# Extreme Environments Perpetuate Cooperation - Supplementary Materials

***Index Terms*—Agent-based modelling, Cooperation, Social Stratification, Environmental Stress**

## I. Introduction

This document serves as the collection of supplementary material produced for the paper "Extreme Environments Perpetuate Cooperation". Section 1 contains an ODD+D description of *NeoCOOP* model used in this work, Section 2 contains a summary of the experiment design, Section 3 details the parameter tuning process of *NeoCOOP*, Section 4 contains plots of the evolution of each individual scenario investigated and Section 5 contains figures of all the other results produced in this work that didn't make it into the final paper.

## II. ODD+D Description of NeoCOOP

### A. Purpose

*1) What is the purpose of the study?:* The general purpose of NeoCOOP is to be a model by which we can simulate the Paleolithic-Neolithic transitionary period that saw humanity move from largely egalitarian hunter gatherer groups to agrarian super polities typically ruled by a social elite.

*2) For whom is the model designed?:* This model is designed primarily for Computational Archaeologists. The model will also be of interest to Computational Social Scientists interested in the modelling of complex social phenomena and the Artificial Life community in general.

### B. Entities, State Variables and Scales

*1) What kinds of entities are in the model?:* There are four kinds of entities within NeoCOOP:

1) **Households**: They are the primary decision making entity in NeoCOOP (the agents). They represent a collection of occupants ruled by a patriarchal figure.
2) **Occupants**: Occupants are contained within households. They are not decision making entities and are only present to determine household resource gathering and consumption levels.
3) **Settlements**: Settlements represent a collection of households. They are also not decision making entities but are used by several of the model's systems for simulating / restricting agent adaptation.

The model represents its environment as a $nxm$ grid-world. Each cell in the grid can technically be thought of as an entity but, their primary purpose is to store local geographical information.

*2) By what attributes(i.e. state variables and parameters) are these entities characterized?:* **Environment Cell**:

1) Resources: Amount of resources $\in [0, 1]$ at a given cell.
2) Slope: The slope ($\circ$) in a given cell.
3) Is Owned: An integer value that indicates if an environmental cell is owned and by whom (-1 for not owned).
4) Is Settlement: An integer value that indicates if an environmental cell is a settlement (-1 for not a settlement).

**Household**:

1) Resources: Amount of resources an agent has.
2) Load: The amount of (decayed over time) resources the agent has donated to other agents.
3) Occupants: A list of occupants in the current Household.
4) Hunger: A value $\in [0, 1]$ that denotes the agent's hunger.
5) Satisfaction: A value $\in [0, 1]$ that denotes how happy the agent is with its living conditions.
6) Owned Land: A list of all of the land currently owned by the agent Household.
7) Storage Decay: A list storing the history of agent resource acquisitions.
8) Able Workers: The number of able workers the Household has at its disposable (This value is equal to the number of occupants whose age is greater than the age_of_maturity).
9) Peer Resource Transfer Chance: The likelihood $\in [0, 1]$ of an agent accepting a resource transfer request from a peer agent.
10) Subordinate Resource Transfer Chance: The likelihood $\in [0, 1]$ of an agent accepting a resource transfer request from a subordinate agent.
11) **Conformity ($\sigma$):** The degree to which an agent accepts cultural influence.
12) **Attachment ($\alpha$):** How much an agent values its current settlement. An agent with a high degree of attachment is less likely to migrate even if the environmental conditions suggest that it should.

**Occupant / Individual**: are only classified by a unique identifier and an age property which denotes how many iterations the occupant has been alive for.

*3) What are the exogenous factors/drivers of the model?:* Climate Change. Specifically an increasing / decreasing likelihood of drought over time.

*4) If applicable, how is space included in the model?:* The environment is a grid-world made up of equally sized cells.

*NeoCOOP* supports using GIS layers to add environmental information to the environment. This includes but, is not limited to, height, slope, water and sand content data inputted into the model as images where each pixel in the image represents the value of a given attribute at that pixel coordinate in the grid-world.

For this work, we replaced the vegetation model with a simpler resource gathering system similar to Angourakis et al. [1]'s work. The details of the changes are detailed later but, the spatially explicit grid-world is still used.

*5) What are the temporal and spacial resolutions and extents of the model?:* One iteration in *NeoCOOP* represents a single year. The soil moisture system does calculate total soil moisture (mm) on a per month basis, but it is entirely self-contained to the vegetation model systems (Global Envrionment System, Soil Moisture System and Vegetation Growth Systems). Each grid cell is 1ha in size and the total size of the grid-world is configurable in height and width $mxn$.

For this work, the soil moisture system was removed because it formed part of the vegetation model. The resource generation system we replaced it with operates on a execute once per iteration basis. We further constrained the environment such that it is always square (ie: of size $n \times n$).

### C. Process Overview and Scheduling

*1) What entity does what and in what order?:* The order of execution can be seen in Figure 1.

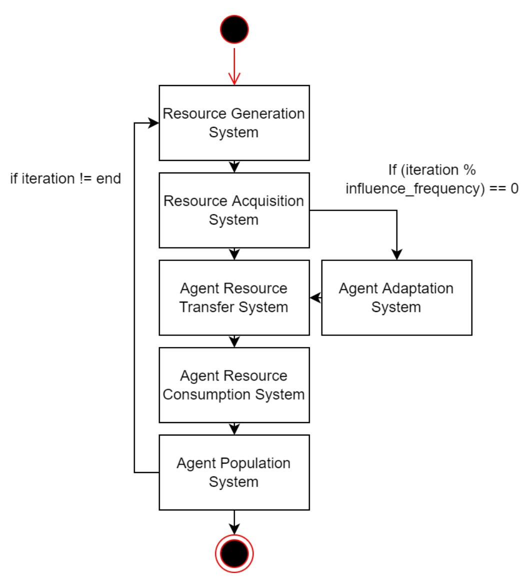



Fig. 1: Execution Cycle of *NeoCOOP*

### D. Theoretical and Empirical Background

*1) Which general theories concepts, theories or hypotheses are underlying the model's design or at the level(s) of the submodel(s) (apart from the decision model)? What is the link to complexity and purpose of the model?:* The Agent-based component of the model is also loosely based on the work of Chliaoutakis and Chalkiadakis [2], [3] and the model makes use of their self-organization scheme for simulating emergent social hierarchies.

Our Agent's resource trading preferences are probability-based. This is an extension to the typically simple cooperative - defective approach. Each agent has a probability $p$ associated with its resource trading preferences and every time a resource trading request needs to be decided on, a random number is generated $\in [0, 1]$ and the number is less than $p$, the resource transfer request is granted.

This work takes inspiration from Angourakis et al. [1]'s and Molin et al's [4] methods of generating resources every iteration. More specifically we blend is Molin et al.'s periodically induced environmental stress and Angourakis et al.'s restricting of resources between two predefined ranges respectively. (Our exact approach is detailed later)

*2) On what assumption is/are agents' decision model(s) based?:* The model assumes that resource trading preferences can be simulated as a stochastic process. Additionally, it assumes that all households were 'ruled' by a single individual and that personal storage is the preferred method of resource storing (as opposed to collective resource pooling or some other hybrid approach).

*3) Why is a/are certain decision model(s) chosen?:* Most of the model's input parameters are based on published works or publicly available data. Table I provides said references and where no references are made, NeoCOOP is tuned using *Optuna*.

*Optuna* performs multi-objective optimization maximizing total population and resources levels in the last iteration. These measures are used because a greater population level is indicative of a more successful parameter configuration and we also consider total resources because having a higher population level with more resources is a greater indicator of success than an equally large population with no resources.

*4) If the model/ a submodel is based on empirical data, where does that data come from?:* See Table I.

