Market for Protection ODD

Steve Doubleday

University of California, Irvine

August 19, 2013

Overview

Purpose

Konrad and Skaperdas (2012) develop an analytical model of the provision of security as a public good. The analysis of this "market for protection" begins with a population of "bandits" preying on "peasants" interacting in "anarchy". An agent-based simulation (MFPsim) has been developed to better understand the analytical model (MFP). In particular, if the analytical model is re-implemented as an agent-based simulation, do the conclusions of the analytical model still hold? For example, what is the effect of changing from continuous populations to discrete populations?

A peasant may spend some portion x of her unit effort on securing her output against bandits, spending the remainder of her effort (1 - x) on productive work. A protection function, $p(x)$, models this security effort; it converts security effort into effective protection of some proportion of a peasant's output, leaving the remainder to be surrendered to a bandit. Given continuous populations and a continuous, non-decreasing protection function, the MFP analysis then identifies an optimal proportion of the peasant's overall effort that should be dedicated to security, and the conditions under which the population of bandits and peasants will reach an equilibrium, where all actors have the same average payoff.

MFPsim modifies the MFP model as follows:

- Populations of bandits and peasants are discrete rather than continuous.
- The protection function is given the specific functional form of a contest success function.
- Peasants are randomly assigned a protection proportion.

In MFPsim, equilibrium is defined as a state where the average payoffs to bandits and to peasants are equal, within a tolerance, and maintained for a configurable number of consecutive periods. When the simulation reaches an end-state, the distribution of protection proportions in the peasant population can be compared to the optimal protection proportion predicted by the MFP model.

A technical purpose of the simulation is to enable future extensions to the simulation in a straight-forward manner to accommodate the remainder of the analytical model, where the original condition of anarchy is modified to introduce various forms of collective organization.

State Variables and Scales

The behavior of individual Peasants varies according to their protection proportions:

- *protectionProportion*: the amount of the peasant's unit effort devoted to defending its output, $\in [0,1]$. Once set, the protection proportion stays constant for the lifetime of the peasant. Note: although the domain is continuous, during creation of the peasant population, the Protection Model allocates peasant proportions in bins, at particular values within the domain, to simplify the subsequent analysis.

All other state in MFPsim consists of a set of global parameters, whose values are set by the Protection Model. These are discussed in the Submodels section.

Process Overview and scheduling

- When a peasant and a bandit interact, the peasant's payoff is:

$$U_p = p(x)(1 - x) \tag{1}$$

  The payoff to a bandit is:

$$U_b = [1 - p(x)](1 - x) \tag{2}$$

  That means that the peasant keeps a proportion $p(x)$ of the productive output 1 - x, and the remainder of that productive output is surrendered to the bandit. If a peasant does not have an interaction with a peasant in a given period, the peasant retains all output 1 - x; if a bandit does not have an interaction in a given period, the bandit's payoff is 0.

- The protection function $p(x)$ is given the functional form of a contest success function:

$$p(x) = \frac{\gamma x}{\gamma x + (1 - \gamma)} \tag{3}$$

  The parameter γ (CONTEST_FUNCTION_GAMMA) is interpreted as the "defensive ability" of the peasant.

- In each period, the population of bandits and peasants are randomly matched 1:1 for interaction. Payoffs for each agent are calculated, and one or more Dynamics are invoked, resulting in a new population of bandits and peasants. A new period commences, with payoffs starting at 0; payoffs do not accumulate. This process continues until one of the following end states occurs:

- Peasants go extinct: peasant population drops to 0.
- Bandits go extinct: bandit population drops to 0.
- Peasants go to maximum: peasant population exceeds MAXIMUM_POPULATION_SIZE
- Bandits go to maximum: bandit population exceeds MAXIMUM_POPULATION_SIZE
- Equilibrium: peasants and bandits have the same average payoff within a tolerance PAYOFF_DISCREPANCY_TOLERANCE, for consecutive periods EQUILIBRIUM_NUMBER_PERIODS_WITHOUT_ADJUSTMENT.
- Run limit exceeded: the number of periods without reaching one of the above end states reaches RUN_LIMIT.

When an end state is reached, the scenario ends.

- There are two dynamics that may be applied to a population, resulting in a new population for the next period.[1]
  - Die/Survive/Thrive (DST) Dynamic: this dynamic may result in the total population of agents changing in size, and is defined by two thresholds. Agents whose payoff is not equal to or greater than the "survive" threshold (SURVIVE_THRESHOLD) die without descendants. An agent whose payoff is equal to or exceeds the survive threshold has one descendant, unless the payoff is equal to or exceeds the "thrive" threshold (THRIVE_THRESHOLD), when the agent

[1] Current implementation: DST dynamic is always invoked; Role-shifting dynamic may then optionally be invoked. The order is not reversible.

has two descendants. A descendant inherits the strategy of its parent; specifically, a peasant in the new population inherits the protection proportion of its parent.[2]

- Role-Shifting Dynamic: this dynamic results in a percentage of the lowest-performing role shifting to the better-performing role. The total population of agents does not change. Role performance is defined by the average payoff for all members of a role in each period. If the difference between the average payoffs is greater than PAYOFF_DISCREPANCY_TOLERANCE, then a percentage ADJUSTMENT_FACTOR_PERCENTAGE of the lower-performing agents shift to the other role. When bandits shift to the peasant role, they are given a protection proportion determined by flag NEW_PEASANT_GETS_BEST_PROTECTION_PROPORTION. If true, they get the current period's best performing protection proportion; otherwise, they get a randomly allocated protection proportion. Fractional adjustments less than 1 are rounded to 1, but fractions for numbers above 1 are rounded down to the next integer.

