Playing Against Future Minds:
A Game Theoretic Approach To Novel Agent Emergence

Dani Burgos-Lisle
Undergraduate, University of Colorado, Boulder

In the *Foundation* series, the clairvoyant Gaal is haunted by visions of her daughter's death at the hands of the Mule - a warlord living one-hundred years in the future. In the present, Gaal strives to tilt the odds in her favor and render this future impossible[1]. Outside of science fiction, the concepts of dueling against a future opponent or cooperating with a future ally are no less compelling. In a sense, they are as much part of the game as we are, and failing to be prepared for them has consequences far beyond fiction and thought experiments. As we will soon see, geopolitical decision-making and large-scale technological integration strategy are at stake. There are significant consequences to a lack of advance information about future player forms in any evolving game. We want to understand how new agents in the future may think and behave so that we can take advantage of their emergence rather than being taken off-guard. Furthermore, we need the ability to simulate the evolution of agents that are fundamentally different than those currently in play. This means that superficial trait and strategy changes are insufficient - agents need to have the ability to change on an atomic level. They also need to be able to behave collectively so that the emergence of larger-scale agents is possible. With these specifics in mind, we turn to a field known for answering questions of strategy amongst adaptive players: evolutionary game theory. Unfortunately, here we find a distinct lack of theory on robustly projecting future agents because the ability to predict significant evolution of an agent beyond simple strategy changes over time is limited[2]. In order to bridge this gap, we propose a theoretical framework (with suggested implementation examples) that allows agents to transform via structurally evolving the dynamical equations that govern their change and movement, and passing these dynamics down to offspring via crossover and mutation.

1 Motivation: New Friends and Foes

Various changing global conditions are accelerating the birth rate of new player forms in a variety of economic games. The need to adapt to growing global population and dwindling resources along with rapid technological growth will necessitate the formation of new organization entities and operational dynamics, each with their own cognitive processes. While these developments are exciting and promising, careful forethought will be required to ensure they do not bring disaster along with them.

[1] Goyer and Friedman (2021)
[2] Adami et al. (2016)

1.1 Changing Population-Resource Dynamics

Governments around the world are struggling to make and keep resources available while implementing sustainable practices. Meanwhile, world population is projected to exceed 10 billion before peaking[3]. In order to meet objectives such as the UN-prescribed 70 percent increase in food production or the integration of resource feedback loops[4], governments are likely to restructure internally[5] while forming and revising alliances[6] leading to new decision-making dynamics across the globe.

1.2 Transformative Technological Growth

The rise of artificial intelligence and the rapid development of digital technology is shifting organizational priorities and demands. New technologies are rapidly providing new sources of labor, strategic depth, and prediction capability in many sectors. Across the board, collaboration between human and machine and competition dynamics are changing and artificial intelligence is leading to more frequent partnerships between academic entities, corporations, and governments[7]. Deloitte projects the increased popularity of decentralized, network-based organizational schemas[8] The runaway growth of technology in the artificial intelligence space necessitates the emergence of new player forms.

1.3 The Plights of the Un-Prepared

These new developments and fundamental structural changes bring risks. When new player forms first enter, checks and balances rarely exist. The First Industrial Revolution's major technological, socio-economic, and cultural improvements came with humanitarian harms that were intricately linked to new corporate entities and industrial practices. As factories proliferated, workers were forced to endure long hours at low wages, all under unsafe working conditions[9]. Industrialization also led to the release of pollutants into the air and water, which impacted public health and ecosystem, while overcrowding and poor sanitation resulted from mass movement from agrarian economy to cities[10]. Meanwhile monopolies formed, driven by profit maximization[11]. All told, the First Industrial Revolution brought unexpected challenges in stability and sustainability that we could only begin solving afterward. Prior insight might have prompted labor laws, environmental regulations, and antitrust laws in advance.

[3] World Bank (2023)
[4] Ellen MacArthur Foundation (2015)
[5] Food and Agriculture Organization of the United Nations (2020)
[6] European Political Strategy Centre (2019)
[7] Baryannis et al. (2019)
[8] Deloitte Insights (2020)
[9] Engels, Friedrich (1845)
[10] Ashton, T.S. (1948)
[11] Chandler, Alfred D. (1977)

2 The Gaps In Standard Game Theory

In attempting to avoid such outcomes, we encounter a number of obstacles in current game theory paradigms. Standard evolutionary game theory (EGT) is centered around defining a set of strategies, studying how they interact when employed by different players, and analyzing how players will change strategies over time. In general, game theory is also limited to fixed player types, so it cannot study how agents transform at a fundamental level - only how they choose from predefined strategies.