*5) At which level of aggregation were the data available?:* It varies from source to source. Table I clarifies how the data was derived.

### E. Individual Decision Making

*1) What are the subjects and objects of decision-making? On which level of aggregation is decision-making modelled? Are multiple levels of decision making included?:* The decision making units are Households.

When the vegetation model is present, agents choose to FARM or FORAGE actions, every iteration, equal to the number of able workers (An able worker is an agent who is older than or equal the *agent_of_maturity* property. Agents are restricted to choosing one action or the other, in fact, agents may choose to have some of their occupants farm, and the rest will forage.

In this work, agents claim a single land cell and gather resources from it directly (using an abstract resource acquisition method). Unless an agent move, it will just gather resources from the same cell every iteration.

*2) What is the basic rationality behind agents' decision-making? Do agents pursue an explicit objective or have other success criteria?:* Explicitly, the agents perform Utility maximization. This results in the agents implicitly trying to minimize their hunger and maximizing their social status.

*3) How do agents make their decisions?:* For resource acquisition, the agents follow the standard e-greedy approach in reinforcement learning. The agents use their hunger as $\epsilon$ to determine whether the should take a greedy action, or a random (exploratory) action. Agents will then update the utility values of each action based on the reward (food) received.

For resource transfer, the agents will look at their peer and subordinate resource transfer properties to determine if they are willing to donate resources for a given iteration. The first time an agent is asked (for both peer and sub requests), a random number $\in [0, 1]$ is generated and compared to the aforementioned resource transfer beliefs. If the random number is lower than the resource transfer beliefs, the agent will donate its resources.

For moving, the agent will look at its satisfaction and a generated random number (both $\in [0, 1]$). If the random number is greater than the satisfaction, the agent will move. Agents will first look at neighbouring settlements, if none of them look desirable (average resources of settlement ¡ required resources of agent), the agent will make a new settlement at random location. There is an additional property called lookback_sids which prevents the agent from going back to the same settlements until it has explored additional options.

*4) Do the agents adapt their behaviour to changing endogenous and exogenous state variables? And if yes, how?:* Yes, as outlined above agents will seek to explore alternative resource acquisition strategies when their current strategy does not work. Similarly, agents will move from one settlement to another when their overall satisfaction is low.

*5) Do social norms or cultural values play a role in the decision making process?:* Yes, if multiple agents decide to leave a settlement in the same iteration, they will all move to the same location.

*6) Do spacial aspects play a role in the decision making process?:* Yes, agents can only farm / forage within a specified max acquisition distance. Similarly, the distance an agent can travel when moving from one settlement to another can be controlled by the vision_square property.

For this work, these properties were set such that an agent could move to any settlement on the grid-world. Similarly, no penalties for the distance from the settlement and the agent's resource cell were applied.

*7) Do temporal aspects play a role in the decision making process?:* Yes, agents only decide whether or not to move every yrs_per_move iterations.

*8) To which extent and how is uncertainty included in the agents' decision rules?:* Not Applicable.

### *F. Learning*

*1) Is individual learning included in the decision process? How do agents' change their rules over time as consequence of their experience?:* Yes, agents typically follow a standard reinforcement learning approach to determine which action (farm or forage) to take (See submodels for more details). In this work this was disabled due to the removal of the vegetation model.

*2) Is collective learning implemented in the model?:* Yes, in the form of generational adaptation. Agents, using a genetic algorithm and a cultural algorithm to exchange information regarding the beliefs (See submodels for more details).

### *G. Individual Sensing*

*1) What endogenous and exogenous state variables are individuals assumed to sense and consider in their decisions? Is their sensing process erroneous?:* It is not erroneous and the agents don't explicitly detect any of the environments properties. What they are aware of though is their hunger, satisfaction and the perceived utility of the forage and farm actions.

*2) What state variables of which other individuals can an individual perceive? Is the sensing process erroneous?:* Agents are aware of the average resource levels of the neighbouring settlements. They are indirectly aware of the general beliefs held by their settlement (captured in their belief space).

*3) What is the spatial scale of sensing?:* Agents are able to sense the vegetation density (or resource availability) and cells they own within max acquisition distance cells around them.

*4) Are the mechanisms by which agents obtain information modelled explicitly, or are individuals simply assumed to know these variables?:* It is assumed.

*5) Are costs for cognition and costs for gathering information included in the model?:* Not explicitly.

### H. Individual Prediction

Agents do not make any explicit predictions.

### I. Interaction

*1) Are interactions among agents and entities assumed as direct or indirect??:* Cultural Influence occurs indirectly while resource transfer is direct.

*2) On what do the interactions depend?:* Resource transfer requires that agents belong to the same settlement. Cultural Influence depends on the social status of the two 'interacting' agents.

*3) If the interactions involve communication, how are such communications represented?:* Not Applicable.

*4) If a coordination network exists, how does it affect the agent behaviour? Is the structure of the network imposed or emergent?:* Not Applicable.

### J. Collectives

*1) Do the individuals form or belong to aggregations that affect and are affected by the individuals? Are these aggregations imposed by the modeller or do they emerge during the simulation?:* Yes. As mentioned above agents are may form settlements. The simulation does not enforce settlements (except at initialization). Agents may form new settlements, leave old ones or even move to other settlements every *yrs_per_move* iterations. An agent always needs to belong to a settlement (so that the adaptation submodels can work) but it is entirely possible that a simulation run may result in every agent forming their own settlement. This is equivalent to having no settlements since each household will adapt individually.

*2) How are collectives represented?:* As noted above. A collection of one or more households makes a settlement.

### K. Heterogeneity

*1) Are the agents heterogeneous? If yes, which state variables and/or processes differ between the agents?:* Yes. All variables listed in Section 2.B.2.

*2) Are the agents heterogeneous in their decision-making? If yes, which decision models or decision objects differ between the agents?:* Yes. All agent decisions are heterogeneous. This includes resource acquisition, resource transfer and relocation.

### L. Stochasticity

*1) What processes (including initialisation) are modelled by assuming they are random or partly random?:* All stochastic processes are pseudorandom. This is to ensure model reproducibility. A list of stochastic processes during model execution are listed below:

1) Monthly Global Environment (rainfall, temperature and solar radiation) values. This includes the direct resource generation approach used in this work.
2) Agent explorative action selection (e-greedy).
3) Agent farm / forage resource gathering patch selection.
4) Household birth.
5) Household death.
6) Resource Transfer Requests.
7) House Split Parent Selection (Genetic Algorithm)
8) Household Split Mutation (Genetic Algorithm).
9) Household Influence knowledge source selection (Cultural Algorithm)
10) Household Move decision.

A list of stochastic processes at initialization are listed below:

1) Settlement placement.
2) Agent initial gene creation.
3) Agent settlement placement.

### M. Observation

*1) What data are collected from the ABM for testing, understanding and analysing it, and how and when are they collected?:* Snapshots of the model are collected every (user-defined) iterations. These snapshots capture all of the necessary aspects for, essentially, recreating the simulation run from the ground up.

All environment data is collected in csv files. All agent and settlement data is collected in JSON files. A log file of the simulation is also recorded which records the result of every stochastic process the model simulates.

*2) What key results, outputs or characteristics of the model are emerging from the individuals? (Emergence):* This is heavily reliant on the input parameters of the model but the key emergent properties are list below:

1) Regardless of initial agent distribution, the agents mean resource trading beliefs (peer and subordinate) will converge to the range $\in [0.4, 0.6]$.
2) As environmental stress is increased, social stratification of the peer and subordinate transfer beliefs will occur.
3) Subordinate Ostracization occurs as the frequency of environmental stress increases.
4) As the frequency of environmental stress increases, so does agent attachment towards their current settlement.

### N. Implementation Details

*1) How has the model been implemented?:* The model was implemented in Python 3 using the *ECAgent* framework.

