Markets, Hierarchies, and Networks:
An Agent-Based Organizational Ecology

Appendix

To be published online with the article

This appendix is intended to give the reader a better knowledge of the structure and parameters of the Agent Based Model (ABM or the model) used in this paper. The appendix includes the expected utility calculations used by the agents, the default settings and a discussion of each parameter, as well as a more in-depth discussion of the working of the model.

Following the model discussion, a set of parameter sweeps are included. We sweep each of the substantive parameters across a nice population, a nasty population, and a moderate population. These sweeps support the findings detailed in the paper, as well as provide the reader with additional intuition as to the output of the model and other emergent properties not discussed in detail in the main paper.

Expected Utility Calculations

This section defines and explains the expected utility calculations that agents make when deciding to join a market, hierarchy or network. In addition to the user-defined parameters summarized in Table 1, agents are defined by their probability of cooperation (γ), which is either fixed (ALLC $\gamma = 1$ and ALLD $\gamma = 0$) or variable (TFT $\gamma = 0$ or 1). For purposes of calculating an agent's expected utility (as opposed to the actual payoffs defined above in the text), $k_{ij} = w(|p_i - Þ|/2)$, where Þ is the agent's belief (continuously updated) about the mean ideal point of the population. For the hierarchy, $k_{ih} = w|p_i - p_h|$.

In addition, the following endogenous variables are created and updated as the simulation unfolds:

β = the agent's belief about the cooperation rate of the population

σ = proportion of the population the agent has not already played

For each agent i:

Expected Utility in the Market:

$$M = \left(\gamma\beta R - k_{ij}\right) + \gamma\left(1-\beta\right)S + \left(1-\gamma\right)\beta T + \left(1-\gamma\right)\left(1-\beta\right)P$$

The payoff for a market interaction is essentially the probability of getting each outcome—based on the probability that the actor itself will cooperate (determined by their strategy type) multiplied by the probability that they believe their opponent will cooperate (determined by their beliefs about the cooperation rate in the population).

Expected Utility in the Hierarchy:

$$\theta\left\{\left(q^2R - k_{ih}\right) + q\left(1-q\right)S + \left[\left(1-q\right)qT - v\right] + \left[\left((1-q)^2\right)P - v\right]\right\} - \left(1-\theta\right)M - \tau$$

The utility for entering a hierarchy will depend on the proportion of the population in the hierarchy the player will join (θ), weighed against the likelihood of cooperation within the hierarchy (q), the punishment for defection (v), the tax (τ) and the ideal point of the hierarchy (p_h).

Expected Utility in the Network for Fixed Strategy Players:

$\eta(Z)+(1-\eta)(M)-\phi$

where η is the affinity rate in the network, and Z is the highest payoff in affinity memory *(m_a)*.

Expected Utility in the Network for TFT Players:

$$\eta(Z) + (1-\eta)\left(\sigma\left[\frac{m}{n-1}\cdot\left(\sum_{y=1}^{n}\beta\alpha^{y}\right)\left(\beta R - k_{ij}\right) + \left(1-\beta\right)P\right] + \left(1-\sigma\right)M\right) - \phi$$

The expected utility from the network is essentially the likelihood that the player is picked into the affinity world times the highest payoff in its affinity memory plus the likelihood that it is not and surveys the network. The value of the network is essentially likelihood that the player receives information about its current partner that changes its behavior (in most cases to prevent being suckered, or receiving the CD payoff) plus the likelihood it does not, less the fee imposed to join the network and gain information (ϕ).

Agents choose that organization with the highest expected utility in each round. Actual payoffs may differ from expected payoffs for any individual agent, but on average will be equal.

Model Description

We describe the ABM here in detail. As explained in the paper, there are three major stages in play: initialization, learning, and organizational choice.

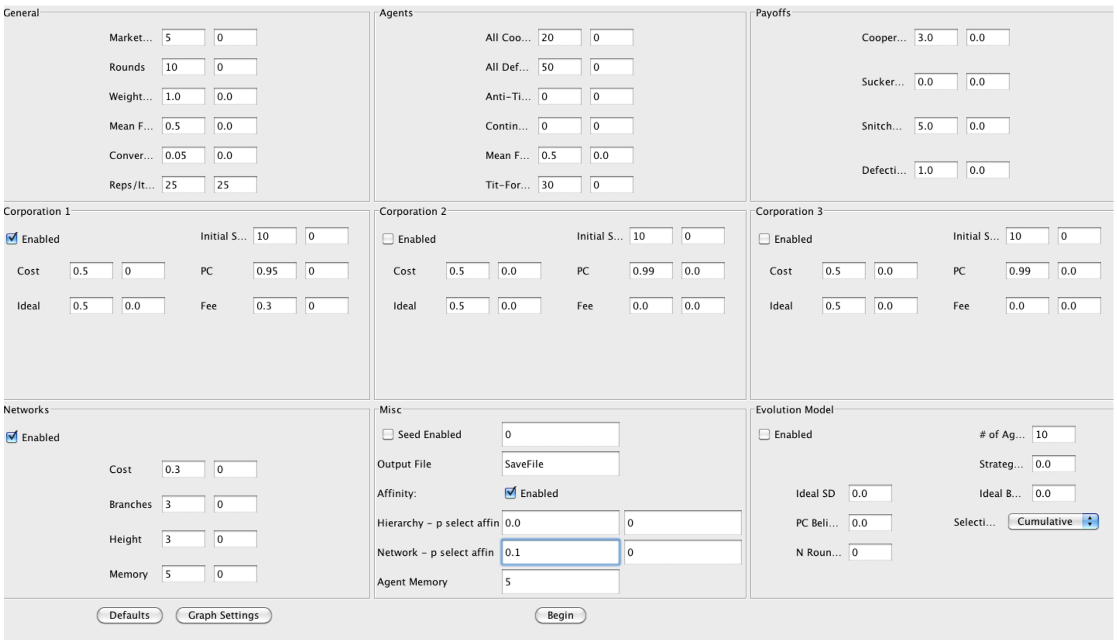


Figure A1. Screenshot of Initialization GUI.[1]

Initialization

The model begins with the specification of 24 user-defined parameters. These parameters and their default values, used in all the simulations presented below unless otherwise specified, are listed in Table A1 below (same as Table 1 in the paper). The default values for the parameters are admittedly arbitrary, but are calibrated here to make all organizational forms somewhat likely in any given simulation. By setting certain parameters higher or lower than our defaults, it would be trivial to simulate worlds in which either markets, hierarchies or networks always predominate or never arise.

[1] The full model that is shown here includes modules not activated for this paper. For example, the model includes but we do not “turn on” multiple hierarchies, selective affinity in the hierarchy, and evolution.

Table A1. User defined parameters and default values.

Parameter	**Symbol (for Expected Utility calculations**	**Description**	**Default Value**
General			
Increments		Number of times the simulation is run incrementing a parameter	20
Repetitions		Number of times the identical simulation is repeated with different random seeds	5
Rounds		Number of rounds of play	20
Mean for ideal point		Distribution of actors' policy preferences in population	0.5
Weight on ideal	W	Weight on policy preferences	1.0
Learning rounds		Set as either number of rounds or population convergence to within a proportion of the true population mean	5 rounds
Agents (Total)			100
All Cooperate		Number of actors of type always cooperate	
All Defect		Number of actors of type always defect	
TFT		Number of actors playing tit-for tat strategy	
Payoffs			
R	R	Payoff for CC outcome	3
S	S	Payoff for CD outcome	0
T	T	Payoff for DC outcome	5
P	P	Payoff for DD outcome	1
Hierarchy			
Initial size	θ	Proportion of the population in hierarchy.[2]	10
Penalty	V	Penalty for defection within the hierarchy	0.5
Probability of Cooperation	Q	Rate at which the agents cooperate with other agents in the hierarchy	0.95
Tax	τ	Tax assessed on members of the hierarchy	0.3
Ideal point	p_h	Ideal point of the hierarchy	0.5
Network			
Cost	ϕ	Fee for joining the network	0.3
Branches	α	Number of past cooperative partners	3

[2] In first round of play, this variable is set exogenously; after the first round, this variable is endogenous and defined as the number of players in the previous round.

		each agent *i* can ask for information about agent *j*	
Depth	L	Number of levels agent *i* can survey	3
Memory	m_n	How many past moves each agent remembers within the network	5
Selective Affinity			
Network Affinity	η	Probability of network players being able to pick their partner	0.1
Affinity Memory	m_a	How far back affinity players can look into their memory	5

General Parameters

The user sets six general simulation parameters.

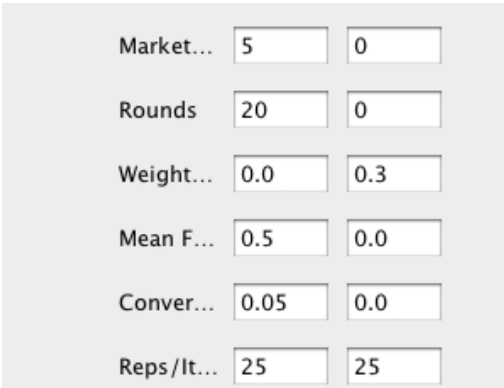


Image A1. General Parameter Input Screen.

The number of learning rounds is the number of rounds in the learning phase of the simulation. These actions are not recorded for output. Agents play only according to their strategy type. In essence, networks and hierarchies are turned "off" during these rounds. The rounds can be thought of as a standard PD without organizations. Learning rounds is set to 5 rounds by default. Agents will be matched randomly and play the PD 5 times in order to accumulate beliefs, as described in more detail below.

The number of rounds is the number of rounds of play in the simulation. Rounds must be set to at least 1, and the default value is set to 20. Agents will go through the process of selecting an organization, being matched, playing, and determining their payoffs from play 20 times in sequence with that setting. Also note that the first round is labeled 0, so graphing round 20 requires graphing the round labeled 19.

The weight on ideal parameter allows the user to place a weight on policy preference (*w)* below. This weight is used by agents to calculate their expected utility in the equations above. The default weight is set to 1.0. For example, if an agent with an ideal point of 0.6 cooperates in the market with an agent B whose ideal point is 0.5, and the weight is 1.0, the assessed ideological penalty is 0.05 (or half of the spatial distance between the two actors). If those same two actors cooperate in a world where the weight on the ideal point is 2.0, they are assessed an ideological penalty of 0.1.

The Mean for Ideal Point variable varies from (0,1). The default setting for this parameter is 0.5. Moving that closer to the minimum or maximum values allows us to determine how "moderate" or "extreme" the policy preferences of agents are. Agents' ideal points are picked from a normal distribution centered at that mean. Ideal points for individual agents also must be found within this range. If an ideal point is picked that is either above 1 or below 0, the maximum or minimum value is recorded. For example, if a population's ideal point is centered at 0.75, and an individual agent's ideal point is "picked" to be above the mean by more than 0.25, it will be automatically adjusted to have an ideal point of 1.0. This adjustment is most likely to happen when the mean for Ideal Point is set to a value that is relatively extreme.