Design Concepts

Emergence

The equilibria of the bandit / peasant population, and the most prevalent protection proportions, emerge from the individual interactions of bandits and peasants.

[2] Current implementation: The DST dynamic is always invoked, but may be effectively disabled by setting the survive threshold to 0 and the thrive threshold to 1. There is a known defect with this implementation, where peasants with $x = 0$ will thrive if not preyed upon.

Adaptation

Agents do not adapt their behavior, nor do their strategies mutate. Once assigned, a peasant and its descendants keep the same protection proportion. Bandits do not have individual strategies.

Fitness/Objectives

Agents attempt to maximize their payoffs (utilities) in their interactions with other agents. But this behavior is determined by the combination of assigned protection proportion, and the matching pattern for interaction; there is no latitude for individual decision at the moment of interaction.

Prediction

Agents do not predict the consequences of their actions.

Sensing

Agents do not sense their environment. They are not placed on a spacial grid; their interaction patterns are determined randomly and externally.

Interactions

There are two interaction patterns, determined by flag NORMAL_INTERACTION_PATTERN. If true, bandits are always paired with peasants. If false, any agent may be paired with any other agent. The latter setting is used to create a single population for the purposes of evolutionary game theory analysis.

The populations are matched randomly in each period, with one agent interacting with one other agent, until one population is exhausted. The remaining agents have no interaction in that period, and receive a payoff as defined in the Process Overview.

Stochasticity

A single long integer is used as a seed for a pseudo-random number generator. This is used first to allocate protection proportions randomly when the population of peasants is first built at the beginning of the scenario. In each period, the random number generator is then used to shuffle one population, whose members are then selected one at a time to interact with the next member of the other population. When bandits move to the peasant role under the Role-Shifting Dynamic, and if NEW_PEASANT_GETS_BEST_PROTECTION_PROPORTION is false, the random number generator is used to assign a protection proportion to the new peasant.

Collectives

There is no collective or social organization to either the bandit or peasant populations; agents interact randomly, and are not affected by the interactions of others.

Observation

Summary statistics are gathered for every scenario, as one record in a file formatted as comma-separated values. Each record contains the values of each parameter, the parameter point, which defines a single scenario. The file thus records the results of traversing the parameter space, as a collection of scenarios, termed a "scenario set". In addition to the parameter values, each record includes these summary statistics:

- Scenario number: unique integer for each scenario in the scenario set
- Stop reason code: integer code defining for which of the six reasons the scenario stopped execution.

- Period: period in which the scenario stopped.
- Numbers of bandits and peasants, before and after replication in the final period of the scenario.
- Average bandit and peasant payoffs
- Discrepancy between average bandit and peasant payoffs
- Adjustment, if any, between the bandit and peasant roles. Positive integers: bandits to peasants; negative integers: peasants to bandits; zero: no change.
- Average, median and mode of the peasant protection proportion
- Average, median and mode of the number of peasants a bandit preys upon[3]
- Number and percentage of peasants with the protection proportion defined by each bin (total bins defined by PROTECTION_PROPORTION_NUMBER_INTERVALS.

A log file is created for each scenario, containing the same statistics as defined in the summary file, for each period of the scenario.

Details

Initialization

Initialization for each scenario includes the following steps:

- Peasant population built: peasants numbering NUMBER_PEASANTS are created, with protection proportions randomly allocated at the boundaries of the bins numbering PROTECTION_PROPORTION_ NUMBER_INTERVALS, each bin spanning

[3]Current implementation includes support for multiple bandits preying on multiple peasants; default is one bandit preys upon one peasant.

an interval of PROTECTION_PROPORTION_ INTERVAL_SIZE. Each peasant is given a contest function; all contest functions for a given scenario share the same γ, CONTEST_FUNCTION_GAMMA.

- Bandit population built: bandits numbering NUMBER_BANDITS are created.
- An Interaction Pattern is initialized with the bandit and peasant populations, with mode determined by flag NORMAL_INTERACTION_PATTERN (see Interactions).
- Statistics are initialized to zeros to collect observation data.
- Dynamics are added to the scenario. A DST dynamic is always added, with thresholds set per SURVIVE_THRESHOLD and THRIVE_THRESHOLD. A Role-Shifting dynamic may be added, if ROLE_SHIFTING is true.
- A pseudo-random number generator is initialized with the value of RANDOM_SEED
- An Equilibrium Seeker is initialized with the above objects. The run limit is set from RUN_LIMIT. The definition of equilibrium is set from EQUILIBRIUM_NUMBER_PERIODS_ WITHOUT_ADJUSTMENT and PAYOFF_DISCREPANCY_TOLERANCE (see Process Overview). The period is set to 1.

