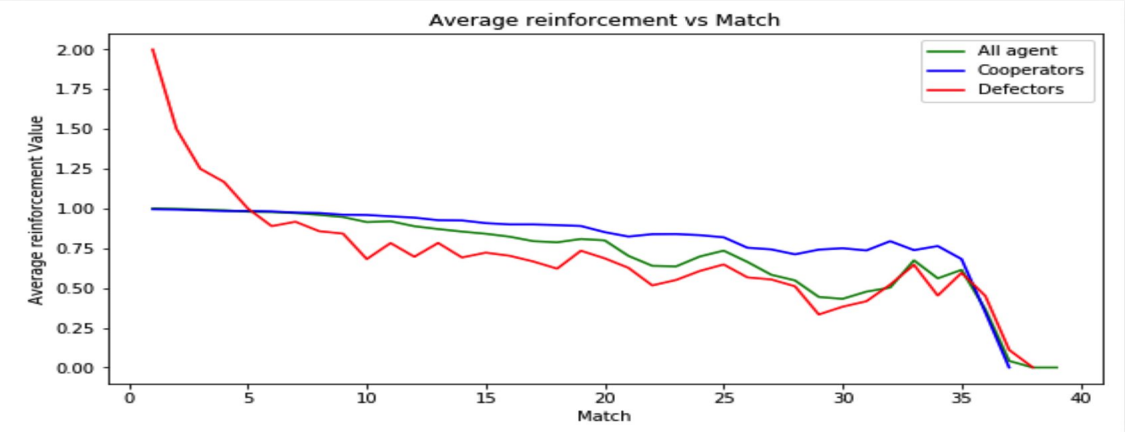
Reinforcement	**Agent Being a Cooperator**	**Agent Being a Defector**
Neighbor is a Cooperator	$1/V_i$	$2/V_i$
Neighbor is a Defector	0	0
Neighbor is Dead	0	0

Results


Percentage of Agent being Cooperators, Defectors, and dead
Percentage of Agents
Match
0
20
40
60
80
100
0
10
20
30
40
50
Cooperators
Defectors
Dead Agents

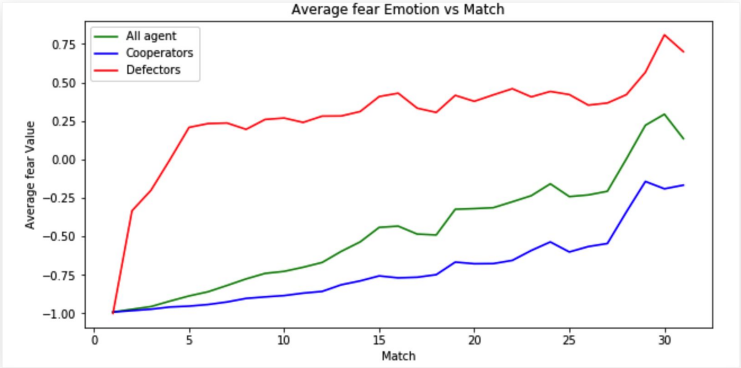
Results

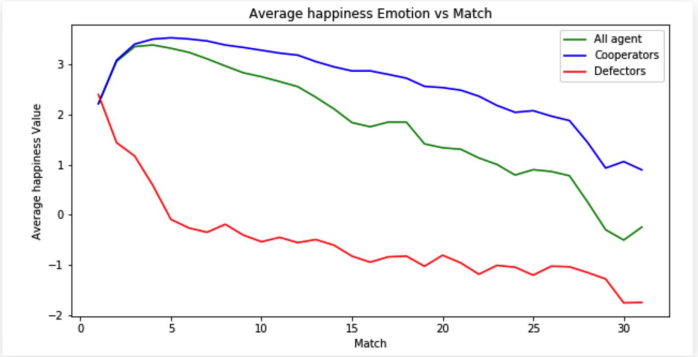

Total Reinforcement vs Match
All agents
Cooperators
Defectors
Maximum Possible Reinforcement
Total Reinforcement
Match
0
50
100
150
200
0
10
20
30
40
50


Average reinforcement vs Match
All agent
Cooperators
Defectors
Average reinforcement Value
Match
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
0
5
10
15
20
25
30
35
40

Emotions

- We tracked and analyzed the change of emotion values in this experiment.
- Equation for calculating emotion values (fear, happiness, anger, sadness, surprise, and disgust) are discussed in the first section.





Final Remarks

- Results for emotions were limited due to the nature of the experiment
- Explore different experimental parameters and test-beds
    - Identify strengths and weaknesses of the design
- Continue developing and concretizing the design of *EDA*
    - Empathy module, reputation function, implicit and explicit knowledge modules
- Implement the architecture in physical, robotic agents