The number of increments (Field Iter to the right of Reps) sets the number of times the simulation is re-run, adding the specified quantity to the relevant parameter. Increments allow us to run comparative statics on a statistically identical population. For example, in order study the effects of the cost of a network, we might set the initial network cost variable to 0.0. It will be incremented 20 times, by a value of 0.05. In the first iteration, the value of network cost is 0.0. In the second iteration, the simulation is run with a value of network cost at 0.05. This continues until the variable has been incremented the set number of times. Also note, while the increment variable is strictly positive (the minimum setting is 1), variables can be incremented by negative amounts (you can subtract value from a variable such as network cost at each iteration).

The number of repetitions is the number of times the same simulation is run again with a different random number. The default value here is 25 repetitions. The simulations are repeated multiple times in order to create confidence that the trends observed are not the result of one particularly idiosyncratic set of population dynamics but, rather, are repeated trends. If "Repetitions" is set to 1, the simulation is only run one time. If it is set to a value greater than 1, the entire simulation will run as many times as specified. Repetitions must be set to at least 1.

Repeating any simulation 25 times produces very reliable trends in our results. For illustration, we reproduce Figure 4a from the paper with 95 percent confidence intervals drawn around the results in Figure A2. The tight fit of this confidence interval around the mean (graphed) suggests that additional repetitions would be unlikely to alter the central tendency. Since the confidence intervals make the graphs even harder to read, especially in black and white, we do not include confidence intervals for other results reported in the paper or in the parameter scans below, but the intervals in those results are similar to those in Figure A2.

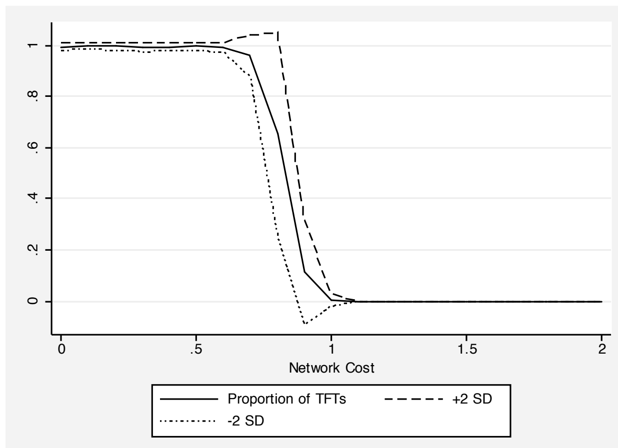


Figure A2. Shows network membership amongst TFT agents as Network cost increases. This figure matches figure 4a from the paper. 95% confidence intervals are drawn around the results.

Agent Parameters

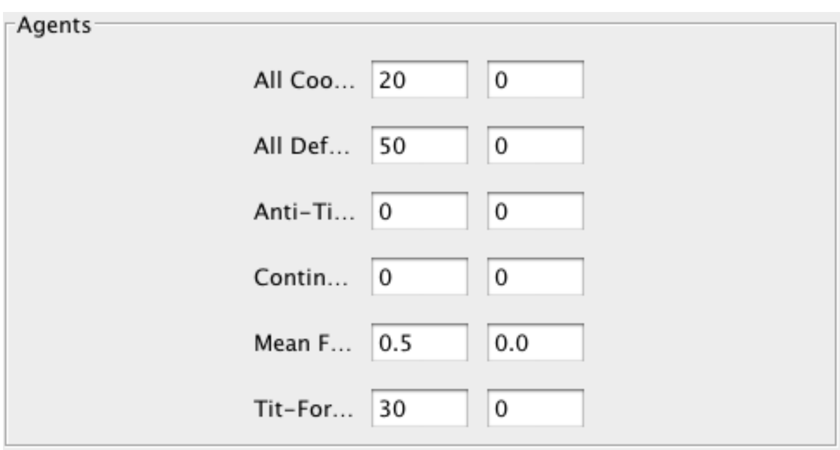


Image A2. Agent specification

The user defines the population of actors, specifically the distribution of strategy types, and their preferences. Like other game-theoretic ABMs, agents are defined by their strategy types. The default total number of agents is 100. This is done for analytical ease. Some other default settings are premised on a default population of 100. For example, we expect the initial or "advertized" size of the hierarchy to be sensitive to the size of the population. The proportion of agents in the hierarchy will affect the expected utility calculations. Set at a default value of 10, the initial advertised size of the hierarchy creates a 10% likelihood of encountering another hierarchy member in a population of 100 agents. In a population of 25 agents, this would be 40%, in a population of 1000, this would be 1%. These proportions will affect organizational choice significantly.

ALLC is the number of agents who are hardwired to cooperate with certainty in the market or network. No matter whom this agent is paired with, it will always cooperate. The only circumstance in which we ever observe an ALLC agent defecting is in the hierarchy when it is randomly chosen to defect at (1-q), where q is the cooperation rate in the hierarchy.

ALLD is the number of agents who are hardwired to defect with certainty in the market or network. No matter whom this agent is paired with, it will always defect. The only circumstance in which we ever observe an ALLD agent cooperating is in the hierarchy, when it is induced to cooperate at a rate of q, where q is the cooperation rate in the hierarchy.

Tit-for-Tat (TFT) is the number of agents who will pursue a tit for tat strategy. A TFT agent by default will cooperate on the first round of play with an unknown agent. If it has played an agent before, it will do what its opponent did in the last round: if its opponent defected in the prior round, it will defect in the current round; if its opponent cooperated in the prior round, it will cooperate in the current round. We expect to observe TFT cooperation and defection in all three organizations. When in the hierarchy, TFTs like ALLCs and ALLDs will cooperate at a rate of q and defect at a rate of 1-q.

For simplicity, in all the simulations reported in the paper we include only varying combinations of ALLC, TFT, and ALLD strategy types. We find that altering the composition of the population merely within these three very simple strategies yields considerable variation in the organizational ecology. However, we have built in two other strategies. In light of the other two strategies that can be specified in the GUI—MIX (or continuous) and Anti-Tit-for-Tat (ATFT)-- we describe them here for completeness as well as to pre-empt questions about their categories in the GUI. Future work will include these strategy types.

For completeness, MIX agents cooperate probabilistically at a specified rate. That rate is determined in the manner that ideal points are drawn. The user specifies the mean of a normal distribution; cooperation rates for MIX agents are picked from that distribution. The process by which MIX agents behave is slightly different, but follows a similar process. For example a MIX agent with a probability of cooperation 0.6 will cooperate on average 6 times out of every 10 rounds. Observed cooperation may differ from this mean, but on average will converge to that mean. A number from 0,1 is picked, if it is above the rate of cooperation the agent will defect for that interaction. If the number drawn is less than or equal to the agent's cooperation rate, the agent will cooperate for that interaction. For example given an agent's cooperation rate is "set" at 0.6, if the number generated is 0.0023 then the agent will cooperate. If the number generated is 0.8324 then the agent will defect. The likelihood of 0.6000 being drawn exactly is very small, but in that event the agent will cooperate. When in the hierarchy, MIX like all other strategy types will cooperate at a rate of q and defect at a rate of 1-q.

Anti-tit-for-tat (ATFT) agents start off by defecting on the first round with an unknown player. If an ATFT agent has played its opponent before, it will do the opposite of what its opponent did in the prior interaction: if the opponent defected in their prior interaction, the ATFT agent will cooperate. We expect to observe ATFT cooperation and defection in all three organizations. When in the hierarchy, ATFTs like all other strategy types will cooperate at a rate of q and defect at a rate of 1-q.

Payoffs

Payoffs for the various outcomes are set: T, R, P, and S. We set the default cardinal payoffs in the PD game as in Axelrod (1984) for purposes of comparability. All other default parameters were then set relative to these default payoffs. The payoffs in the model can be manipulated to create any 2x2 game by changing the ordering of the payoffs.

R is the payoff a player gets for a mutually cooperative (CC) outcome. R is set by default to 3.

S is the payoff a player gets for the "sucker" outcome (CD). S is set by default to 0.

T is the temptation payoff; the payoff a player gets for suckering another (DC). T is set by default to 5.

P is the payoff for mutual defection (DD). P is set by default to 1.

The preference ordering T>R>P>S is consistent with the traditional conception of the Prisoner's Dilemma.

Image A3. Payoffs Specification

Organizational Parameters

Image A5. Network Specification

The organizational parameters are also set at the initialization stage. Networks are defined by their width (α) (labeled Branches in the GUI), the number of other agents each agent can directly ask about the agent it has been randomly paired with, and their depth (l) (labeled Height in the GUI), the number of levels of agents that are polled. A 3x3 ($\alpha = 3$, $l = 3$) network is illustrated in Figure A3. Although each agent has a potentially infinite memory of its own interactions with each other agent in the population, the network is limited to a fixed memory (m_n) defined by the number of previous rounds over which it can poll. That is, if memory is set at five, any agent can poll only those agents with whom it has cooperated in the last five rounds whether they have interacted with the other agent with whom it has been randomly paired in the current round. The longer the memory (the larger is m_n) for the network, the more useful information it returns to the agent.

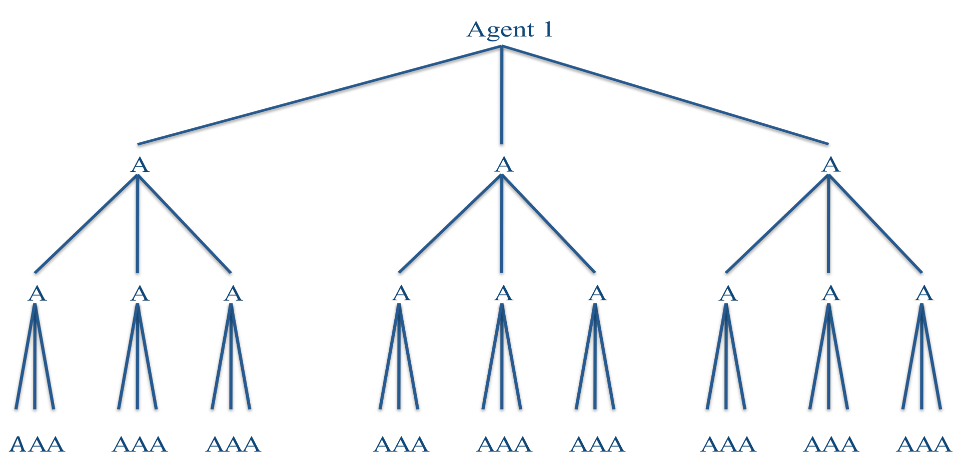
The width (α), the number of other agents each agent can directly ask about the agent it has been randomly paired with is set by default to 3. This means agent *(i),* the surveying agent, will go into its memory and ask the three most recent partners with whom *(i)* has had a cooperative interaction if they have any information about the behavior or ideal point of agent *(j)*. If this number is set to 5, agents will go back into their memory and ask up to five partners.

The number of levels (l) along which polling takes place indicates the depth of the network. The parameter *(l)* is set by default to 3. This means that those three actors whom agent *(i)* surveyed will survey their most recent three cooperative partners, and those agents will survey their three most recent cooperative partners about that same agent *(j)*. See Figure 2 below for a visual representation of this default network.

