ODD protocol for the agent-based model

"Grade languages in peer review".

This model description follows the ODD (Overview, Design concepts, Details) protocol for describing individual- and agent-based models (Grimm et al., 2006, 2010).

1. Purpose

This ABM re-implements and extends the simulation model of peer review described in Squazzoni & Gandelli (Squazzoni & Gandelli, 2013a, 2012) (hereafter: 'SG'). The SG model was originally developed for NetLogo and is available in CoMSES (Squazzoni & Gandelli, 2013b).
The purpose of the original SG model was to explore how different author and reviewer strategies would impact the outcome of a journal peer review system on an array of dimensions including peer review efficacy, efficiency and equality. In SG, reviewer evaluation consists of a continuous variable in the range [0,1], and this evaluation scale is the same for all reviewers. Our present extension to the SG model allows to explore the consequences of two more realistic assumptions on reviewer evaluation: (1) that the evaluation scale is discrete (e.g. like in a Likert-scale); (2) that there may be differences among their interpretation of the grades of the evaluation scale (i.e. that the grade language is heterogeneous).

2. Entities, state variables, and scales

This ABM has two entities: scholars (i.e. the agents) and the peer review system they are in (i.e. the environment). Here follows a list of scholar attributes and a list of global variables. Some of these attributes and variables are static (they do not change during the simulation run), whereas some others are dynamic. The simulation time proceeds in discrete steps: for each time point/simulation step all dynamic scholar attributes and global variables are updated. Unless noted otherwise, these variables will have the same name in the R script as stated here, written in camelCase and possibly with minor spelling differences.

Scholars

Here is the list of scholar attributes.

- Role (dichotomous variable, updated at every time point). A scholar can either serve as author or reviewer and can switch role during the simulation run.

- Behavior (dichotomous variable, updated at every time point). Scholars can either act reliably or unreliably, and the behavior may change over time.

- Resources (continuous variable in [0, +∞], updated at every time point). Scholar resources model the amount/quality of time, tools and skills available for them to invest in authoring or reviewing submissions.

- Submission quality (continuous variable in [0,1], where higher values signify higher quality, updated at every time point). This variable defines the quality of the most recent submission made by scholars and is updated every time a scholar's role is set to 'author'.

- Submission count (integer in [0, time limit], updated at every time point). This is the tally of submissions made by a scholar since the start of the simulation run. An equivalent way to define this variable is the tally of time points during which a scholar's role was 'author'.

- Review count (integer in [0, time limit], updated at every time point). Like the submission count, the review count is the tally of reviews made by a scholar since the start of the simulation (or the tally of time points when the scholar's role was 'reviewer').

- Publication count (integer in [0, submission count + 1], updated at every time point). This is a tally of submissions which were accepted for publication since the start of the simulation run. The ABM assumes that scholars start out at time=1 with a publication count of 1 if their role is 'author'; and 0 otherwise.

- Got published (logical; updated at every time point). This variable is flagged true if the most recent submission by a scholar was accepted for publication; false otherwise. At the start of the simulation (time point =1), all scholars whose role is 'author' are assumed to have had their most recent submission accepted for publication (got published = true).

- Deserved publication (logical; updated at every time point). This variable is set to true for scholars with role = 'author' and with a submission quality higher than a publication quality threshold (see global variables in the next section).

- Expected quality (continuous in [0, 1], updated at every time point). The expected quality refers to the quality that a scholar's submission can be expected to have at any given time point, if the scholar's role at that time point is 'author'.

- Network (or 'nw' in the script - scholar index, updated at every time point). This variable implements the reviewer network as an undirected network. The ABM assumes that, at each time point, half the scholars are 'reviewers' and the other half are 'authors'; each author is reviewed by one randomly chosen reviewer. For authors, the network variable stores the index of the reviewer by whom they are currently being reviewed; conversely, for reviewers this variable stores the index of the author whose submission they are reviewing.

- Cost of authoring (continuous variable in [0, +∞], updated at every time point). This variable defines how many scholar resources are being spent by a scholar whose role is 'author'.

- Cost of reviewing (continuous variable in [0, +∞], updated at every time point). Similar to the above, this variable defines the resources spent by a 'reviewer' scholar.

- Reviewer assessment (continuous variable in [0, 1], updated at every time point). This is the grade that 'author' scholars have received by the reviewer for their (i.e. the authors') current submission.