*2) Is the model accessible, and if so where?:* Yes, it is publicly available at: https://github.com/BrandonGower-Winter/NeoCOOP.

*O. Initialization*

*1) What is the initial state of the model world, i.e. at time t = 0 of a simulation run?:* Agents have been randomly allocated to settlements. Settlements have been randomly placed. All agents have zero resources and zero load. The rest of the initial conditions are based on the decoder file used to create the simulation.

*2) Is the initialisation always the same, or is it allowed to vary among simulations?:* It varies depending on the seed used. If the same seed is used, the initialization (and model execution) will be exactly the same.

*3) Are the initial values chosen arbitrarily or based on data?:* Randomly. If agent homogeneity is enforced, the model will not randomly assign agents to each settlement (all settlements will be given the same number of starting agents) but the settlement locations will still be random.

*P. Input Data*

*1) Does the model use input from external sources such as data files or other models to represent processes that change over time?:* Yes. *NeoCOOP* uses what we call *decoder* files. They share the same structure as table I (in JSON format) and are given to the model at initialization. The model also takes in a heightmap, sandcontent map and slopemap (usually derived from the heightmap). In this work, only a heightmap was needed which we just set to a flat plain.

*Q. Submodels*

*1) What, in detail, are the submodels that represent the processes listed in 'Process overview and scheduling'?:* Unlike most cooperation-based ABMs, *NeoCOOP* allows agents to make decisions based on their social status and the social status of the agents they are interacting with. In *NeoCOOP*, social status is defined as the sum of an agent's available resources and its *load* where *load* is the amount of resources the agent has donated over a period of time. To facilitate social stratification, we use the self-organization scheme described by [2] whereby a relationship type can be determined for every agent pair by comparing their social statuses. We define each of the relationship types as follows:

$$is\_acq(h_1, h_2) = h_1.settlement == h_2.settlement \quad (1)$$

$$is\_peer(h_1, h_2) = \frac{|h_2.ss - h_1.ss|}{max(h_1.ss, h_2.ss)} < L \quad (2)$$

$$is\_auth(h_1, h_2) = \frac{(h_2.ss - h_1.ss)}{max(h_1.ss, h_2.ss)} > L \quad (3)$$

$$is\_sub(h_1, h_2) = is\_auth(h_2, h_1) \quad (4)$$

Where $is\_acq$, $is\_peer$, $is\_auth$, $is\_sub$ describe whether household $h_2$ has an acquaintance, peer, authority or subordinate relationship with household $h_1$ respectively. $h_n.ss$ is a household's social status. $L$ is the *load_difference* $\in [0, 1]$ input parameter which describes how much more social status an agent requires to be considered an authority over another agent. Note that in order for a peer, authority or subordinate relationship to be formed, the two households must be from the same settlement (ie: $is\_acq = true$).

**Environment & Vegetation Model:**
*NeoCOOP* places agents on a $n \times n$ grid-world. Each cell on the grid contains resources $\in [0, 1]$ that are assigned to it every iteration. Stress is applied to these cells by varying the amount of collectable resources received each iteration according to *sine* waves of different frequencies. Denoting $f$ as the desired number of stress waves, we linearly interpolate (Equation 6) every iteration $i$ between two predefined ranges called $max\_resources = [0.4, 1.0]$ and $min\_resources = [0.0, 0.6]$ using the the output of the sine waves (Equation 5) at iteration $i/M$ as the mixing parameter $x$. This approach blends work by Molin, Kanwal and Stone [4] and Angourakis et al. [1] where environmental stress is induced periodically and between to predefined ranges respectively. This approach allows us to simulate a wide variety of stress scenarios ranging from short, but frequent, periods of stress (at high $f$) to longer, infrequent, periods of stress (at low $f$). Averaged over an entire simulation run, a household is expected to receive a total of 0.5 resources per iteration. An example of what the result of this process looks like can be seen in Figure 2.

$$s(x) = 0.5sin(2\pi.x.f) + 0.5 \quad (5)$$

$$lerp(r_{min}, r_{max}, x) = r_{min} + s(x) * (r_{max} - r_{min}) \quad (6)$$

The motivation for choosing a spatially explicit environment is because even ideal environments have a carrying capacity. Most spatially implicit ABMs do not consider population carrying capacity which limits their capabilities of accessing cooperative behaviour dynamics between two distinct population groups (those with and without direct access to resources).

**Resource Acquisition, Transfer and Consumption:**

Every iteration, agents gather resources from a patch of land that they own. The amount of resources gathered is equal to the full amount of resources available at said patch. These resources are then put into the agent's *storage*. In this work, agents are only allowed to own one patch of land. If an agent does not own any land, it will try to claim some by looking at its settlement's neighbouring cells. An agent that does not own any land will not receive any resources during the resource acquisition phase.

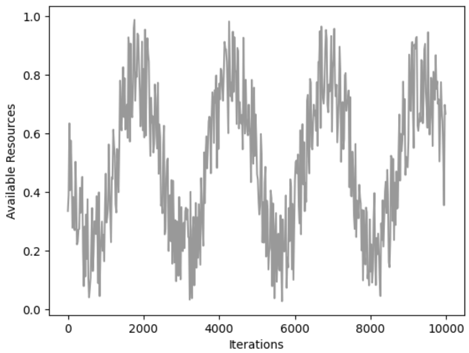



Fig. 2: Example of the available resources on a single environment cell over the course of the simulation run when $f = 4$.

Once acquisition is complete, agents determine if they have enough resources to satisfy their needs for the iteration. An agent needs to consume $0.5$ resources per iteration to avoid the risk of dying. If an agent does not have enough resources, it first asks its authority agents if they would be willing to donate some of their excess resources. For each authority asked, a random value $\in [0, 1]$ is generated and compared to the authority agent's *subordinate_transfer* property. If the generated value is less than the *subordinate_transfer* property, the authority agent is willing to grant donations for that iteration. Whenever a donation is granted, the authority agent has its *load* property increased by the resources donated. If an agent has asked all of its authority agents for resources and it will still go hungry, it then repeats this process for its peer relationships with the donating agent using its *peer_transfer* property to determine if the donation succeeds.

If that is still not sufficient, the agent will then ask all of its subordinates for resources. Given that we are modelling Neolithic households, if a subordinate is asked to give any of its excess resources to an authority agent, it does so with 100% certainty. The peer and subordinate transfer properties allow us to simulate agent types that exhibit varying degrees of altruistic and selfish behaviour. For example, an agent may exhibit nepotistic tendencies whereby it is is more likely to grant resource donations to its peers (high *peer_transfer*) but less likely to grant the same donations to its subordinates (low *subordinate_transfer*).

When resource transfer is complete, agents consume their resources and determine their *hunger* using equation 7.

$$hunger(h) = min(\frac{h.resources}{h.required\_resources}, 1.0) \quad (7)$$

**Population Growth, Loss and Migration**:

Every iteration, households may birth additional households in accordance with the *birth_rate* and their *hunger* (Equation 8). When this occurs, the *split_household* function is called and the household is divided into two separate households. Resources are split amongst the two new households but load is not. That is, the new household signifies the arrival of a new patriarchal figure in the community and one who must work to gain the same social status as their parent household.

$$birth(h) = random() < h.hunger * birth\_rate \quad (8)$$

Households may lose one or more occupants in accordance with the *death_rate* and their *hunger* (Equation 9). If a household dies of starvation, it is removed from the simulation.

$$death(h) = random() * h.hunger < death\_rate \quad (9)$$

Agents can migrate to another settlement or form a settlement of their own every *yrs_per_move* iterations. This decision is based on the agent's *satisfaction* and its *attachment*. *Satisfaction* is the average *hunger* of the agent over the past *yrs_per_move* iterations. The boolean function for determining if an agent will move is described by Equation 10.

$$\textit{move(h)} = 2\alpha_h * \textit{satisfaction(h)} < \textit{random()} \quad (10)$$

Where $\alpha$ is the attachment of household $h$ and *random()* returns a random value $\in [0, 1]$. If the *satisfaction* of the agent is low, it is more likely to move. This is partly mediated by the agent's *attachment* which when $< 0.5$, makes the agent skittish and when $> 0.5$ makes the agent more likely to stay at a given location regardless of its objective circumstances. In population migration research, the inverse of *satisfaction* is often called *grievance* [5].