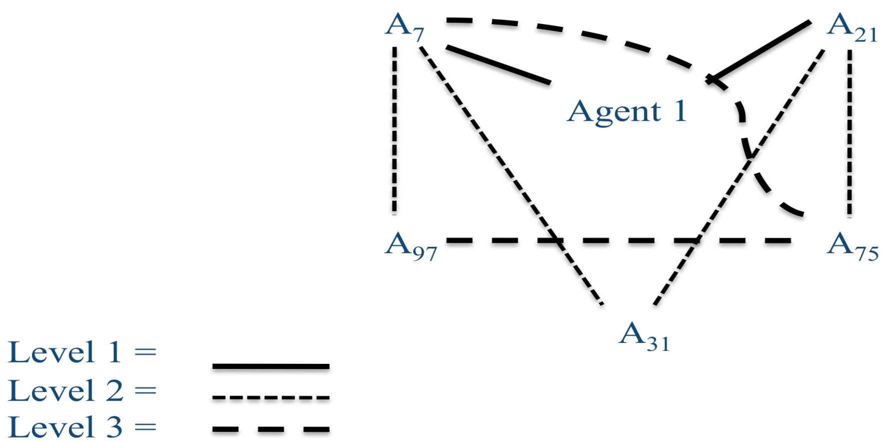
Agents have a fixed memory for the purpose of the network. By default agents remember their past five partners, and will poll only those with whom they have cooperated (CC) within the last five rounds. In allowing an agent's own memory of past play and the network's "memory" to differ, we are essentially assuming that an individual's memory of others lasts longer than that individual's social interactions. This seems reasonable. Those of us who hold grudges and have only fleeting friendships typically remember others who have treated us badly in the past longer than we engage in sets of social relationships. This assumption is consequential only for the declining utility of networks discussed in the paper. If agent memory were limited to the same as the network memory, networks would remain more robust over more rounds of the game. Conversely, without this restriction on network memory, the network would return "too much" information in early rounds and become obsolete almost immediately. If an individual receives no information from the network, it plays its default strategy.

The cost for any agent *i* who wishes to use the network is set. This parameter ϕ is set by default to 0.3.

Figure A3. Network Structure within the ABM



Panel a. A network of width (α) 3 and depth (*l*) 3. Agent 1 polls up to three other agents with whom it has cooperated in the last m_n number of rounds, who poll up to three other agents with whom they have cooperated in the last m_n number of rounds, who poll up to three other agents with whom they have cooperated in the last m_n number of rounds whether they have ever played the agent with whom Agent 1 is randomly paired. In a 3x3 network, Agent 1 will receive a maximum of 39 responses (depending on the number of agents each agent has cooperated with in the past *m* rounds). The larger the width or depth of a network, the more responses Agent 1 will receive. In addition, as each agent can only poll those agents with whom it has cooperated in the last *m* rounds, the more cooperative the population the greater the number of responses Agent 1 will receive, on average.



Panel b. A hypothetical network with limited memory and redundant pathways. Agents polled need not be unique as in Figure 1a, and in a finite population most likely will not be (as an agent at level 1 may well be an agent at level 2 or 3 in another branch of the tree). In a population with limited numbers of cooperators and a short memory, it is unlikely that agents will cooperate with more than a small number of other agents. In this case, the number of unique responses is likely to be quite small. A hypothetical network illustrates this point. Levels correspond to those in panel 1a. In this example, Agent 1 would receive a total of eight responses but only 5 unique responses).

Selective Affinity

In order to account for the propensity for actors to play and want to play with others with whom they have interacted in the past, we have included an extension to the main model that allows for selective affinity. This extension can be switched on or off. If it is switched off, the model functions as described above. The basic intuition of the way we model affinity is that in the network a player gets to pick the player with whom it has had the best payoff in recent memory. Networks can have an informational role, as well as an affinity role.

To initiate selective affinity, the user sets core parameters. The user specifies the affinity rate in the network (default is 0.1, or 10 percent of interactions for network agents) as well as the agent's affinity memory (5). If Selective Affinity is enabled, but the affinity rates are set to 0.0 the model functions essentially as in the non-affinity world. The restriction on memory causes minor differences in the model, but does not affect the results in a meaningful way. Note that selective affinity for the hierarchy is never enabled in any of the simulations reported in the paper or the parameter scans below.

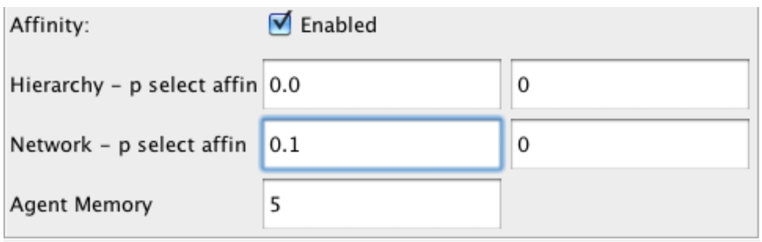


Image A6. Selective Affinity Core Parameters

Hierarchy parameters

A hierarchy is defined by its assigned ideal point (p_h), the probability that any agent will cooperate with other agents in the hierarchy (q), the penalty that is imposed on agents for defecting on other agents in the hierarchy (v), and the tax assessed on members (τ). These parameters are common knowledge to all agents. Since the expected utility for joining the hierarchy is contingent on the number of other agents in the hierarchy (θ), in the first round of organizational play the user sets an "advertised" number of agents in the hierarchy, which need not be the same as the actual number of agents who join. In subsequent rounds, agents know the actual number of agents who joined the hierarchy in the previous round.

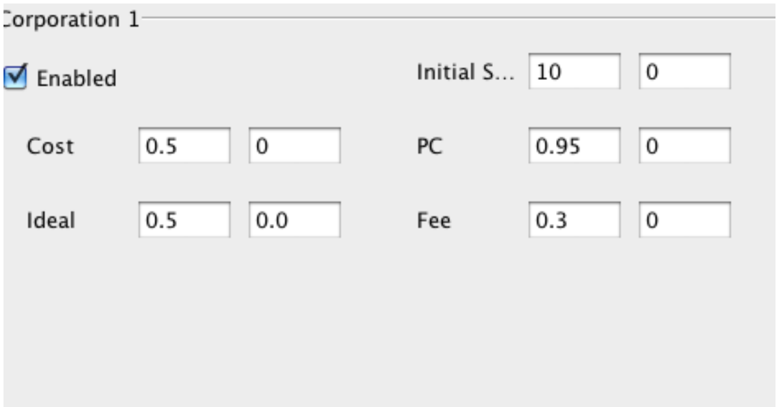


Image A7. Sample hierarchy (corporation) characteristics input panel.

The size of each hierarchy (θ) is initially is set to 10. After the first round, the value of θ is set to the actual number of agents who joined that hierarchy in the previous round. The larger the initial size relative to the population size, the more attractive the hierarchy becomes. Hierarchy essentially creates enforced probabilistic cooperation. This cooperation becomes more likely and therefore more valuable as the proportion of the population in the hierarchy grows larger. Additionally, if θ is initially set to 0, or if, during the play of the game, the actual membership of a hierarchy goes to 0, no players will join in the first round or in subsequent rounds. At this point that hierarchy is effectively "dead." Although no agent will ever see it as advantageous to join an organization that has no members to enforce cooperation internally, even with a tax of 0, agents will always continue to calculate the expected utility for joining.

The ideal point of the hierarchy (p_h) is set by the user. The default value of this is 0.5. This variable ranges from (0,1). The placement of the hierarchy's ideal point is very important. This is the value at which all "induced cooperation" takes place— no longer the mid-point between paired actors. It should be noted that this point is not assigned in the manner that players' ideal points are assigned, pulled from a distribution; the ideal point for the hierarchy is assigned for this particular hierarchy at that specific point.

The penalty for defection (cost in the GUI) within the hierarchy *(v)* is set to a default value of 0.5. This is the penalty for (probabilistically) defecting on another member within the same hierarchy. Essentially if an agent "suckers" another member of their hierarchy, the specified penalty is assessed by the hierarch. Though very unlikely, this payoff is also assessed in situations where both players defect. Most of values of internal hierarchic cooperation that we look at are well above 0.9, meaning even at this very low threshold, the likelihood of a mutual defection occurring within a hierarchy is 0.01—*within hierarchic interactions*, which are a subset of total interactions. As the rate of cooperation increases to the default 0.99, this probability of mutual defection shrinks to 0.0001

The probability of cooperation (q) within the hierarchy is set to 0.95. If two agents are paired within the hierarchy, they will cooperate with this probability. A high default cooperation rate makes the choice of a hierarchy a near if not absolute certainty.

The tax (fee in the GUI) assessed to members who join the hierarchy is analogous to the fee for information in the network. The tax variable τ is set by default to 0.3

Learning

Agents begin the simulation without any knowledge of the distribution of the other agents' strategies or the mean ideal point of agents in the population. In the learning phase, agents are randomly paired with other agents with whom they then play a round of the game according to their fixed strategy type with payoffs as specified. Each agent is an Agent *i* and matched with an Agent *j* randomly. Some agents may be picked multiple times as an Agent *j* within the same round of play. Agents record and remember their interactions as both Agent *i* and Agent *j*.

During this period there is no organizational choice. Networks and hierarchies are turned off—there is no enforced cooperation, selective affinity, or purchase of information. Over the course of these rounds, each agent develops beliefs about two parameters from their interactions with other agents: about the distribution of strategy types and ideal points in the population. Agents play their strategies: ALLC agents are matched and always cooperate, ALLD agents are matched and always defect; TFT agents will cooperate against an Agent *j* with whom they have never played, if a TFT agent has played an Agent *j* before it will do the same thing the Agent *j* did in the last round.

Agents learn about the distribution of other strategy types. Observing their own payoffs, they then back out whether the other agent cooperated or defected, store this action in memory by agent, and update a running estimate of the proportion of cooperators and defectors in the population (β_i). From this, agents learn whether the environment is relatively nice or nasty. If in the (default) five learning rounds an agent experiences two defections and three cooperative interactions, its belief about the cooperation rate in the population is set to 0.6. If there were two cooperative interactions in those five, its belief is 0.4. Essentially agents keep a running average of cooperative over total interactions.

Importantly, agents observe only the other's actions, limited to cooperation or defection, and not their underlying types. Thus, each agent assigns and then subsequently updates each agent it plays a single running probability of cooperation. An agent playing an ALLD agent twice would have a running probability of cooperation for that agent set to 0 since it had observed no cooperative outcomes in the number of interactions.

Second, when they cooperate with other agents, agents also learn about the distribution of preferences in the population and whether their own preferences are relatively extreme or moderate. Again, knowing only their own preference, agents who cooperate with one another examine their payoffs and back out the ideal point of the other agent, store this knowledge in memory, and then update their beliefs about the mean ideal point in the population (Þ). In a system analogous to the one above, agents track the ideological penalty to create a running tally of the ideal points known (which it backs out given the ideological penalty) over the number of mutual cooperative interactions (the only instances where ideal points are revealed).

In this phase of the simulation, agents are restricted to the knowledge they accumulate about other agents through direct play. Interactions in this phase are equivalent to what we later call market interactions. Intuitively it is best to think of these interactions as just a simple iterated PD—or other iterated two-player game. As mentioned above, many different types of iterated games can be represented in this format. The payoffs here are always ordered to meet the conditions for a prisoner's dilemma, but it can also capture pure coordination games, or games like battle of the sexes or stag hunt.