Modeling frameworks that pay more attention to individual players have been studied extensively. Agent-based models considers players with different traits and evolution dynamics and in cellular automata models like Conwy's Game of Life, players exist as disturbances in the grid-field rather than as self-contained objects. While both are compelling for studying self-organization behavior, they both require additional layers to model complex interactions.

3 Building Blocks for a Broader Theory

3.1 Agent-Based Modeling

Agent based modeling, while incomplete on its own, offers an excellent boilerplate for simulating players in our game environment. In contrast with standard EGT, it lets us model and study the cognitive processes of each individual player, rather than analyzing trends. It's important to keep track of these agents because we want to understand how they interact and come together to form agents. Our goal is to grasp the ways these agents maintain states. However it wouldn't be practical to represent agents, which consist of both humans (in contexts) and advanced technology (with complex components) using overly simplistic binary state machines like cellular automata grids. Instead we can capture scenarios by representing the agents positions in a dimensional space, where each dimension has its own unique rules. By utilizing both spatial and state dimensions we can accurately model proximity as well as incorporate attributes that allow for subtle interactions, such as similarity in characteristics.

3.2 Dynamical Systems

Dynamical systems theory can describe how the state of one agent influences those of nearby agents. It provides a robust way to define rules a system's evolution over time. The behavior of agents are often described by differential or difference equations that formulate the dynamics[12]. In the following simple example, $\frac{dx}{dt}$ could represent the velocity of an agent in space, with $\frac{dE}{dt}$ the change in the agent's energy level:

Example 1. Energy-Motion Dynamic

$$\frac{dx}{dt} = -k\frac{dE}{dt} \qquad k > 0$$

This equation formulates the rule that if the agent accelerates, it expends energy, and as it decelerates it can rest and regain energy.

[12] Strogatz, Steven H. (1994)

Similar rules can include how how the motion and attributes of one agent effects those of another. Furthermore, these interactions can lead to complex behaviors including synchronization[13]. In socioeconomic systems, differential equations can be used to model communication, mimicry, and more complex adaptive processes, as well as describe how individual actions can produce group dynamics.

Brownian Motion

In many dynamical models, Brownian motion is introduced to add randomness behavior of a simulated object. In models of social systems, this helps account for that fact that real players are not strictly rational and their is always an unpredictable component to their actions. A Brownian motion process adds a random, normally-distributed offset over a given time interval, which is completely independent of the offset at any other time. Brownian motion has been used extensively in modeling social systems[14], including those where information propagation is a key factor[15], such as those in our context. To tackle the transfer of genetic information, we turn to another theory.

3.3 Genetic Algorithms

Genetic algorithms form a class of evolutionary algorithms used to solve optimization and search problems by mimicking natural selection. This method can effectively model both information communication and genetic reproduction and mutation. In genetic algorithms the "fittest" information are selected and replicated, as in natural selection, with operations for crossover and mutation[16]. Crossover occurs in reproduction where parents exchange genetic information and create offspring that inherits features from both. Mutation introduces random changes to transmitted information, allowing the exploration of new agent forms in the evolution space[17].

[13] Pikovsky et al. (2003)
[14] Crossland, Rachel (2018)
[15] Suz Gutiez, Manuel (2020)
[16] Goldberg (1989)
[17] Mitchell (1998)

4 Integrating The New Framework

Now that we have established the building blocks that individually offer pieces of the functionality we seek in our new theoretical framework, we can proceed to fit them together in a suitable fashion. This is not a fully-defined model with all moving parts and their ranges defined, but rather a framework that defines where each mathematical tool in our box ought to be used, based on our prior review, with detailed guidelines for how to apply them in those specific areas. Agent-based modeling will provide us with a basis for keeping track of agents and their states and allowing for the emergence of irreducible group dynamics and complexity, while genetic algorithms and dynamical systems together will provide us with a method for describing and transformatively evolving individual and collective behavior.

4.1 Agents in Position and State-Space[18]

Agents form the common thread of our new model. Although it does not provide a complete picture, agent-based modeling is an excellent basis for describing the individual agents in our game space. For convenience, we refer to the individual agents or cells, which may make any number of up systems, organisms, and organizations, as primitive agents A^p. In a two-dimensional game, each A^p has position vector $\vec{r} = (x, y)$, movement vector $\vec{v} = (x', y')$, finite state vector $\vec{s} = (a, b, c, ...)$, and state-change vector $\vec{w} = (a', b', c', ...)$ (we use the derivative prime notation to indicate that each value is the rate-of-change of a corresponding quantity). In physical space, $\vec{r}$ and $\vec{v}$ define the motion of A^p and are used to detect proximity and possibly collisions with other agents, while $\vec{r} = (x, y)$ defines the position of A^p in physical space. On the other hand, $\vec{s}$ and $\vec{w}$ define the traits and attributes of the agent are how they changing at a given moment. For instance d' in $\vec{w}$ might represent the aging or decay rate, or E in the state $\vec{s} = (..., E, ...)$ might refer to the current energy, or locomotive potential, of A^p, with a value of zero denoting expiration.