When an agent moves, it chooses between all settlements in its vicinity or an unclaimed cell. Typically, an agent will move to the settlement with the most resources. However, if none of the neighbouring settlements have an average resource value $\geq 0.5$, the agent will choose to make its own settlement at a new, randomly chosen, location.

*Agent Adaptation:*
In this model, agent adaptation uses two evolutionary algorithms: a *Genetic Algorithm* (GA) [6] for vertical generational adaptation and a *Cultural Algorithm* (CA) [7] for horizontal generational adaptation. Both the GA and CA utilize the agent genotype described before and a concept called *influence*. *Influence* is used to determine best performing settlements and describes the probability that two settlements will interact with each other. This is done using XTENT [3] (Equation 11):

$$I(s_1, s_2) = W(s_2)^{\beta} - mD(s_1, s_2) \quad (11)$$

Where, $s_1$ and $s_2$ are settlements, $I(s_1, s_2)$ is the influence of $s_2$ on $s_1$, $W(s_2)$ is the social status of $s_2$, $D(s_1, s_1)$ is distance from $s_1$ to $s_2$. $\beta$ and $m$ are coefficients describing the required social status of one settlement to influence another.

Calculating the *influence* of every settlement on a given settlement, gives a probability distribution (equation 12).

$$P(s_1, s_2) = \frac{I(s_1, s_2)}{\sum_{k \in K} I(s_1, s_k)} \tag{12}$$

Where $P(s_1, s_2)$ is the probability of settlement $s_2$ influencing settlement $s_1$ and $K$ is the set of neighbouring settlements that have a positive *influence* value $I(s_1, s_k)$ on $s_1$.

The GA executes whenever the *split_household* function is called. The child agent produced is a combination of two parents with the first parent being the household that called the *split_household* function and the second parent gotten via *roulette wheel* selection [6]. This selection uses the social status of other agents within the same settlement of the first parent and from other settlements that have enough influence ($I(s_1, s_2) > 0$). The offspring agent is produced using *Uniform crossover* and random mutation.

The CA uses *Knowledge Sources* [8] to diversify how agents are influenced. These are:

- **Normative:** Influence on agent genes from its settlement.
- **Spatial:** Influence on agent genes from another settlement.
- **Domain:** Equivalent to GA mutation function, where domain influence mutates one of the agent's genes.

Every *influence_frequency* iterations, agents are influenced in accordance with the *influence_rate*. If an agent is selected for influencing, a roulette wheel is spun to determine from which knowledge source influence will come from. Influence from the Domain knowledge source occurs at a rate defined by the *mutation_rate* parameter. Influence from the Normative and Spatial knowledge sources occur with varying probability defined by equations 13 and 14.

$$N(s_h, s_i) = max(\frac{s_h.ss}{s_i.ss}, 1.0) \tag{13}$$

$$S(s_h, s_i) = 1.0 - N(s_h, s_i) \tag{14}$$

Where, $N$ and $S$ are the probability of choosing the Normative and Spatial knowledge sources respectively, $s_h$ is the settlement of the agent being influenced, $s_i$ is the settlement that would influence agent $h$. If the spatial knowledge source is selected. $s_i$ is determined by performing roulette wheel selection on all neighbouring settlements with a positive *influence* on settlement $s_h$. Roulette wheel weights are determined by the values returned by Equation 12.

Each settlement's beliefs are represented by *Belief Spaces* $B_s$. Belief Spaces have the same structure as the agent genotype with each property calculated using a weighted average of the corresponding property of all agents within that settlement. The weight an agent contributes to the belief space is determined using its social status relative to the social status of the other agents in the same settlement. If an agent is influenced by the *normative* knowledge source, the belief space that influences it is the belief space of the settlement the agent belongs to $B_{s_h}$. If the agent is influenced by the *spatial* knowledge source, the belief space that will influence the agent is the belief space of the settlement selected during roulette wheel selection ($B_{s_i}$). Agent properties are influenced as follows (equation 15):

$$G_{h,t+1}(p) = G_{h,t}(p) + \sigma_h(B_{s,t}(p) - G_{h,t}(p)) * \Phi(h, B_{s,t}) \tag{15}$$

Where, $p$ is the agent property (genes 1-4), $t$ is the timestep, G is the agent's genotype, $\sigma_h$ is the *conformity* of the agent, $B$ is the selected belief space ($B_{s_h}$ or $B_{s_i}$) and $\Phi$ is the Homophily term which returns a value $\in [0, 1]$ describing how similar the agent's genes are to the belief space that is influencing it. Homophily is a sociological principle that describes the tendency for individuals that are similar, either biologically or culturally, to gather together. The value of $Phi$ is 1.0 for interacting entities that have exactly the same genes, and close to 0.0 for entities whose gene values are further apart. This approach is similar to interaction probability in Axelrod's cultural dissemination model [9]. In our model, $\Phi$ limits the degree to which an agent is influenced if the belief space influencing it contains drastically different gene values. Formally, $\Phi$ is one minus the average absolute difference between the agent and influencing belief space's genes.

*2) What are the model parameters, their dimensions and reference values?:* See Table I

*3) How were the submodels designed or chosen, and how were they parameterised and then tested?:* Parameter tuning the model involved using values derived from other works and, where no parameters could be be found, we performed multi-objective optimization using *Optuna*. The optimization process ran for 119 simulation runs and final input parameters for our model can be seen in Table I. A report of the optimization process is outlined in Section 4.

## III. Experiment Design

Before running our experiments, we parameter tuned our model, a report of which is described in Section 4. Given our goal to find the environmental conditions under which social stratification occur, we ran our experiments as follows.

We first defined initial resource trading belief distributions for the agent types (denoted $[A, S, F]$). For purely altruistic $A$ initialization, agents have their peer and sub transfer properties initialized to $1.0$. For purely selfish $S$ initialization, agent peer and sub transfer properties are set to $0.0$ and the mixed population $F$ scheme initializes the agents' resources trading beliefs such that half of them follow the $A$ initialization scheme and the other half follow the $S$ initialization scheme. We use differing initialization schemes since the initial resource trading beliefs of an agent population may affect how they evolve over time.

When then defined the *stress scenarios* investigated as follows: $f \in [1, 2, 4, 8, 16, 32, 64, 128]$ . We also explore two scenarios in which environmental stress is confined to the range $[0.4, 1.0]$ (non-existent) and $[0.0, 0.6]$ (perpetual) . These two scenarios are denoted as $N$ and $P$ respectively. The motivation for choosing the aforementioned frequencies is based on preliminary experiments where it was observed that selfish behaviour could emerge at low $f$-values. We then expanded the scope of the experiments to include higher $f$-values to see if this trend persisted.

Using the three initialization schemes and 10 $f$-values, 30 scenarios where created. For each scenario, 50 simulations were run for a total of 1500 simulations across all scenarios. Each simulation was initialized with 100 agents and 10 settlements. At initialization, each agent in the model had their *peer_transfer* and *sub_transfer* agent properties set to either 1.0 or 0.0 depending on the initialization scheme (A scenario denoted as $16A$ indicates that the $A$ initialization scheme was used with an $f$-value of 16). Settlements were randomly placed on the grid-world and the model was run for $M = 10000$ iterations. All stochastic processes utilized a pseudo-random number generator to ensure reproducibility.