In this article, for simplicity, the number of rounds for the learning period is arbitrarily fixed at five. Each agent develops unique beliefs over the course of play, meaning that even agents with the same strategy type and similar or identical ideal points will make different organizational choices in the next stage. Agents who believe the population is nastier than it really is are *pessimists* and agents who believe the population is nicer than in actuality are *optimists*.

Organizational Choice and Play

Once the learning period is concluded, the main simulation of interest begins and continues for a fixed number of rounds. In this phase, a round is defined by two actions: the organizational choice of each agent for that round and the actual play in that round. Agents begin each round by calculating their expected utility for joining each type of organization and select the one they calculate will yield the highest return. The expected utility for market interactions is the same as an agent would get in play during the learning phase described above.

Agents calculate their expected utility for all organizations (as defined above). They enter into the organization with the highest expected utility. After agents choose the organization they will join for that round, the next stage is actual play within each organization. As in the learning phase, agents are randomly paired with another agent for that round (except for selective affinity).

If a player selects the market it plays its fixed strategy. It gets the payoffs associated with that outcome (set above) and, if it's a mutually cooperative outcome, the payoff is adjusted for an ideological penalty. For example, if two ALLC players cooperate—a likely situation—their payoff, by default 3 each, will be adjusted by the distance between their ideal points multiplied by the weight. For example, if their ideal points were 0.4 and 0.6 and the weight on the ideal point is 1, each player would receive a net payoff for that interaction of 2.9, or 3.0 less 0.1.

If a TFT agent selects the network, it will query the specified past cooperators about the agent with whom it has been randomly paired and be given a number [0,1] representing the probability of cooperation to expect from that partner. If that agent believes the other agent is likely to cooperate (the probability is ≥ 0.5), it will cooperate, otherwise the agent defects. The information returned from the network is treated as equivalent to the agent's own beliefs about the randomly paired agent acquired through direct play. That is, if agent *i* has no past play with agent *j*, and it receives a signal from the network that *j* cooperates 0.7, it will update its belief about *j*'s type to 0.7. Similarly, if *i* believes on the basis of a single past interaction that *j* cooperates 1.0 and it receives a signal from the network that *j* has cooperated with five networked agents at a rate of 0.7, it revises its belief about *j* to 0.75—weighting its own experience equally with those received from the network. In this way, we assume that all agents are sincere in their reporting and are known to be so by all other agents. Agents receive the payoff associated with the outcome, adjusted for weighted ideological penalties where applicable and the network cost.

Agents of all types including fixed strategy types who choose the network are selected into the affinity world at the specified "rate." For example, if an agent selects the network, and the probability of being chosen into the affinity world is 0.1, it has a 10% chance of being able to choose its partner. If an agent is selected into the affinity world and has no memory (only likely in the first round), it will play in the "regular" world, and be matched accordingly (i.e., randomly from the population). Agents with a longer history of play will be advantaged (specifically where the number of rounds is greater than or equal to the memory in the affinity world), as they are then more likely to find a nice agent within their memory. There may be situations early on where TFT agents have an unlucky draw and have been paired with only ALLD agents. In this case, their "best" partners are nasty ones, but they must select and play one accordingly. As the simulation progresses this should even out. Agents calculate their expected utilities for each organizational form, as above, but using the new equations to take the possibility and probability of affinity into account. Agents selected into the networked affinity scan their memory (within the set limit) to review their past interactions. They choose the agent with whom they have interacted and received the highest net payoff (less ideological penalties and fees if appropriate). These may be agents from the network, hierarchy, or the market. This agent may also be one that they have exploited (DC) in the past.

If agents with fixed strategies choose the network and are not selected into the affinity world, they are matched and will play as in the market, however they still pay the network fee. If a TFT agent chooses the network but is not selected into the affinity world, they will poll and will use the networks informational capacity to make their decision. Agents in the non-affinity world choose their partners at random, as in the "regular" model. After agents play, payoffs are assigned, and adjusted for organizational fees and ideological penalties.

If the agent chooses to join the hierarchy, its play depends on whether or not it is matched with another player in the hierarchy. If the two players belong to the same hierarchy, the agent will cooperate at the rate that the hierarchy enforces (q). If the agent defects ($1-q$), it will be punished at the defined level (v). If a player is matched with a player outside of its hierarchy, it will play as if it were interacting in the market. For example, if two agents (ideal points 0.3 and 0.6) within the hierarchy are matched, first they determine if they will cooperate at the assigned rate (by default 0.95), assuming both fall below the randomly generated cutoff, both will cooperate. They receive a default payoff of 3.0. Assuming the hierarchy's ideal point is 0.5 (default) and the tax is 0.3 (default), the first agent will receive a net payoff of 2.5 (3.0 - 0.2 - 0.3). The second agent will receive a payoff of 2.6. If in a new scenario, the second agent probabilistically drew a randomly generated number above 0.95, the first agent would receive a sucker (CD) payoff, less the tax; the second agent would receive the DC or temptation payoff, less the penalty (0.5 by default) and the tax.

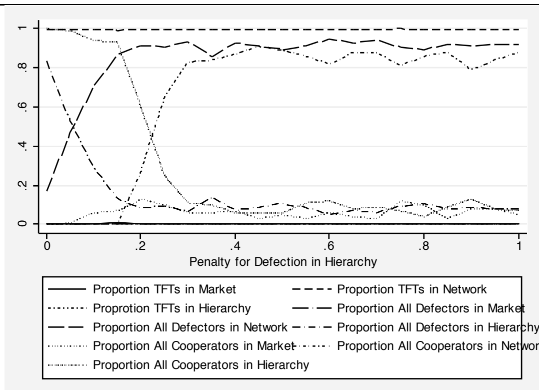
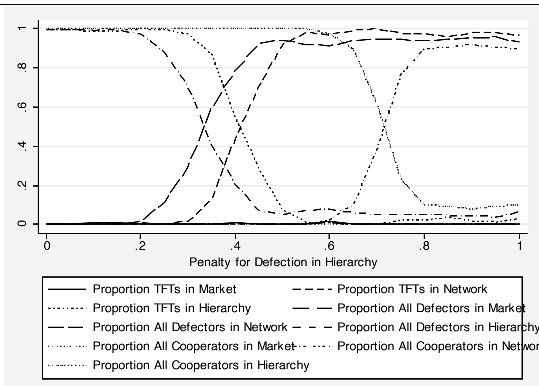
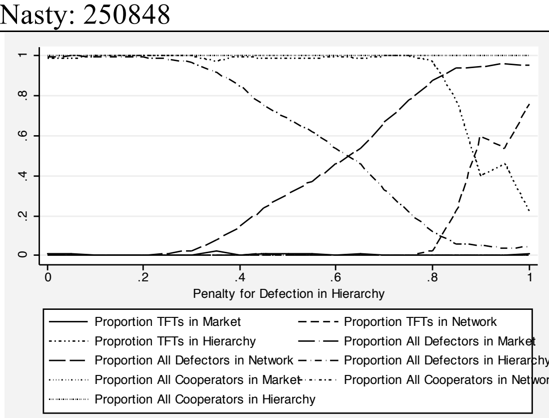
Following play, real payoffs are calculated as a function of the outcome of play, adjusted for the players' ideal points (k) if the outcome was cooperative, punishments, and fees prescribed by their organizations. Actual payoffs can differ from expected payoffs, but are on average the same. Agents also update their beliefs. All results are recorded and, if appropriate, play continues for another round.

Parameter Scans

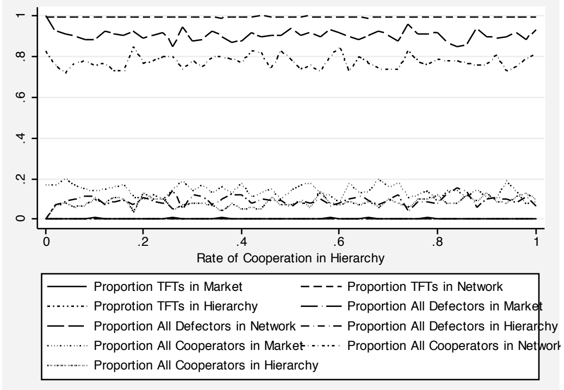
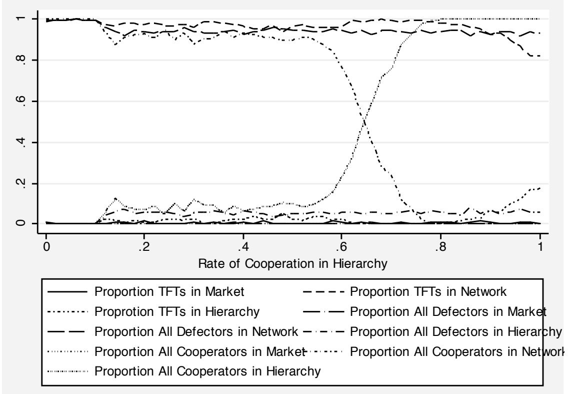
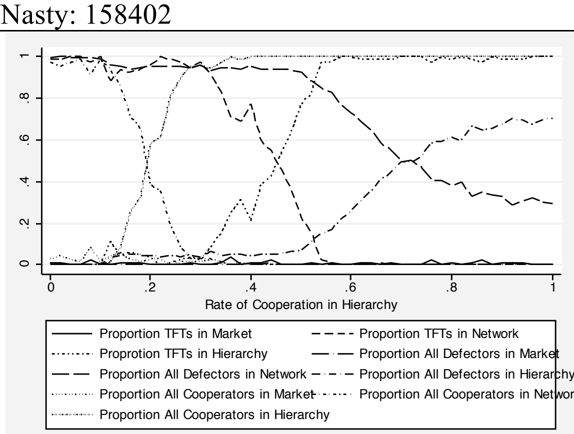
We now present parameter scans of the final model. In the following graphs, we vary each of the core parameters of interest across three populations: a nice population (10ALLCs/15ALLDs/75TFTs), a moderate population (20 ALLCs/50ALLDs/30TFTs), and a nasty population (7ALLCs/85ALLDs/8TFTs). Due to the large number of runs necessary to produce these scans, all the simulations show comparative statics at round 10 and the number of repetitions is set to 10.