- Discretization thresholds (a tuple of continuous values in [0,1], constant throughout the simulation run; in the code, scholars' discretization thresholds are stored in a list named "gradeLanguages"). Discretization thresholds are used to convert the assessment of a submission from the continuous range [0, 1] to a discrete range with as many steps as defined by the global variable "evaluation scale" (see next section). When scholars have different thresholds, it means that they have different grading standards (i.e. the grade language is heterogeneous).

- Discrepancy (continuous variable in [0, 1], updated at every time point). Absolute difference between authors' self-evaluation of their own submission and the reviewer's assessment. This variable models the author's self-perceived *fairness* in the review they received.

Peer review system

This list contains the global variables (or 'parameters') of the ABM, which are constant throughout the simulation run:

- Random seed (any 32-bit integer). Seed for the random number generation.

- Number of scholars (even integer ≥ 10; in the code this is called "numberOfAgents). This indicates the population size. Since there must be exactly one reviewer for each author, the population size must be an even number.

- Acceptance rate (continuous variable in [0, 1). This variable defines the proportion of submissions which will be accepted for publication at a given time point.

- Scenario (factor; in the code: "SGscenario"). Can be "no reciprocity", "indirect reciprocity", "self-interested authors", "fairness" or "fair authors". This is the main independent variable in SG, and it is an important one in our model as well. See section "Design concepts/Basic principles" for details.

- Grade language (dichotomous). Can either be "homogeneous" (i.e. all scholars have the same discretization thresholds), or "heterogeneous" (discretization thresholds may vary from scholar to scholar).

- Resources gain (continuous variable in [0, +∞)). This variable defines how many resources are added to scholars' resources at each time point.

- Resources gain factor (continuous in [1, 1.5]). This is a multiplier used to calculate how many resources are gained at a given time point by 'author' scholars.

- Baseline evaluation bias (continuous in [0, +∞]). This is a multiplier used in the rate function (see related entry in the section 'submodels'), whereby reviewers produce their assessment of a submission.

- Probability of unreliable reviewers (continuous in [0, 1]). For some scenarios, scholar's behavior is set at random via a Bernoulli trial: this variable defines the probability of success (i.e. behavior = unreliable).

- Standard deviation of submissions' quality (continuous in [0, +∞]; in the code: "sdSubmissionQuality"). This variable is used in the calculation of scholars' submission quality.

- Evaluation scale (integer ≥ 2). This variable defines the number of categories in the evaluation scale: the higher the value, the more fine-grained the evaluation scale.

- Time limit (integer > 0). This variable defines how many iterations of authoring/reviewing scholars will go through.

In addition to the global variables, we also calculate the outcome variables outlined in the accompanying paper and following SG. These variables are updated at the end of each time point (see section: "Process overview and scheduling").

- Evaluation error (in the code named 'evaluationBias' after the original code in SG). Let the number of wrong reviews be the number of scholars who were not accepted for publication (got published == false) despite deserving to be accepted (deserved publication == true). Eval-

uation error is defined as the number of wrong reviews divided by the number of accepted submissions (i.e. number of scholars that got published == true), times 100.

- Resource loss. Let the quality of the best submissions be the sum of the submission quality of scholars who deserved to be accepted for publication (deserved publication == true). Let the quality of published submissions be the sum of the submission quality of scholars who got published. Resource loss is defined as the difference between the quality of the best submissions and the quality of published submissions, times 100.

- Reviewing expenses. This is the sum of scholars' cost of reviewing divided by the sum of scholars' cost of authoring, times 100.

- System productivity. Defined as the sum of scholar resources.

- Gini index. The Gini index is measured over scholar resources.

3. Process overview and scheduling

The ABM is divided into two steps: initialization and simulation. The simulation is run for as many discrete time steps as specified by the global variable "time limit". Here follows the pseudocode for the ABM.