As highlighted in the results section of the paper, additional experiments were also run for the F initialization scheme for frequencies $f = [24, 40, 48, 56]$. The same process of 50 repeated realizations initialized with 100 agents and 10 settlements for each new scenario for a total of 200 additional simulation runs.

## IV. Parameter Tuning

### A. Optuna

*NeoCOOP* is parameter tuned using multi-objective optimization in *Optuna*. The process sought to maximize a simulation run's total population and total resources. The motivation for this is because a higher population level is indicative of a more resilient agent and because resources are essential to their survival, agents who are able to maintain a higher degree of surplus resources are more successful that agents that have not. Note that we do not care about the distribution of the resources per agent and only consider mean surplus resources across all agents. This is to ensure that we do not favour one type of organizational scheme over another (Authoritarian vs. Egalitarian)

The optimization process was intended to run for 150 simulations (31 of them did not complete because of a power outage) so the final results of the 119 that did finish are included below: (This can all be replicated using *Optuna.py*):

```
Pareto Optimal Solutions:
[{
    'influence_frequency': 6,
    'influence_rate': 0.0516647,
    'mutation_rate': 0.130285,
    'learning_rate_lower': 0.200033,
    'learning_rate_upper': 0.995248
}.
{
    'influence_frequency': 9,
    'influence_rate': 0.0573575,
    'mutation_rate': 0.0772241,
    'learning_rate_lower': 0.0683736,
    'learning_rate_upper': 0.125681
},
{
    'influence_frequency': 15,
    'influence_rate': 0.195384,
    'mutation_rate': 0.110884,
    'learning_rate_lower': 0.00644802,
    'learning_rate_upper': 0.856429
},
{
    'influence_frequency': 16,
    'influence_rate': 0.179273,
    'mutation_rate': 0.0444818,
    'learning_rate_lower': 0.0820841,
    'learning_rate_upper': 0.763823
},
{
    'influence_frequency': 16,
    'influence_rate': 0.111578,
    'mutation_rate': 0.0978724,
    'learning_rate_lower': 0.17061,
    'learning_rate_upper': 0.786016
},
{
    'influence_frequency': 18,
    'influence_rate': 0.111578,
    'mutation_rate': 0.0978724,
    'learning_rate_lower': 0.334526,
    'learning_rate_upper': 0.760217
}]

Average Results:
{
    'influence_frequency': 13.333333333333334,
    'influence_rate': 0.11780586666666665,
    'mutation_rate': 0.09310328333333333,
    'learning_rate_lower': 0.14367912,
    'learning_rate_upper': 0.714569
}
```

For simplicity, each value was rounded to nearest appropriate number:

```
Final Results:
{
    'influence_frequency': 15,
    'influence_rate': 0.1,
    'mutation_rate': 0.1,
    'learning_rate_lower': 0.15,
    'learning_rate_upper': 0.70
```

}

## V. Resource Transfer Belief Plots

As highlighted in our paper, a Wilcoxon rank-sum test ($p = 0.05$) revealed that for all $f \geq 8$, significant stratification occurred between the peer and subordinate transfer properties. Figures 3 to 12 showcase the evolution of each of the 30 *stress scenario* initialization scheme combinations investigated. In the these figures, the shaded regions represent 1 standard deviation from the mean.

As shown in Figures 7 to 11, clear visual evidence of stratification between the two resource transfer beliefs can be seen which agents evolving more selfish behaviour towards their subordinates at higher frequencies. In particular, oscillation of the mean subordinate transfer property in Figures 7 and 8 show the direct the effect environmental stress has on the emergence of selfish behavior towards subordinates. These figures show clear empirical evidence supporting the theory that selfish behaviour is is the result of frequent environmental stress.

As noted in the main paper, the magnitude of the stratification was not consistent across all scenarios. Figure 14 showcases this trend which suggested that peak stratification occurred somewhere around the $f = 16$ scenarios. This is what led to us conducting the additional experiments with the $F$ initialization scheme whose results are plotted in Figure 13. Again, the Wilcoxon rank-sum tests ($p = 0.05$) indicated significant stratification of the resource trading beliefs.

## VI. The evolution of household attachment

The last aspect of our results that we would like to discuss pertain to the evolution of the *attachment* property. Figure 15 showcases the evolution of this property for all 30 scenario combinations investigated. Attachment positively correlated with environmental stress frequency suggesting that in times of extreme stress, a high degree of mobility is favoured. When stress is frequent but, not harsh, attachment is high.

These results were very promising as the behaviour exhibited by these agents are the same as those described in Lewis et al. [10] where it was found that a sedentary lifestyle would lead to death in extreme environmental stress scenarios. Bands or groups would frequently move around to ensure they were maximizing their resource acquisition opportunities. Similarly, the bands were typically egalitarian which is exactly the type of behaviour we saw from our agents (In harsh environments, there was no significant distinction between the peer and subordinate transfer beliefs).

There is certainly future research opportunity here. Our current results show promise but, to efficiently study population migration dynamics of Neolithic studies, food storage and food storage strategies would need to be investigated [1]. *NeoCOOP* already supports investigating scenarios with varying food efficiency but, effort would need to be made to include cooperative food stocks whereby households may choose to donate some of their food to a group storage which any member of said group could access at any time.

## References


[1] A. Angourakis, J. Santos, J. Galán, and A. Balbo, "Food for all: An agent-based model to explore the emergence and implications of cooperation for food storage," *Environmental Archaeology*, vol. 20, no. 4, pp. 349–363, 2015.

[2] A. Chliaoutakis and G. Chalkiadakis, "Agent-based modeling of ancient societies and their organization structure," *Autonomous agents and multi-agent systems*, vol. 30, no. 6, pp. 1072–1116, 2016.

[3] A. Chliaoutakis and G. Chalkiadakis, "An agent-based model for simulating inter-settlement trade in past societies," *Journal of Artificial Societies and Social Simulation*, vol. 23, no. 3, 2020.

[4] L. Molin, J. Kanwal, and C. Stone, "Resource availability and evolution of cooperation in a 3D agent-based simulation," in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 93–101, 2021.

[5] A. Srbljinovic, D. Penzar, P. Rodik, K. Kardov, *et al.*, "An agent-based model of ethnic mobilisation," *Journal of Artificial Societies and Social Simulation*, vol. 6, no. 1, p. 1, 2003.

[6] A. Eiben and J. Smith, *Evolutionary Computing*. Berlin, Heidelberg: Springer, 2nd ed., 2015.

[7] R. Reynolds, "An introduction to cultural algorithms," in *Proceedings of the third annual conference on evolutionary programming*, vol. 24, pp. 131–139, World Scientific, 1994.

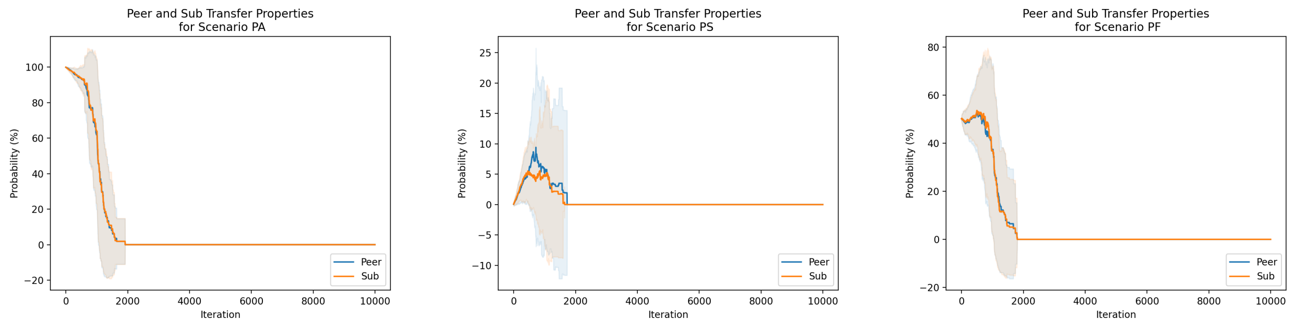
[8] R. Reynolds and B. Peng, "Cultural algorithms: modeling of how cultures learn to solve problems," in *16th IEEE International Conference on Tools with Artificial Intelligence*, pp. 166–172, IEEE, 2004.