HIERARCHY VARIABLES

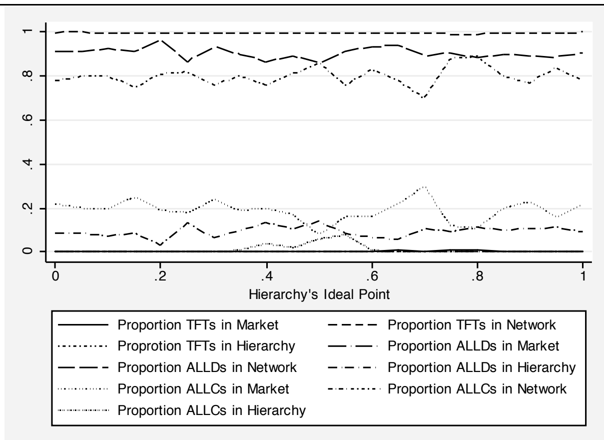
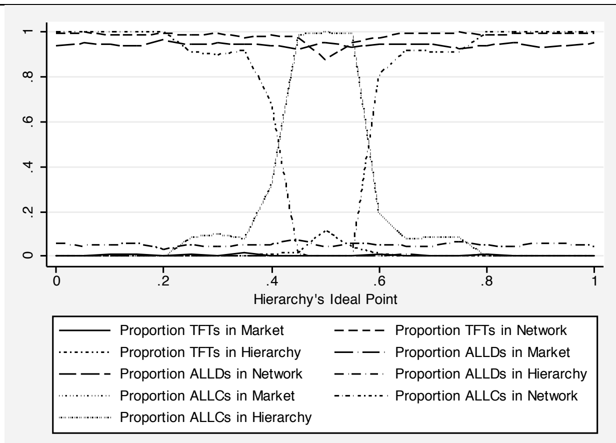
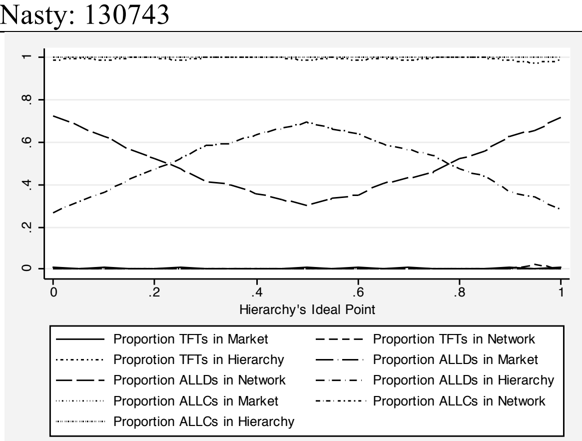
Penalty for Defection in the Hierarchy (*v)* (incremented from 0.0 by +0.05 over 21 increments)

<table>
<tr><td>Nice: 630360</td><td>Description:</td></tr>
<tr><td>
</td><td>TFTs in the network; ALLDs leave hierarchy for network as penalty increases; ALLCs leave hierarchy for market and network.</td></tr>
<tr><td>Moderate: 219460</td><td></td></tr>
<tr><td>
</td><td>All types go from hierarchy to network. ALLCs willing to pay largest penalties.</td></tr>
<tr><td>Nasty: 250848</td><td></td></tr>
<tr><td>
</td><td>ALLCs are in the hierarchy; ALLDs move from hierarchy to network; TFTs move from hierarchy to network as the penalty for defection increases.</td></tr>
</table>

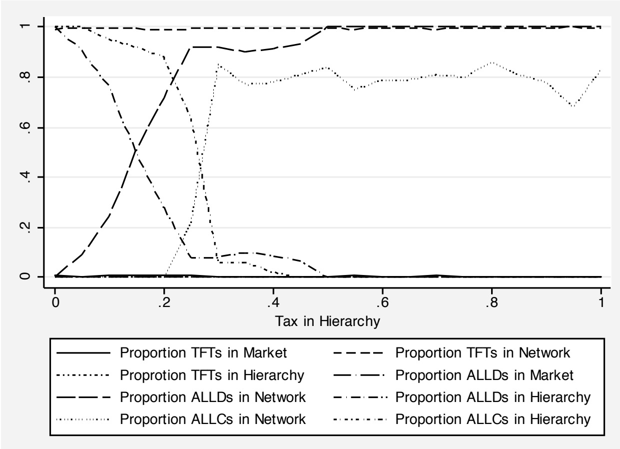
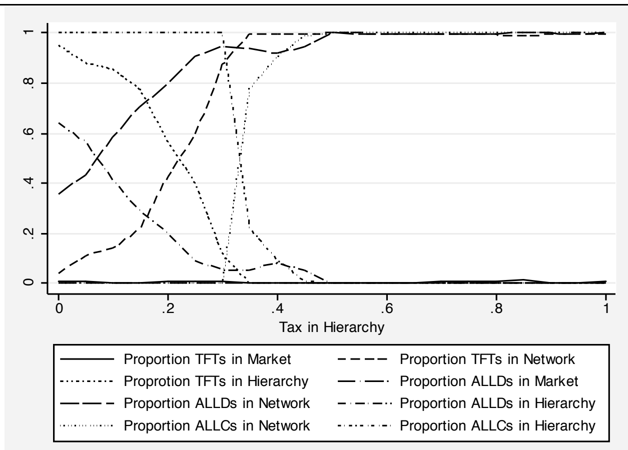
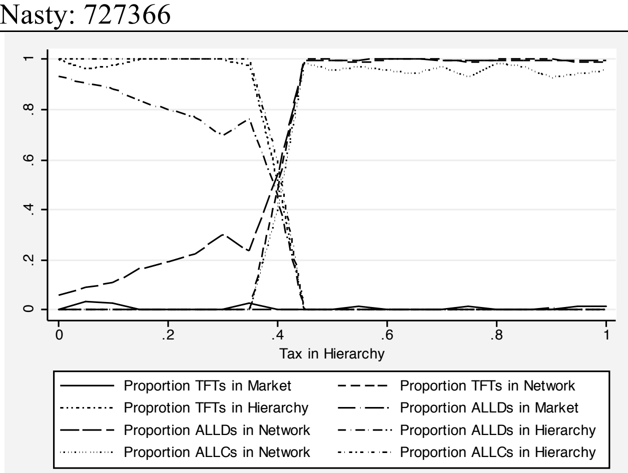
Hierarchy: probability of cooperation *(q)* (incremented from 0.0 by +0.2 over 51 increments)

<table>
<tr><td>Nice: 250848</td><td>Notes:</td></tr>
<tr><td>
</td><td>All types predominantly in network.</td></tr>
<tr><td>Moderate: 367337</td><td></td></tr>
<tr><td>
</td><td>TFTs and ALLDs in network predominantly; ALLCs move from network to hierarchy.</td></tr>
<tr><td>Nasty: 158402</td><td></td></tr>
<tr><td>
</td><td>TFTs: start in Market and network, move to hierarchy and network; ALLDs move slowly from network to hierarchy; ALLCs move quickly from market and network to hierarchy.</td></tr>
</table>

Hierarchy Ideal point (p_h) Incremented from 0.0, 21 times by +0.05.

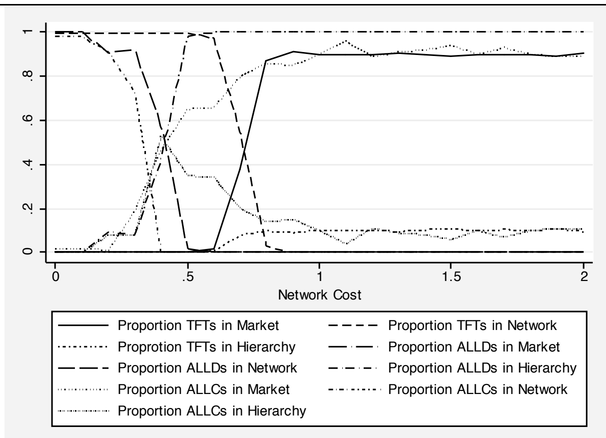
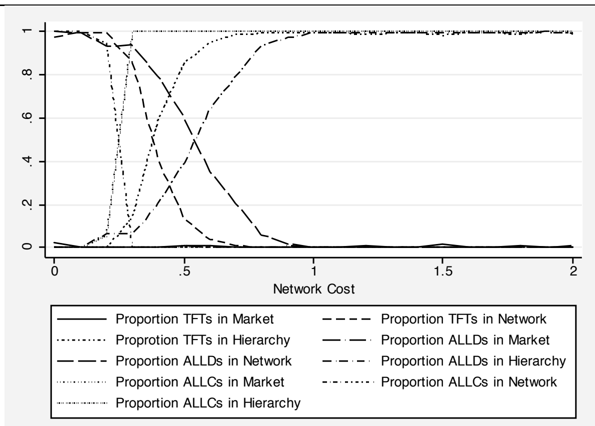
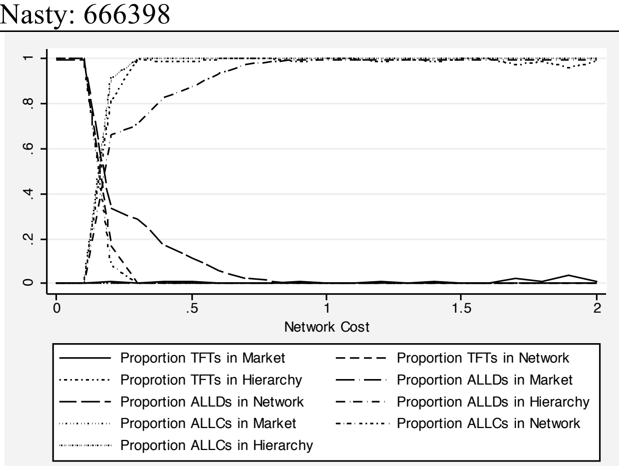
<table>
<tr><td>Nice: 477755</td><td></td></tr>
<tr><td>
</td><td>TFTs in the network,; ALLDs in network and hierarchy; ALLCs mostly in network, but also in market and hierarchy..</td></tr>
<tr><td>Moderate: 323251</td><td></td></tr>
<tr><td>
</td><td>TFTs primarily in network, except a few enter hierarchy toward center of distribution; ALLDs mostly in network, low levels of hierarchy membership; ALLCs in network at extreme values, enter hierarchy at moderate values.</td></tr>
<tr><td>Nasty: 130743</td><td></td></tr>
<tr><td>
</td><td>TFTs in hierarchy; ALLDs in network and hierarchy (peak hierarchy membership around 0.5); ALLCs in hierarchy.</td></tr>
</table>

Tax in Hierarchy (τ) (fee on GUI) Incremented 51 times from 0.0 by + 0.02.

