Details of the Regional Security Game

The model is implemented in NetLogo 6.0.1. A replication of the model in Python 3.6.5 is available from the author upon request. Agents operate in discrete intervals of time with an infinite horizon and intervals [ticks] noted as $t = 1,2,3,\ldots$. This is a game of imperfect but complete information: agents know everyone's power and can observe their previous moves, but do not know others' moves for the current round in advance (because moves occur simultaneously) and cannot directly observe their strategies. For the purposes of the experiment, the model stops iterating at $t = 500$, which allows most (but not all) populations to reach a steady state[1].

The first stage of the game is structured as an *n-player* public goods game. Let N_t^C be the number of contributors in period t and N_t^D be the number of defectors. The amount that each state contributes to the common pool each round $q_{i,t} = a \times p_i$, where a is a fixed contribution ratio following Stone *et al* (2008). This paper will set $a = 1$, but the model permits a to be varied or evolve endogenously. The smaller a, the less of its capabilities a state contributes each round. The total amount of regional security provided by contributing states on tick t is $Q_t = \sum q_{i,t}$.

Following the standard practice in public goods games, the *regional security game* also multiplies the total security benefits by a synergy factor r, creating the possibility of a greater net positive yield for players that contribute. Each agent's share is the benefit is proportional to both the synergy factor and its proportion of the total system capabilities. Therefore, after the first stage of the game, the payouts are

[1] For the sake of argument, we consider one 'tick' to be the equivalent of one week in foreign policy time, so that 500 ticks are approximately ten years in the life of the regional security system. However, this choice is arbitrary in relation to the other variables in the model. For example, if states learn faster, fewer ticks pass before most populations reach a steady state. The number of ticks and the rate of learning have thus been selected by the author to allow some granularity in the observations.

$$\Delta b_i^D = rQ_t \times \frac{p_i}{P}$$

$$\Delta b_i^C = rQ_t \times \frac{p_i}{P} - (a \times p_i).$$

An instrumentally rational player would choose to defect rather than contribute only when $p_i < \frac{P}{r}$; in other words, without a synergy factor, states always have an incentive to free-ride and defect. In the *all-defect* equilibrium, the distribution of lifetime benefits remains unchanged after each round ($Q_t = 0$). Similarly, if the population is in the *all-contribute* state ($Q_t = P$), then there are also no relative gains and agents' relative scores do not change.

Let N_t^M be the number of 'moralists' in a given round who both cooperate and punish and N_t^I the number of 'immoralists' who defect while also punishing other defectors. Most studies are interested in the number of 'moralists', and disregard 'immoral' strategies where defectors punish other defectors. However, the *regional security game* tracks both types.

In order to punish, a state i pays a cost proportional to the power of the target $\beta_i = p_{-i} \times c$ where c is a fixed 'punishment ratio': i.e. some fraction $0 \leq c < 1$ by which the punisher seeks to reduce the defector's payoff. The smaller c, the fewer capabilities a state is required to set aside for punishing defectors, but the smaller the deterrent effect on potential defectors. For a punishing state, it's cheaper to impose costs on a weak state than a more capable one. At the end of the punishment stage, a punishing state pays a final cost $c \times \sum_n^{N^D} p_n$ such that the total cost of punishment paid is $C_t = c \times \sum_n^{N^D} p_n \times (N_t^M + N_t^I)$.

At the end of the punishment stage, each defecting state receives a total punishment $B_{i,t} = p_i \times c \times (N_t^M + N_t^I)$, summing for all states that chose to punish that round and multiplying by a fraction of its own power defined by the punishment ratio. It's clearly more costly to defect when the frequency of punishing states is high. Obviously, $\sum_n^{N^D} B_{i,t} = C_t$ such that the total punishment paid each round is identical to the total punishments imposed.

Therefore, for every individual player, the payoffs of each of the four move pairs after both stages are defined by:

$$\Delta b_i^{C,DP} = rQ_t \times \frac{p_i}{P} - a \times p_i.$$

$$\Delta b_i^{C,P} = rQ_t \times \frac{p_i}{P} - a \times p_i - c \times \sum_n^{N^D} p_n$$

$$\Delta b_i^{D,DP} = rQ_t \times \frac{p_i}{P} - p_i \times c \times (N_t^M + N_t^I)$$

$$\Delta b_i^{D,P} = rQ_t \times \frac{p_i}{P} - p_i \times c \times (N_t^M + N_t^I) - c \times \sum_n^{N^D} p_n$$

At the end of every round, every agent selects another with the largest lifetime payoff relative to its power p_i (its score) within a radius defined by its tolerance (the $LocalWinner$ and $LocalWinScore$). The model records whether the winning strategy contributed and punished defectors ($S_w = [LocalWinDidCoop, LocalWinDidPunish]$), which is behaviour visible to the other players. Then, every state incrementally updates their strategy towards the local winning behaviour (S_w), depending on the magnitude of the difference between their own strategy and the observed behaviour. Each state playing a strategy $s_{i,t}$ replaces its strategy for the next round with strategy $s_{i,t+1} = s_i + random \times (S_w - s_i)$ where the random number is generated from a normal distribution with $\bar{x} = 0.05$ & $\sigma = 0.025$. That is, a state does not change its strategy all at once, but optimises incrementally towards the strategy with the highest payoff.