[9] R. Axelrod, "The dissemination of culture: A model with local convergence and global polarization," *Journal of conflict resolution*, vol. 41, no. 2, pp. 203–226, 1997.

[10] H. M. Lewis, L. Vinicius, J. Strods, R. Mace, and A. B. Migliano, "High mobility explains demand sharing and enforced cooperation in egalitarian hunter-gatherers," *Nature Communications*, vol. 5, no. 1, pp. 1–8, 2014.

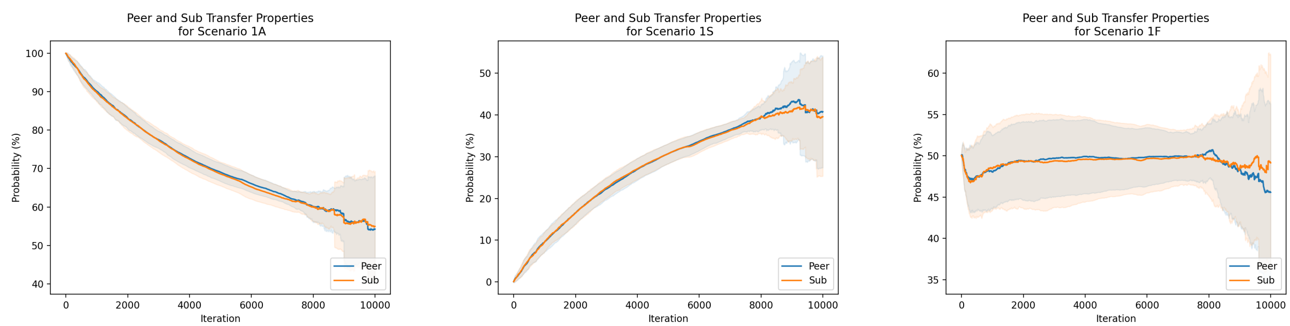
| | NeoCOOP | |
|---|---|---|
| Property | Value | Reference |
| | Model Parameters | |
| Iterations | 10 000 | |
| map_height | 100 | |
| map_width | 100 | |
| offset_x | 0 | |
| offset_y | 0 | |
| max_height | 1400m | |
| min_height | 0m | |
| cell_dimensions | 100m | This makes the area of a single cell = 1ha |
| | Global Environment System | |
| Priority | 10 | |
| min_resources | [0.0, 0.6] | [1] |
| max_resources | [0.4, 1.0] | [1] |
| interpolator_range | 10000 | Note: Must be equal to the number of iterations. |
| resource_interpolator | cosine | Set the frequency $f$ equal to the environmental stress frequency. |
| | Agent Resource Acquisition System | |
| Priority | 8 | |
| farmers_per_patch | 1 | Set to 1 because max occupants set to 1 |
| max_acquisition_distance | 50 | |
| storage_yrs | N/A | Not used in this study |
| | Agent IE Adaptation System | |
| Priority | 7 | |
| influence_rate | 0.1 | Parameter Tuned |
| influence_type | AVG | |
| persist_belief_space | true | |
| frequency | 15 | Parameter Tuned |
| | Agent Resource Transfer System | |
| Priority | 6 | |
| load_decay | 0.0 | |
| | Agent Resource Consumption System | |
| Priority | 5 | |
| storage efficiency | 1.0 | No resource decay. |
| | Agent Population System | |
| Priority | 4 | |
| birth_rate | 0.15% | |
| death_rate | 0.10% | |
| yrs_per_move | 5 iterations | . [2] |
| init_settlements | 10 | |
| cell_capacity | 100 | |
| | Agents | |
| number | 100 | |
| age_of_maturity | N/A | |
| consumption_rate | 0.5 | [1] |
| child_factor | N/A | Children eat less than adults, this takes that into account. |
| init_occupants | 1 | Occupancy dynamics not investigated in the work. |
| init_age_range | N/A | |
| vision_square | 10000 | So agents can see whole map. |
| move_lookback | 3 | |
| load_difference | 0.6 | [2] |
| learning_rate_range | [0.15, 0.7] | Parameter Tuned |
| conformity_range | [0.15, 0.7] | Parameter Tuned |
| mutation_rate | 0.1 | Parameter Tuned |
| b | 1.5 | [2] |
| m | 0.005 | [2] |

TABLE I: A comprehensive list of *NeoCOOPs* model properties.



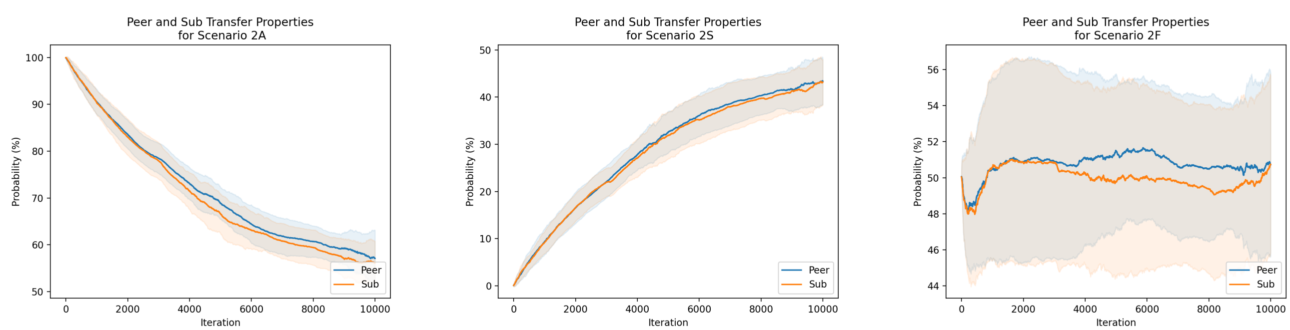

(a) (b) (c)

Fig. 3: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $PA$ (a), $PS$ (b) and $PF$ (c) scenarios.



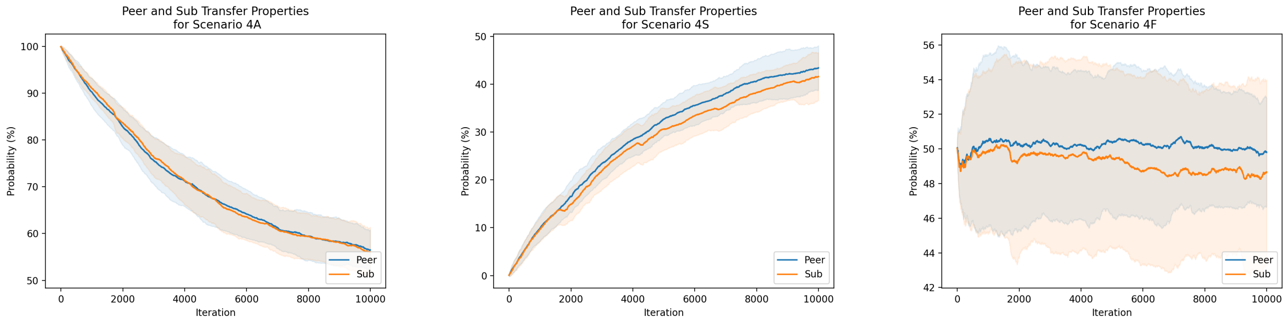

(a) (b) (c)

Fig. 4: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $1A$ (a), $1S$ (b) and $1F$ (c) scenarios.