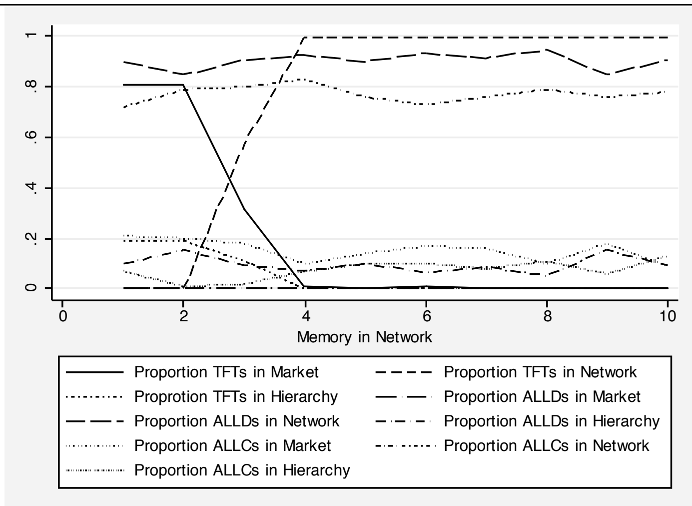
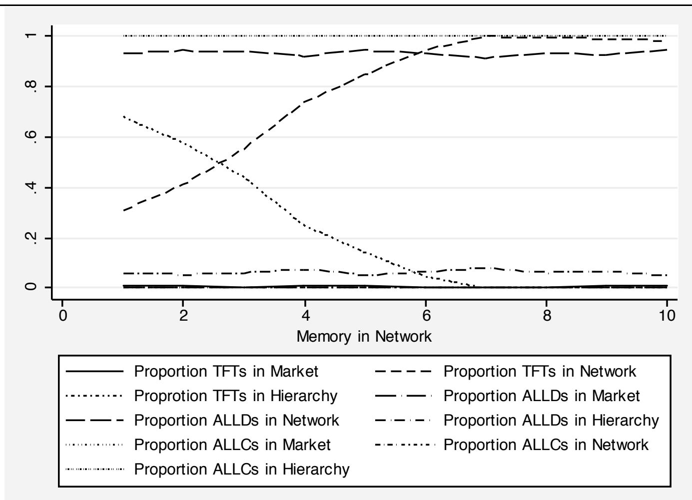
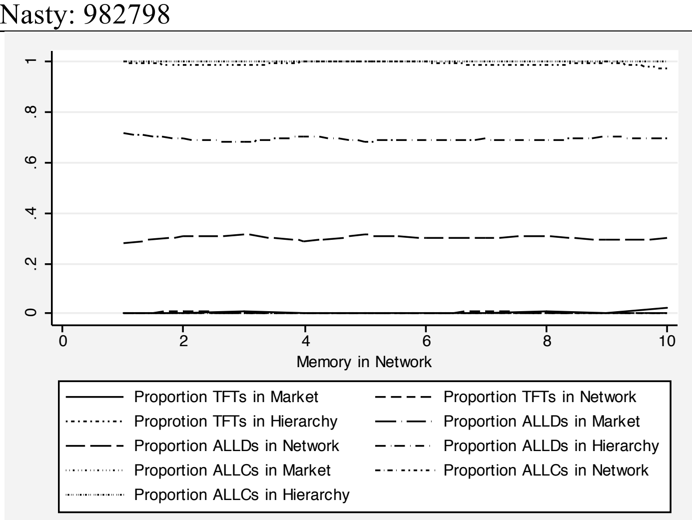
<table>
<tr><td>Nice: 856560</td><td></td></tr>
<tr><td>
</td><td>TFTs in Network; ALLDs leave hierarchy for network as tax increases; ALLCs leave hierarchy for network and market as tax increases.</td></tr>
<tr><td>Moderate: 919180</td><td></td></tr>
<tr><td>
</td><td>TFTs leave hierarchy for network as tax increases; ALLDs leave hierarchy for network earlier than TFTs; ALLCs leave hierarchy for network latest of all types.</td></tr>
<tr><td>Nasty: 727366</td><td></td></tr>
<tr><td>
</td><td>TFTs leave hierarchy for network as tax increases, low level market use throughout; ALLDs leave hierarchy for network; ALLCs leave hierarchy for network and market.</td></tr>
</table>

NETWORK VARIABLES

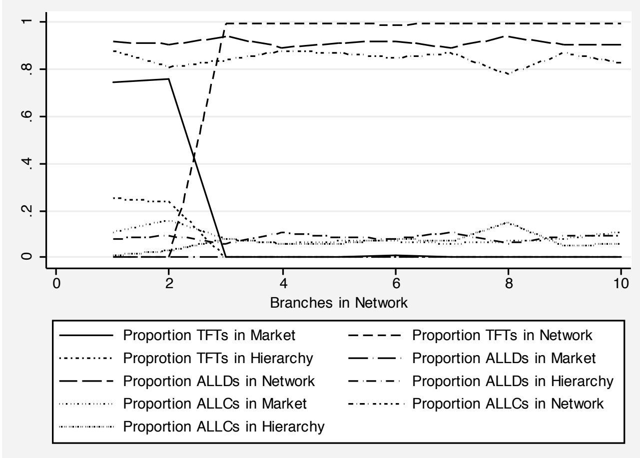
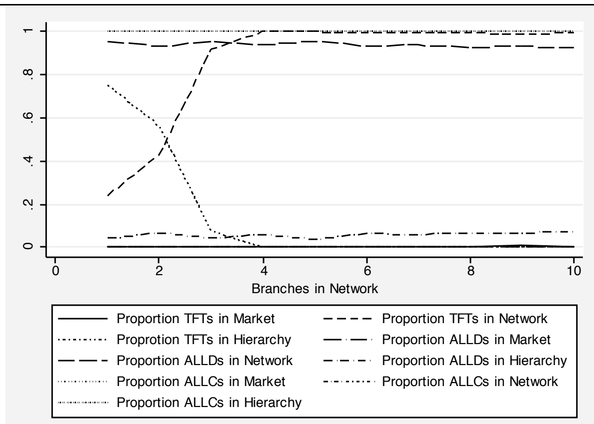
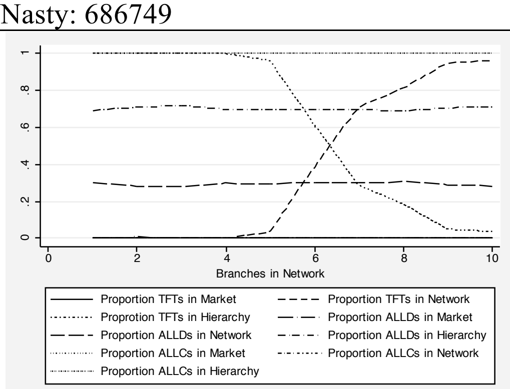
Network Cost (ϕ) Incremented 21 times from 0, by +0.1

Nice: 165714	
	TFT agents leave network for the market and hierarchy as cost increases; ALLDs leave the network for the hierarchy; ALLCs leave the network earliest for the market and some for the hierarchy.
Moderate: 333946	
	TFTs move from network to the hierarchy with low levels of agents in the market throughout; ALLDs move from network to hierarchy after TFTs; ALLCs move from network to hierarchy earliest.
Nasty: 666398	
	TFTs move from network to hierarchy and some in market; ALLDs move from network to hierarchy as net cost increase; ALLCs move earliest and fastest from network to hierarchy.

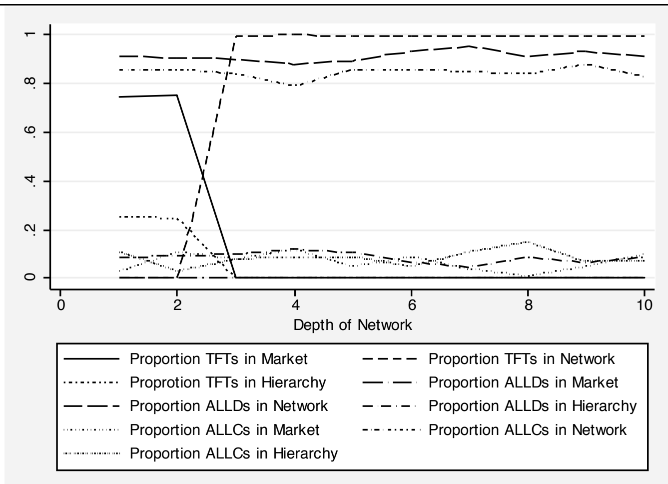
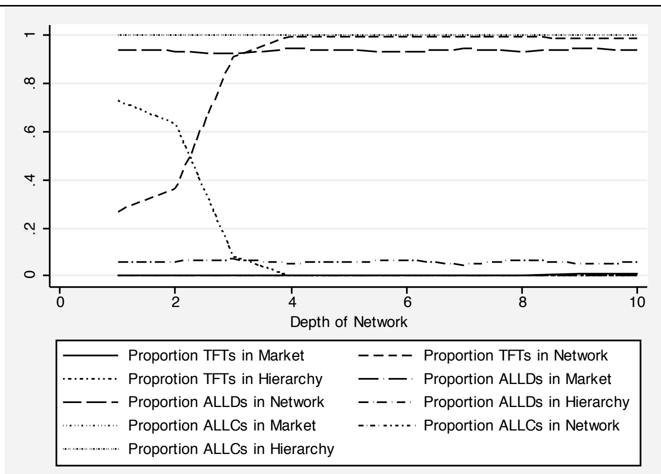
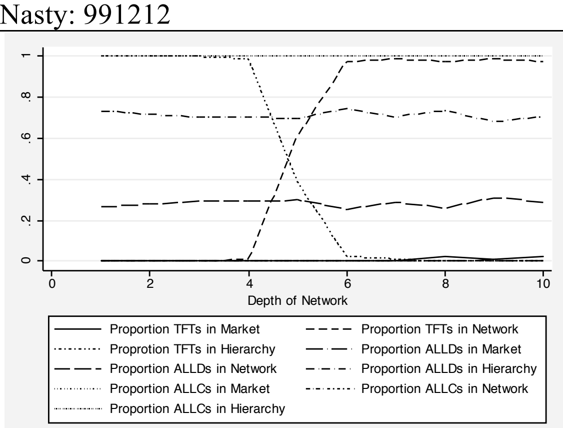
Network memory (m_n) Incremented 10 times from 1 by +1.0.

<table>
<tr><td>Nice: 475652</td><td></td></tr>
<tr><td>
</td><td>TFT agents move from the market and hierarchy into the network; ALLDs constantly in network and hierarchy; ALLCs largely in network, but also low level membership in the hierarchy and market, like the ALLDs these are stable.</td></tr>
<tr><td>Moderate: 302860</td><td></td></tr>
<tr><td>
</td><td>TFTs leave hierarchy for the network as memory increases; ALLDs in network with a few in the hierarchy; ALLCs in hierarchy .</td></tr>
<tr><td>Nasty: 982798</td><td></td></tr>
<tr><td>
</td><td>TFTs in hierarchy; ALLDs in hierarchy with some in network; ALLCS in hierarchy. Across all types memory in the network does not affect organizational choice significantly</td></tr>
</table>

Network Width (α) (Branches in GUI). Incremented 10 times, from 1 by +1.