Whenever possible, the name of the calls to functions indicated in the pseudo code correspond to the function names used in the code (where they will be in camelCase). All these functions will be defined in the section "Submodels".

initialization	`Set random seed for random number generation` `Create scholars`

simulation	`Repeat as many times as specified by "time limit":`
	`  Update scholar resources`
	`  Define scholar roles`
	`  Calculate scholar expected quality`
	`  Define review network`
	`  Define scholar behavior`
	`  Prepare submissions`
	`  Rate submissions`
	`  Discretize reviewer assessments`
	`  Update scholar attributes`
	`  Update outcome variables`
	
	`Return outcome variables`

Note: some of these functions entail loops over all 'author' scholars or over all 'reviewer' scholars. The resulting updates are carried out asynchronously and following the order of scholars' index number. However, an implementation with synchronous updates and/or a randomized order would be fully equivalent. This is because there are no interdependent interactions among agents (or between agents and the environment) occurring during the execution of these functions.

4. Design concepts

Elaborating on the eleven design concepts (only some of which are applicable to this ABM):

Basic principles

An overview of the basic principles can be divided in two parts: on scholar behavioral strategies, and on the assumptions about the evaluation scale and grade language.

Behavioral strategies relate to the fact that, in peer review, scholars play two roles: as authors and reviewers. Since scholars' time and resources are finite, scholars face a trade-off: how much effort to put into authoring vs into reviewing. The assumption we make is that the more resources (e.g. time) are spent on one task, the fewer remain available for the other task. Furthermore, the more resources are spent, the better the result (for authors, higher quality submissions; for reviewers, more accurate reviews).
Previous research has explored a set of heuristics scholars may adopt when deciding how much to invest in authoring or reviewing (see SG). These heuristics are captured by the global variable "scenario": each scenario defines the set of rules based on which scholars decide whether to adopt a reliable behavior (i.e. invest a lot in a task) or unreliable (invest less):

- Scenario "no reciprocity": reviewers behave unreliably with a probability as of "probability of unreliable reviewers"; reliably otherwise. Authors are always reliable.
- Scenario "indirect reciprocity": reviewers only behave reliably if their own most recent submission was accepted for publication (got published == true). Authors are always reliable.
- Scenario "fairness": Reviewers calculate how fairly their own most recent submission was evaluated. If their previous submission was given a grade close to what they believe was its true quality (discrepancy ≤ 0.1), they behave reliably. Authors are assumed to be always reliable.
- Scenario "self-interested authors": Reviewers behave as in the "indirect reciprocity" scenario (i.e. reliable only if their most recent submission was published). Authors are assumed to be reliable only if their previous submission was published (got published == true).
- Scenario "fair authors": Reviewers behave as in the "fairness" scenario (i.e. reliably only if discrepancy ≤ 0.1). Likewise, authors are assumed reliable only if their previous submission was evaluated fairly (again, if discrepancy ≤ 0.1).

Authors' submission quality and cost of authoring will be higher when authors behave reliably; similarly, reviewers' accuracy and cost of reviewing will be higher in case of reliable behavior.

Another set of basic principle concerns the assumptions on the evaluation scale and grade language. In real-world peer review systems, when reviewers are asked to grade a submission (i.e. to give it a score), the evaluation scale often resembles a Likert scale (e.g. ranging from "very poor submission" to "outstanding submission"). A reviewer's final score will depend (1) on the very quality of the submission; (2) on the granularity of the scale (i.e. how many categories there are in the evaluation scale); (3) on reviewer's own understanding of the categories of the evaluation scale.
Relative to the original model by SG, our work adds and explores the granularity of the scale (point 2) and heterogeneity in reviewers' grading standards (point 3). The granularity of the scale is modeled via the global variable "evaluation scale"; heterogeneity in grading standards is modeled via "grade language". Specifically, we model grade language heterogeneity by giving scholars random discretization thresholds, so that a submission quality that a reviewer would rate "outstanding" may be given a lower score by a more severe reviewer (i.e. a reviewer with a higher threshold for what qualifies as outstanding).
Our hypothesis is that finer-grained evaluation scales (i.e. evaluation scales with more categories), by allowing more precise evaluations, reduce evaluation noise and thus improve peer review (vis-à-vis the outcome variables states above). Furthermore, heterogeneity in the grade language may add noise and affect some scenarios more than others; thus, we would expect scenarios to perform differently with or without grade language heterogeneity.

Emergence

Scenarios, granularity and heterogeneity affect different aspects of peer review, including its efficacy (the degree to which it successfully identifies submissions with the highest quality), efficiency (i.e. the cost of running the peer review system), and equality (i.e. the degree to which peer review hinders the accumulation of research resources into the hands of few scholars.
Of the outcome measures listed above, "evaluation error" indicates the efficacy of peer review; "resource loss" is an indicator of efficiency, and the "Gini index" measures the inequality in the distribution of resources.