State-Wise Relationships Between Agents

Compared to network models, wherein the existence of connections between two objects indicate relationships, position-state-space information lets us to define and explore more complex and higher-dimensional relationships between A^ps. In this paradigm, Euclidean distance[19] could represent a strong measure of the relationship between two agents. In following sections, we will how proximity of agents can be written into inter-agent dynamics and genetic signaling so that agents with more state-similarity will self-organize and cooperate more readily than they would otherwise, and more state-dissimilar agents may behave with more competitiveness, hostility, or have predator-prey relationships.

4.2 Dynamical Motion and State-Change

Now that we have a defined structure to represent agents and their states at any given point in time, we proceed to outline the processes by which they evolve. Here the dy-

[18] the position of agents in a theoretical space where dimensions represent attributes rather than physical directions

[19] generalized distance formula, as for the longest side of a right-triangle given the shorter sides

namical systems approach and genetic algorithms approaches are heavily intertwined. To illustrate how dynamical systems let us describe the behavior of agents, consider the following model.

Example 2. Imitation of Similar Neighbors' Movements
An agent A^p with state $\vec{s} = (\nu : \text{natural speed}, \psi : \text{group-sensitivity}, c : \text{sight range}, \chi : \text{species}, \xi : \text{interspecies-sociability})$ moves according to the following equations:

$$\vec{v} = \nu(\psi \vec{V}(t) + \vec{W}) \tag{1}$$

$$\vec{V}(t) = \sum_{i \in R(t) \cap X(t)} \vec{v}_i(t) \tag{2}$$

$$R = \{k : |\vec{r}_k(t) - \vec{r}(t)| < c\} \tag{3}$$

$$X = \{k : |\chi_k(t) - \chi(t)| < \xi\} \tag{4}$$

where $\vec{W}$ is the Brownian motion term, which adds randomness.

Moving from top to bottom, the main idea of is that A^p's velocity is partially determined by the motion of similar nearby agents and partially random. The vector $\vec{V}(t)$ is the summed velocity of similar, nearby agents at time t. The sets R and X reference other agents A_i^p within A^p's sight range c, and those within its species-sociable range ξ in χ (those it "considers" its in-group).

Proceeding back up from (4), the intersection $R \cap X$ refers to agents that are both nearby and similar, letting up summing the total velocity to find $\vec{V}$. Then the contribution of A^p's velocity is scaled by its sensitivity ψ to the motion of other A_i^ps. The Brownian motion derivative W guarantees A^p will not simply stop and do nothing in the absence of similar nearby neighbors. Finally, both the social term and the random term are scaled by ν.

We could add many more variables drawing from our knowledge of physics and cognition. We might define dynamics to facilitate sexual (or crossover) replication by adding a motion dynamic for attraction based on agents' shared replicative potential. By analyzing the nature of a system and carefully formulating dynamics, we can model many systems with detail and accuracy. Fortunately, we may not always need to handwrite every dynamic thanks to the next part of our framework: genetic replication.

4.3 Crossover Replication and Mutation Using Genetic Algorithms

When constructing an evolutionary model we must decide how to implement replicator dynamics - the mechanisms by which learning occurs through generation and information is transmitted and mutated. Here, genetic algorithms are extremely well-suited. We can let an agent's birth state $\vec{s}_b = (a_b, b_b, c_b, ...)$ be starting properties of an agent before they are changed throughout the course of its life, and further let the birth state and dynamical equations comprise its genome. To define the replication and mutation dynamics of a specific model, we must design algorithms (series of steps) for each. The first algorithm - crossover replication - decides how the genomes of parents are combined to give that of the offspring. Consider the following set of crossover algorithms.

Example 3. Crossover Algorithms
For each pair of dynamics equations:

1. Add the equations from both parents together.

2. Divide the sums by 2.

3. Pass the resulting dynamics equations to the offspring.

Using this algorithm, if parent A_i^p's movement equation is $\vec{v}_i = \nu_i(\psi_i \vec{V}_i(t) + \vec{W})$ and parent A_j^p's movement equation is $\vec{v}_j = \nu_j(\psi_j \vec{V}_j(t) + 2\vec{W})$, then the offspring's equation will be $\vec{v} = \nu(\psi \vec{V}(t) + \frac{3}{2}\vec{W})$. This is simply the average. Next, for birth state crossover :

For each variable in $\vec{s}_b$:

1. Choose a between parents A_i^p and A_j^p with equal probability.

2. Assign the value for the variable in the chosen parent's birth state to the offspring's birth state.

Using this method, the offspring will get a random mixture of the parents traits across its birth state.