The Protection Model invokes the Equilibrium Seeker to begin execution.

Input

There are no input data; all execution is controlled by parameters (see Submodels)

Submodels

The routines of the Protection Model are controlled by the following parameters; details of their logic are outlined in the prior sections.[4]

- PROTECTION_PROPORTION_NUMBER_INTERVALS: positive integer number of bins into which Peasant protection proportions can be allocated
- PROTECTION_PROPORTION_INTERVAL_SIZE: size of each bin for Peasant protection proportions, $\in$ [0,1]. Interval_size * number_intervals should equal 1.0.
- CONTEST_FUNCTION_GAMMA: input to the Peasant contest success function defining the defensive ability of the Peasant, $\in$ [0.5,1]. Shared for all peasants in the population for a given scenario.
- ROLE_SHIFTING: determines whether the Role-Shifting dynamic will be invoked, $\in$ {true, false}
- SURVIVE_THRESHOLD: determines the minimum payoff an agent must achieve in a period to have a single descendant in the next period, $\in$ [0,1]
- THRIVE_THRESHOLD: determines the minimum payoff an agent must achieve in a period to have two descendants in the next period, $\in$ [0,1]; greater than or equal to SURVIVE_THRESHOLD
- PAYOFF_DISCREPANCY_TOLERANCE: defines the maximum difference between the average payoffs for bandits and peasants for the two populations to be considered to be in equilibrium in the current period, $\in$ (0,1)

[4] Current implementation: there are other parameters in the v1.1 implementation that function, but have not been described in the studies that use the v1.1 code. Left at their defaults, they will not affect the function of the code. They implement: a transaction cost to bandits for preying on peasants; enable multiple bandits to prey upon multiple peasants; enable a matching function to determine the probability that a bandit is successful in preying on a target peasant.

- ADJUSTMENT_FACTOR_PERCENTAGE: percentage of the lower-performing role that will be shifted to the better-performing role at the conclusion of this period, if ROLE_SHIFTING is true, $\in$ (0,1)

- EQUILIBRIUM_NUMBER_PERIODS_WITHOUT_ADJUSTMENT: positive integer number of consecutive periods without adjustment that must elapse for the scenario to be considered to be in equilibrium.

- NUMBER_PEASANTS: positive integer number of peasants that will be created during initialization

- NUMBER_BANDITS: positive integer number of bandits that will be created during initialization

- RUN_LIMIT: positive integer number of periods that the scenario will execute without reaching equilibrium, before the scenario will stop.

- NORMAL_INTERACTION_PATTERN: determines whether bandits only interact with peasants, if true, or whether any agent can interact with any other agent, $\in$ {true, false}

- MAXIMUM_POPULATION_SIZE: positive integer setting the maximum size of the population of either bandits or peasants after replication; when exceeded, the scenario will stop after that period.

- NEW_PEASANT_GETS_BEST_PROTECTION_PROPORTION: positive integer setting the maximum size of the population of either bandits or peasants after replication; when exceeded, the scenario will stop after that period.

- FORCE_PEASANT_ALLOCATION_TO_HIGH_LOW: determines whether peasants will be initialized with one of two protection proportions, if true; $\in$ {true, false}. For use in evolutionary game theory analyses. Defaults to false.

- FORCE_PEASANT_ALLOCATION_LOW_INITIAL_PEASANTS: non-negative integer number of peasants less than or equal to NUMBER_PEASANTS that will be initialized with protection proportion FORCE_PEASANT_ALLOCATION_LOW_PROPORTION, if FORCE_PEASANT_ALLOCATION_TO_HIGH_LOW is true.

- FORCE_PEASANT_ALLOCATION_LOW_PROPORTION: if FORCE_PEASANT_ALLOCATION_TO_HIGH_LOW is true, sets the protection proportion $\in$ [0,1] of a number of peasants equaling FORCE_PEASANT_ALLOCATION_LOW_INITIAL_PEASANTS.

- FORCE_PEASANT_ALLOCATION_HIGH_PROPORTION: if FORCE_PEASANT_ALLOCATION_TO_HIGH_LOW is true, sets the protection proportion $\in$ [0,1] of a number of peasants equaling NUMBER_PEASANTS - FORCE_PEASANT_ALLOCATION_LOW_INITIAL_PEASANTS.

- RANDOM_SEED: long integer used as the seed for a pseudo-random number generator.

References

Kai A Konrad and Stergios Skaperdas. The Market for Protection and the Origin of the State. *Economic Theory*, 50(2):417–443, 2012.