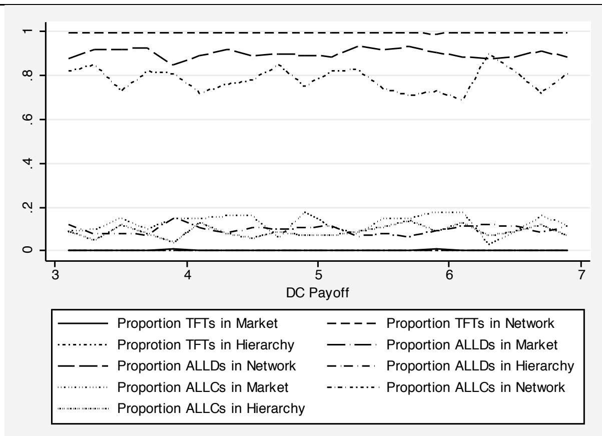
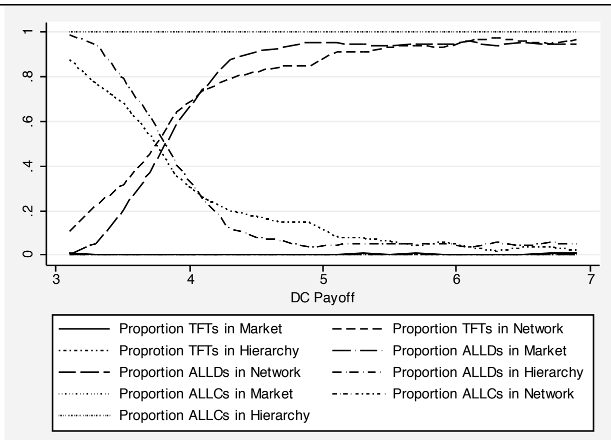
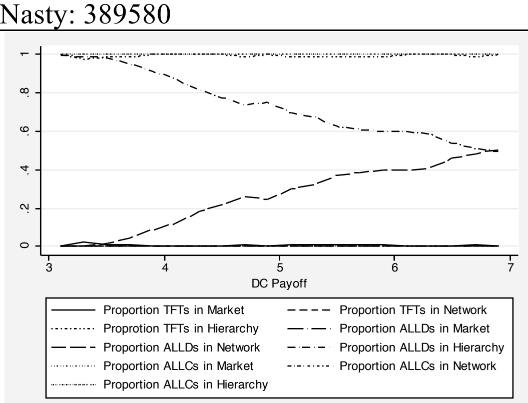
<table>
<tr><td>Nice: 101983</td><td></td></tr>
<tr><td>
</td><td>TFTs move from market and hierarchy into the network as branches increases above 3; ALLDs in network and low levels in the hierarchy; ALLCs in Network with low levels of market and hierarchy membership.</td></tr>
<tr><td>Moderate: 624314</td><td></td></tr>
<tr><td>
</td><td>TFT agents move from hierarchy to network as the branches increase, but more slowly than in the nice population; ALLDs predominantly in network with low levels of hierarchy membership throughout; ALLCS in hierarchy.</td></tr>
<tr><td>Nasty: 686749</td><td></td></tr>
<tr><td>
</td><td>TFTs move from hierarchy to network as branches increase; ALLDS are largely in hierarchy with some in the network; ALLCS are all in hierarchy.</td></tr>
</table>

Network Depth *(l)* (Height in GUI). Incremented 10 times starting from 1 by +1.

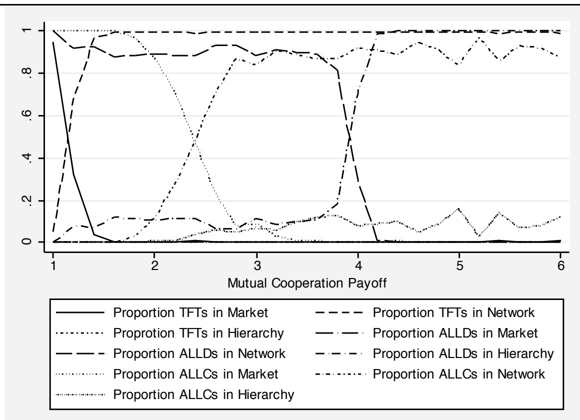
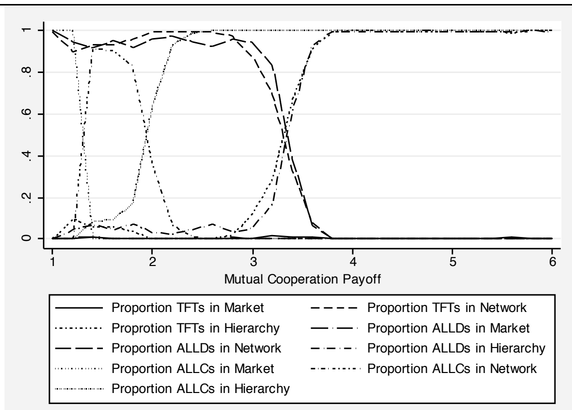
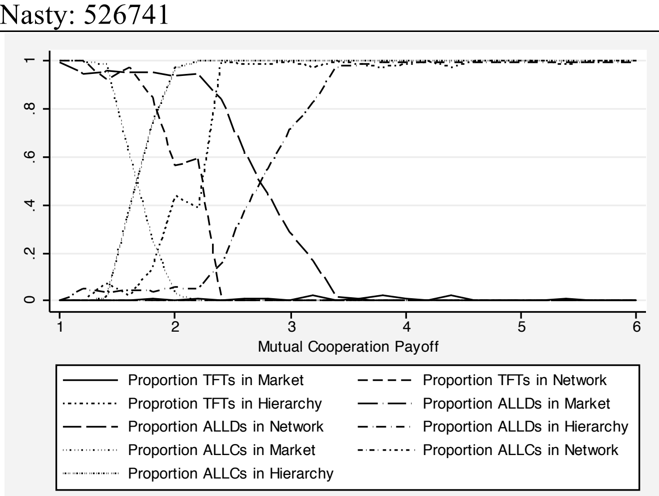
Nice: 64654	
	TFTs leave market and hierarchy for network as the depth increases; ALLDs in network with a few in the hierarchy; ALLCs in network predominantly, with a few in the market and hierarchy.
Moderate: 809536	
	TFTS leave hierarchy for network as height of network increases; ALLDS mostly in network with some in hierarchy; ALLCs in hierarchy.
Nasty: 991212	
	TFTs leave hierarchy for network and market; ALLDs in hierarchy with some in network; ALLCs in hierarchy.

PAYOFFS:

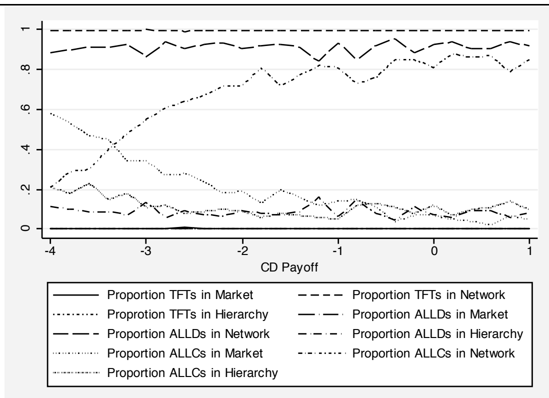
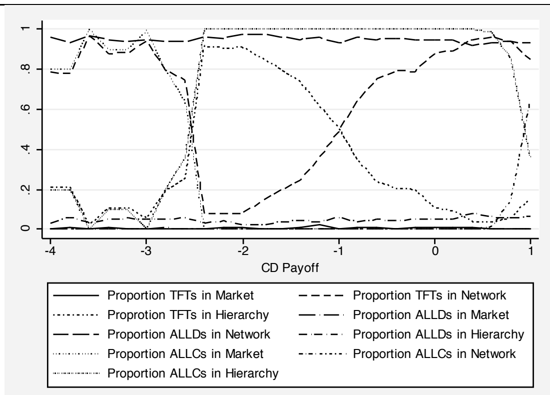
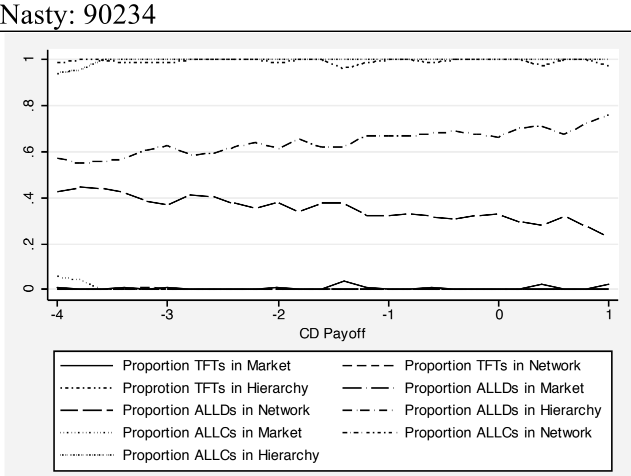
DC payoff (T) Incremented 50 times from 3.1 by +0.1.

Nice: 645478	
	TFTs in network; ALLDs in Network and Market; ALLCs largely in network, but also in market and hierarchy.
Moderate: 679885	
	TFTs and ALLDs leave hierarchy for network; ALLCs all in hierarchy.
Nasty: 389580	
	TFTs almost exclusively in hierarchy, very low levels of market; ALLDs leave hierarchy and enter network as DC payoff increases; ALLCs in hierarchy.

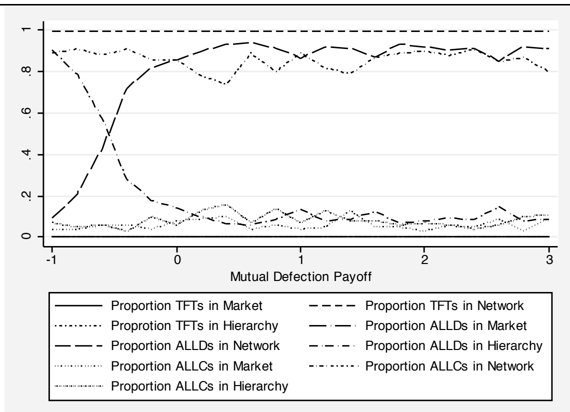
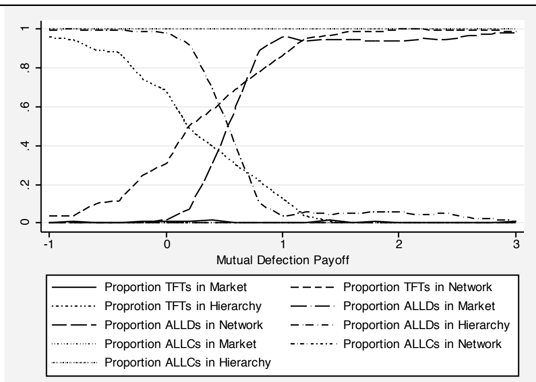
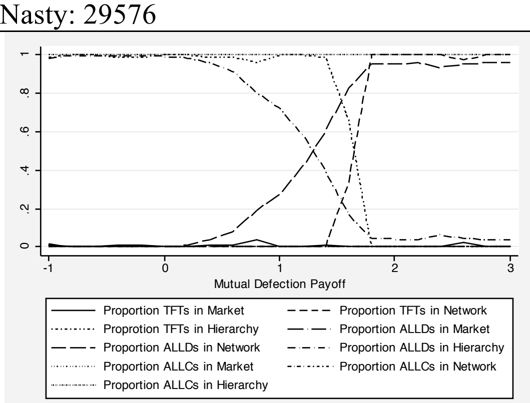
Mutual cooperation payoff (R). Incremented from 1.0 by increments of +0.2 over 26 increments.

Nice: 299007	
	TFTs leave market for network; ALLDs leave network for hierarchy above 4; ALLCs leave market for network and a few for the hierarchy.
Moderate: 71721	
	TFTs leave network for the hierarchy; ALLDs leave network for hierarchy; ALLCs move from market to network to hierarchy.
Nasty: 526741	
	TFTs move from network to hierarchy; ALLDs move from network to hierarchy; ALLCs move from market to hierarchy.

CD Payoff (S) Incremented 21 times from 1.0 by -0.2 .