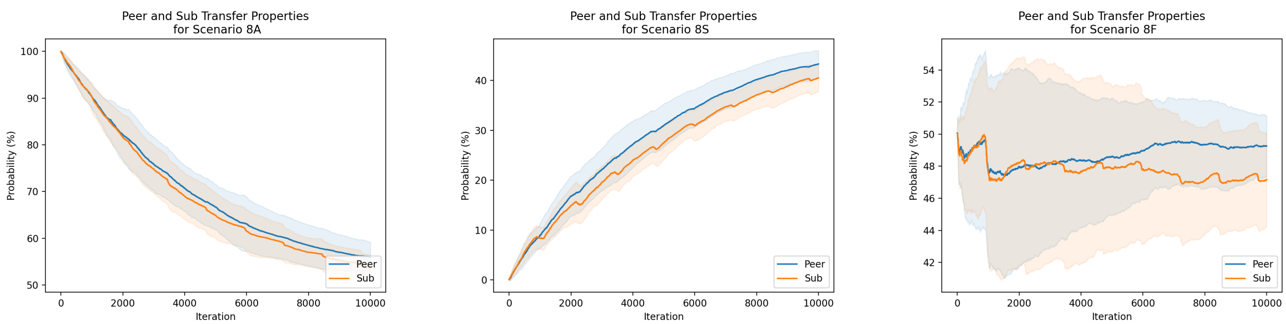

(a) (b) (c)

Fig. 5: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $2A$ (a), $2S$ (b) and $2F$ (c) scenarios.



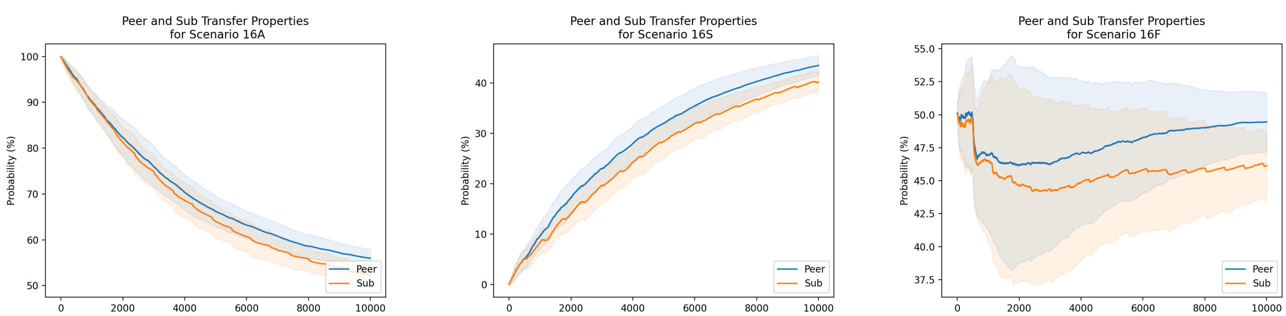

(a) (b) (c)

Fig. 6: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $4A$ (a), $4S$ (b) and $4F$ (c) scenarios.



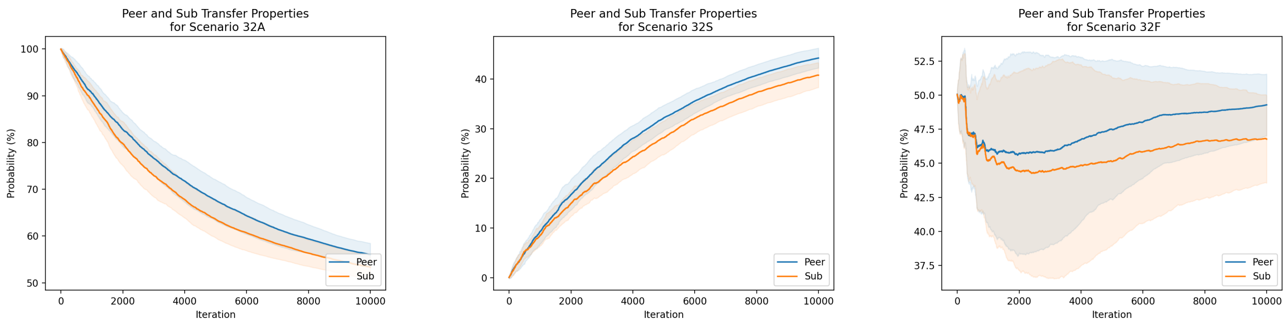

(a) (b) (c)

Fig. 7: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $8A$ (a), $8S$ (b) and $8F$ (c) scenarios.



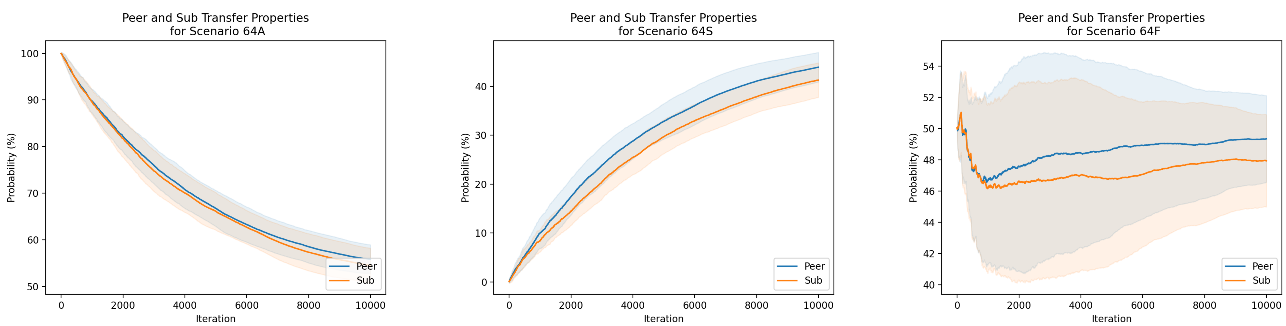

(a) (b) (c)

Fig. 8: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $16A$ (a), $16S$ (b) and $16F$ (c) scenarios.



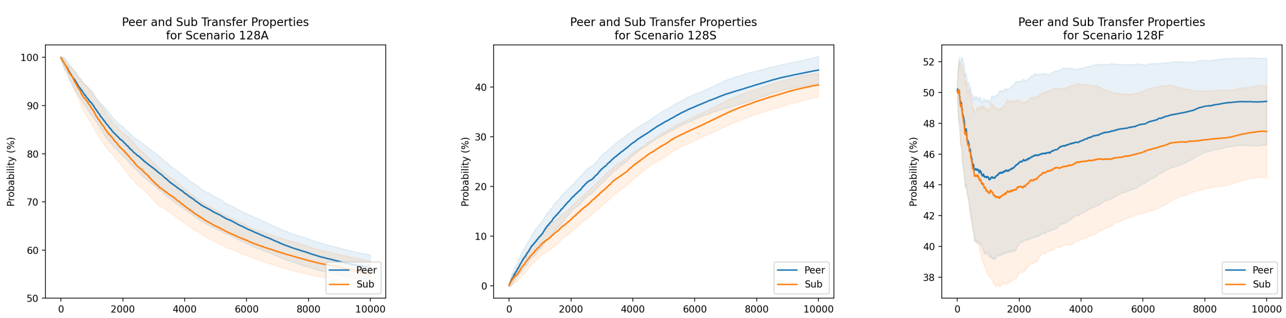

(a) (b) (c)

Fig. 9: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $32A$ (a), $32S$ (b) and $32F$ (c) scenarios.