For the second algorithm - mutation - we must devise a way of randomly modifying parts of the child's birth state after the crossover algorithm has been applied. For birth state, we could randomly increase or decrease each variable by a normally distributed random amount. For dynamics mutation, the possibilities here are broad, and potential for discovering new equation forms through an evolutionary process is highly compelling. But for simplicity consider the following algorithm, which emphasizes stability and factors in the possibility of losing or adding genetic information.

Example 4. Behavior (Dynamics) Mutation Algorithm
For each equation:

1. With probability 0.1, randomly choose one term of the equation (in fully expanded form) and remove it.

2. With probability 0.1, randomly choose two terms of the equation (the same term if there is only one), and add their product, scaled by factor randomly selected between -1 and 1, as a new term in the equation.

In this example, many offspring will be ill-fit to survive since randomly adding and removing terms has no guarantee of giving them advantageous behavior. However, given enough time, as in natural evolution, offspring with adaptive qualities are likely to appear and replicate.

Moreover, generating new dynamics for primitive agents will generate new dynamics for groups of agents, leading collective entities that cooperate in different ways. This provides the opportunity for novel agents to emerge and interact on a virtually infinite number of scales, making this framework highly versatile and exceptionally well-suited to our objective of modeling novel agent emergence squarely outside the standard strategy population modeling paradigm.

Conclusion

Shifting focus beyond strategies alone and toward the evolving minds that produce them is not only compelling but imperative. In this dizzyingly dynamic world, individuals and coordinated groups, rather than mere things and events, are sometimes the only patterns we can manage to grasp in time to react. To extend this approach into the future, we should base our game theoretic models on agents that evolve in physical and state-space according to dynamical equations, and evolve novel qualities and emergent group dynamics via genetic algorithms. This allows for high-dimensional states and complex, nuanced inter-agent relationships. The combination of genetic algorithms and dynamical equations was also explored by Chen in the context of discovering of partial differential equations to model physical systems[20]. The application of genetic algorithms to dynamical models effectively creates "genetic differential equations" with the power to creatively discover novel behaviors without normal boundaries. While transformative, this capability only scratches the surface of the power needed to accurately forecast future agents within the limits of computational tractability. Many further iterations and developments will follow, ever chasing the flow of time and more necessary as evolution progress. But for us, standing here on so many verges, it is no longer enough to seek mountaintop perspective - we must strive to gaze through the eyes not yet formed.

[20] Chen et al. (2022)

References

Adami, C., Schossau, J., and Hintze, A. (2016). Evolutionary game theory using agent-based methods.

Ashton, T.S. (1948). The industrial revolution.

Baryannis, G., Dani, S., and Antoniou, G. (2019). Supply chain risk management and artificial intelligence: state of the art and future research directions.

Chandler, Alfred D. (1977). The visible hand: The managerial revolution in american business.

Chen, Y., Luo, Y., Liu, Q., Xu, H., and Zhang, D. (2022). Symbolic genetic algorithm for discovering open-form partial differential equations (sga-pde).

Crossland, Rachel (2018). Brownian Motion and Crowd Psychology: Shared Moment, Shared Discourse.

Deloitte Insights (2020). Tech trends 2020.

Ellen MacArthur Foundation (2015). *The Circular Economy: A Wealth of Flows*. Ellen MacArthur Foundation.

Engels, Friedrich (1845). The condition of the working class in england.

European Political Strategy Centre (2019). Global trends to 2030: Challenges and choices for europe. https://ec.europa.eu/epsc/publications/strategic-notes/global-trends-2030$_en$.

Food and Agriculture Organization of the United Nations (2020). The state of food security and nutrition in the world 2020. http://www.fao.org/publications/card/en/c/CA9692EN.

Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning.

Goyer, D. S. and Friedman, J. (2021). Foundation. Apple TV+. Television series.

Mitchell, M. (1998). An introduction to genetic algorithms.

Pikovsky, A., Rosenblum, M., and Kurths, J. (2003). Synchronization: A universal concept in nonlinear sciences.

Strogatz, Steven H. (1994). Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering.

Suz Gutiez, Manuel (2020). Grado de percepcicial sobre la violencia en mco a partir de datos difundidos en la red social online de twitter.

World Bank (2023). World population will continue to grow. https://datatopics.worldbank.org/world-development-indicators/stories/world-population-will-continue-to-grow.html. Accessed: 2023-12-20.