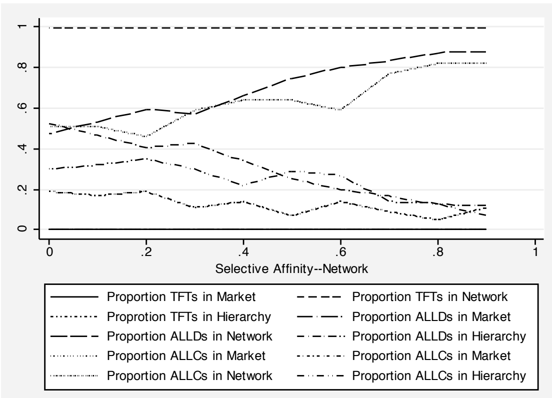
<table>
<tr><td>Nice: 658535</td><td></td></tr>
<tr><td>
</td><td>TFTs in network; ALLDs in network with a few in the hierarchy; ALLCs leave market and hierarchy for the network.</td></tr>
<tr><td>Moderate: 83359</td><td></td></tr>
<tr><td>
</td><td>TFTs leave network for hierarchy over middle values then re-enter network as CD payoffs become positive; ALLDs in network and some in hierarchy; ALLCs in market and leave for hierarchy; then enter network.</td></tr>
<tr><td>Nasty: 90234</td><td></td></tr>
<tr><td>
</td><td>TFTs.in hierarchy; ALLDs in hierarchy and network; ALLCs in hierarchy.</td></tr>
</table>

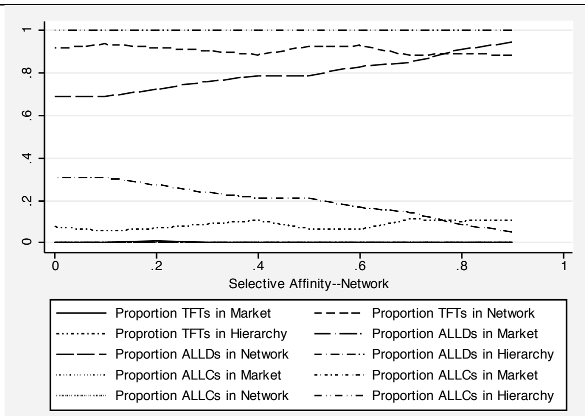
Mutual defection (P) Incremented 21 times from 3.0, by -0.2.

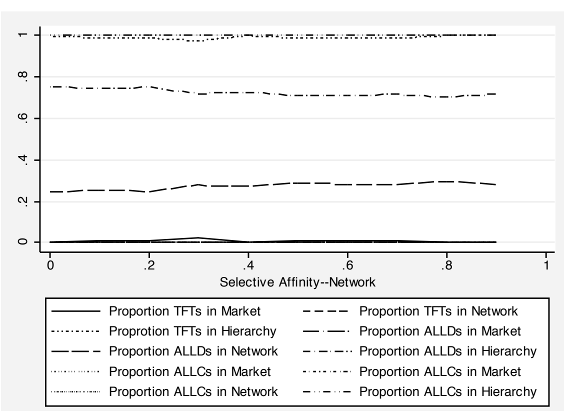
<table>
<tr><td>Nice: 177667</td><td></td></tr>
<tr><td>
</td><td>TFTs use network primarily; ALLDs go from hierarchy to network; ALLCs primarily in network, low levels of market and hierarchy.</td></tr>
<tr><td>Moderate: 720862</td><td></td></tr>
<tr><td>
</td><td>TFTs leave hierarchy for the network; ALLDs leave hierarchy for the network; ALLCs in hierarchy.</td></tr>
<tr><td>Nasty: 29576</td><td></td></tr>
<tr><td>
</td><td>TFTs leave hierarchy for network; ALLDs leave hierarchy for network before TFTs; ALLCs in hierarchy.</td></tr>
</table>

SELECTIVE AFFINITY AND IDEAL POINT

Selective Affinity (η) incremented 10 times from 0 by increments of 0.1.

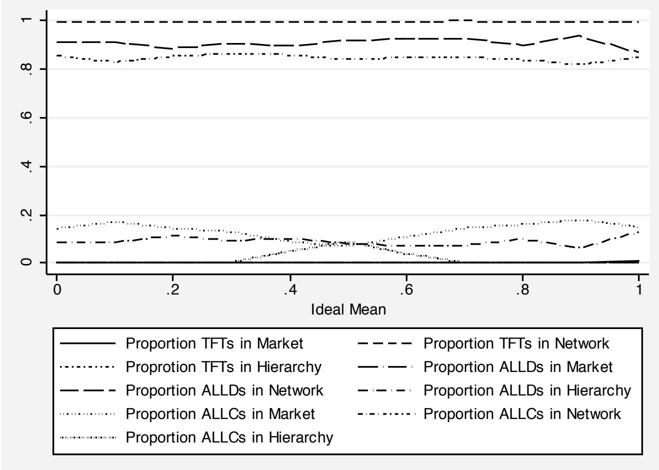
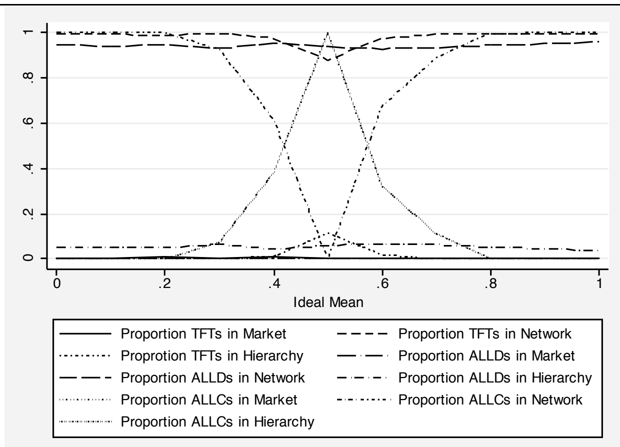
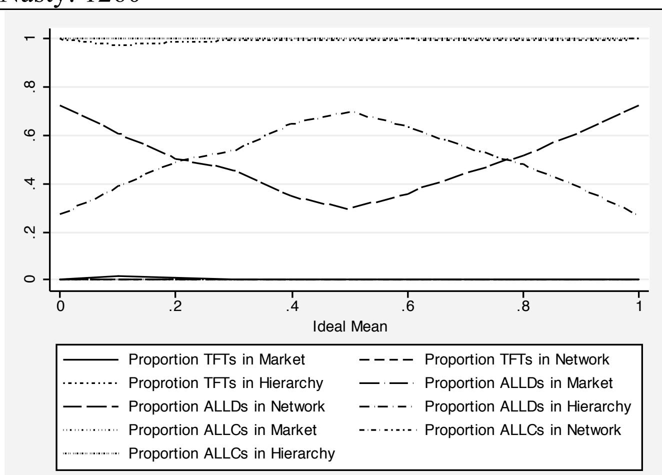
<table>
<tr><td>Nice 535767</td><td></td></tr>
<tr><td>


</td><td>TFTs in network; ALLDs leave hierarchy for network; ALLCs leave hierarchy and market for the network.</td></tr>
<tr><td>Moderate: 137593</td><td></td></tr>
<tr><td>


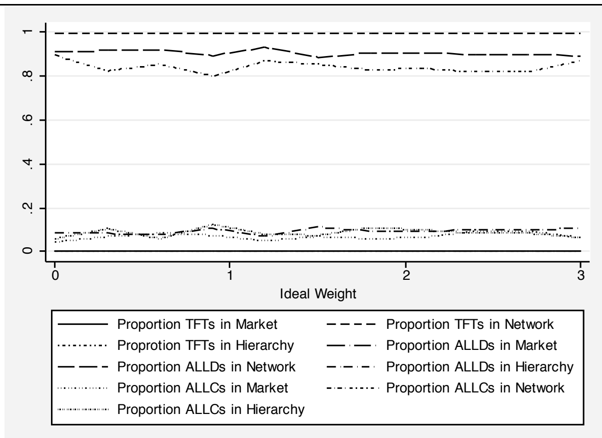
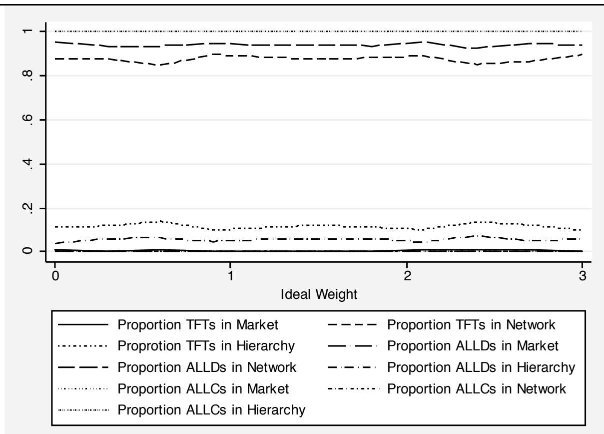
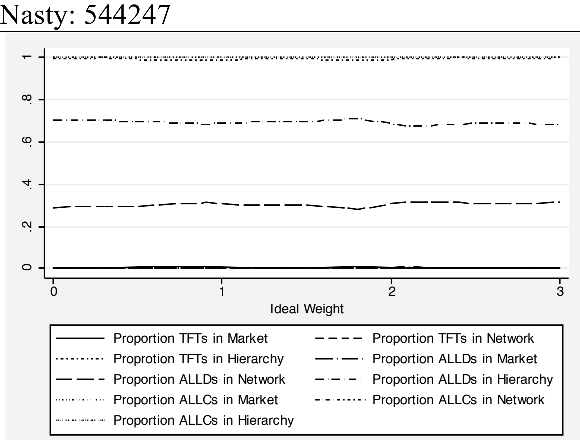
</td><td>TFTs in network predominantly; ALLDs leave hierarchy for network, ALLCs in hierarchy.</td></tr>
<tr><td>Nasty: 147005</td><td></td></tr>
<tr><td>


</td><td>TFTs in hierarchy; ALLDs in hierarchy and network;
ALLCs in hierarchy</td></tr>
</table>

Population Ideal Mean Incremented 11 times from 0 by +0.1

<table>
<tr><td>Nice: 974363</td><td></td></tr>
<tr><td>
</td><td>TFTs in Network; ALLDs in network; ALLCs in Network, with a few in market and hierarchy.</td></tr>
<tr><td>Moderate: 756209</td><td></td></tr>
<tr><td>
</td><td>TFTs in network, at center of distribution will enter the hierarchy; ALLDs primarily in network; ALLCs in network until the center; they leave for the hierarachy when its ideal point is close to their own.</td></tr>
<tr><td>Nasty: 1260</td><td></td></tr>
<tr><td>
</td><td>TFTs in hierarchy; ALLDs leave network for hierarchy in the middle and reverse; ALLCs in hierarchy.</td></tr>
</table>

Weight on Ideal *(w)* Incremented 11 times from 0.0 by +0.3

<table>
<tr><td>Nice: 177219</td><td></td></tr>
<tr><td>
</td><td>TFTs and ALLDs in network; ALLCs in network with a few in market and hierarchy.</td></tr>
<tr><td>Moderate: 90378</td><td></td></tr>
<tr><td>
</td><td>TFTs in network; ALLDs in network; ALLCS in hierarchy.</td></tr>
<tr><td>Nasty: 544247</td><td></td></tr>
<tr><td>
</td><td>TFTs in hierarchy; ALLDs in hierarchy and network; ALLCs in hierarchy.</td></tr>
</table>