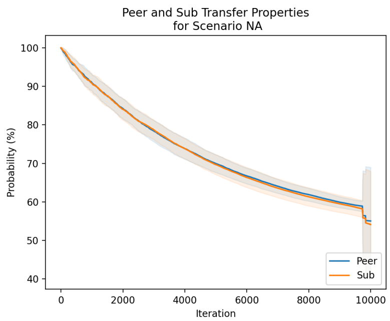

(a) (b) (c)

Fig. 10: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $64A$ (a), $64S$ (b) and $64F$ (c) scenarios.

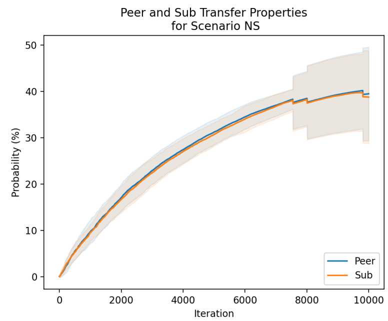



(a) (b) (c)

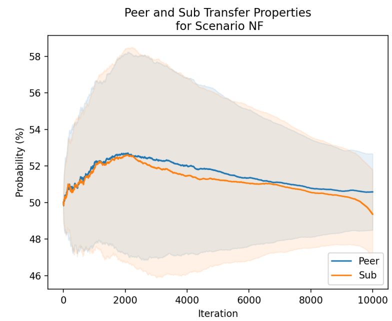
Fig. 11: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $128A$ (a), $128S$ (b) and $128F$ (c) scenarios.




(a)

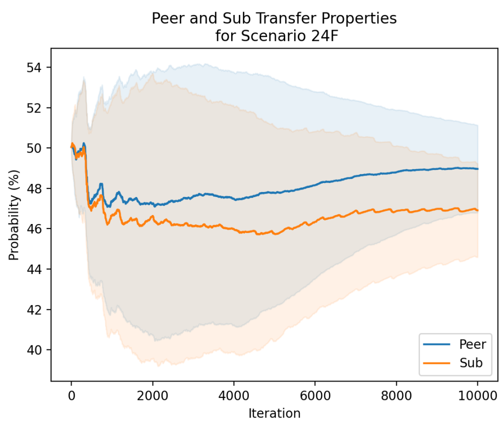



(b)

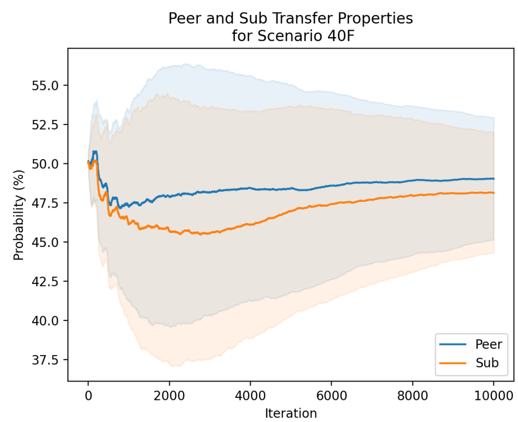



(c)

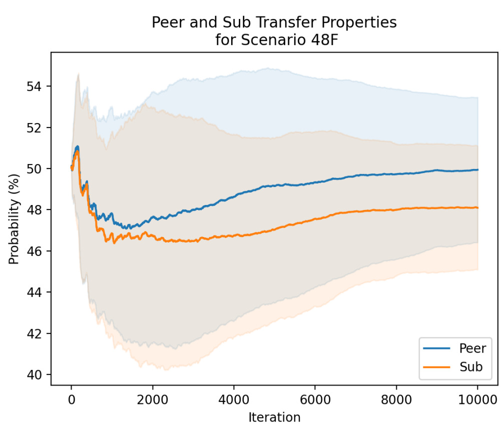
Fig. 12: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $NA$ (a), $NS$ (b) and $NF$ (c) scenarios.



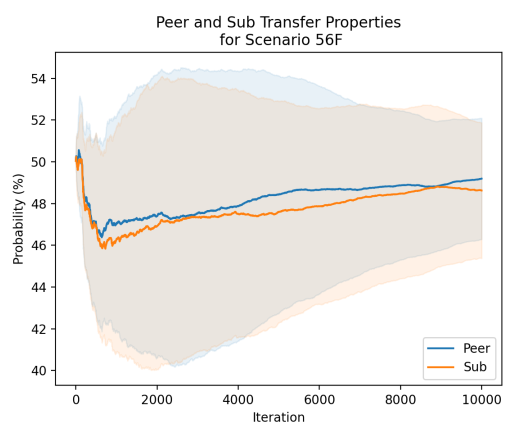

(a)




(b)

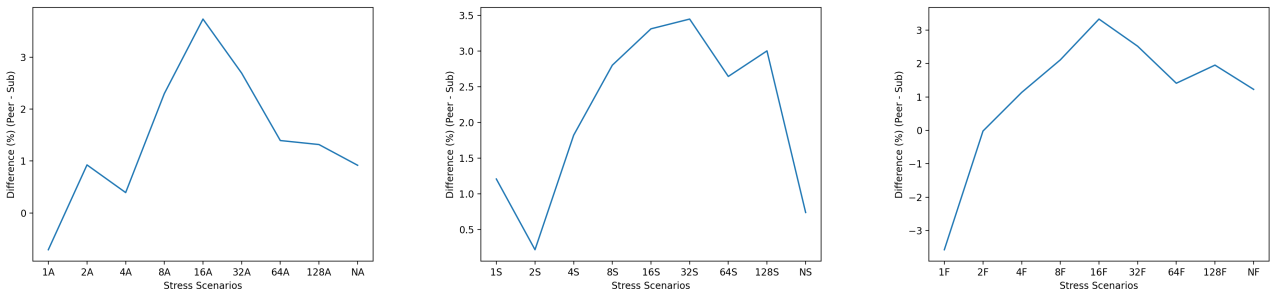



(c)



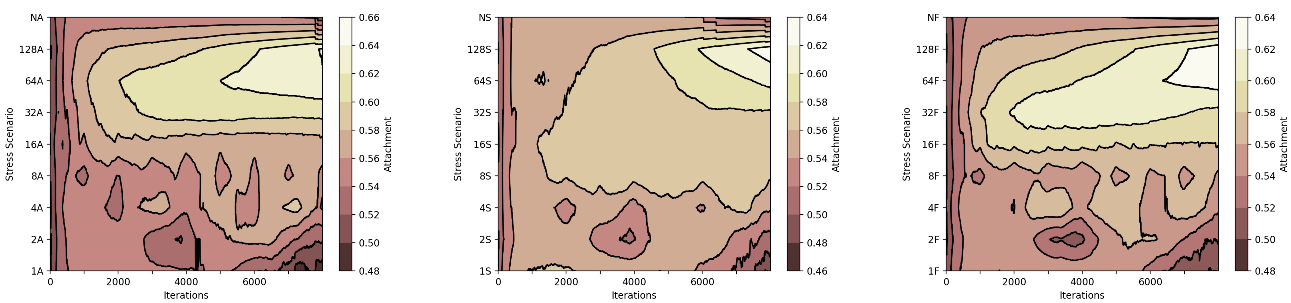

(d)

Fig. 13: The average value of the *peer* and *subordinate* transfer agent properties over the course of a simulation for the $24F$ (a), $40F$ (b), $48F$ (c) and $56F$ (d) scenarios.




(a) (b) (c)

Fig. 14: The average final difference of the *peer* and *subordinate* transfer agent properties (peer - sub) for all $A$ (a), $S$ (b) and $F$ (c) initialized agents across all *stress scenarios* investigated.




(a) (b) (c)

Fig. 15: The average value of the *attachment* agent property over the course of a simulation for all $A$ (a), $S$ (b) and $F$ (c) initialized stress scenarios.