Adaptation

Scholars feature adaptive strategies in all scenarios other than "no reciprocity". To summarize what was explained in detail above, in these scenarios scholars decide whether to invest a lot of resources in their work based on their past experience (e.g. on whether they were successful authors in the previous round, or whether they perceived their most recent submission was evaluated fairly by the reviewer).

Objectives

Scholars' objective is to maximize their resources. This can be achieved by spending less of the resources available (i.e. behaving unreliably by authoring poor quality submissions and/or making hasty/inaccurate reviews). Alternatively, resources can be increased by getting published: investing more resources as authors produces higher quality submissions which carry a higher chance of being accepted for publication – publication, in turn, yields greater resources.

Learning

Agents in this ABM exhibit adaptive behavior but not learning.

Prediction

In this ABM, agent behavior is defined as causal, not teleological: agents pick a course of action depending on the stimuli received during previous interactions; not in an attempt to achieve a goal. This said, there is the implicit assumption that scholars strive to maximize their resources. This ABM allows to explore which of the scenarios (i.e. behavioral strategies) are best for achieving this (but, to state this again, resources are not used by agents to guide their behavioral choices).

Sensing

Scholars only process two signals from their social environment: the reviewer assessment received on their latest submissions ("reviewer assessment"), and their previous success as authors ("got published"). Reviewer assessment is signaled by the one reviewer who has reviewed the submission; the previous success also depends on the quality of the other competing submissions and the assessment they received. This links to the next section:

Interaction

This model entails both direct and indirect interactions. Direct interactions are dyadic and play out via the review network: authors send a signal to their review in the form of a submission, and the reviewer responds by returning a reviewer assessment. Indirect interactions concern the competition for achieving publication and, ultimately, collecting more resources.

Stochasticity

Stochasticity plays a role in various instances:

- In the calculation of submission quality. Here, randomness is meant to model variability in submission quality (whose causes are not worth, or possible, to model explicitly here).

- In the calculation of reviewer assessment (i.e. in the rating function). Randomness captures all variability in reviewer cognitive process other than their accuracy (which depends on their behavioral choice) and interpretation of the grade language.
- In the definition of the review network. Here, randomness is used to mimic interactions in a community where all pairs of scholars may eventually interact.
- In some scenarios (i.e. "no reciprocity"), in the choice of reviewer behavior. In this case, randomness defines whether reviewers will be reliable or unreliable, and thus serves as benchmark for comparing all other scenarios where reviewer behavior is deterministic.
- In the creation of discretization thresholds. If "grade language" is set to "heterogeneous", then these thresholds are drawn at random from a uniform distribution. This models scholars' variation in their interpretation of the grade language.

Collectives

This ABM does not model collectives other than the whole agentset.

Observation

The output of the ABM consists of the five outcome variables as measured in the last time point of the simulation. These outcome variables are calculated on the whole population of agents, and for the sake of the analyses they are averaged across a battery of independent simulation runs initialized with the same parameter configuration and a different random seed.
Regarding the use of simulation data for the accompanying paper, the process of running simulation batteries is described in comments to the battery script ("simulation batteries.r"); the extraction and analysis of observation data is carried out and described in comments to the script "plots.r".

5. Initialization

The first initialization step consists of setting the random seed for the random number generation. For this step we rely on the defaults of the R function "set.seed", which is the Marsenne-Twister method. The second step is the creation of the agents (scholars) and the initialization of some of their attributes. These are the attributes that are initialized:

- Resources: set to 0 for all scholars.
- Count submissions: set to 0 for all scholars.
- Count reviews: set to 0 for all scholars.
- Count publications: set to 0 for half of the scholars (chosen randomly with a uniform probability); set to 1 for the other half.
- Got Published: set to 1 for all scholars with "count publications" > 0. Set to 0 otherwise.
- Deserved publication: set to false for all scholars.
- Behavior: defined by a Bernoulli trial: it is set to "unreliable" with a probability "probability of unreliable reviewer"; set to "reliable" otherwise.
- Discretization thresholds. The discretization thresholds will consist of a tuple of N values, where N is the number of categories in the evaluation scale (i.e. global variable "evaluation scale", minus 1).
    - If the global variable "grade language" is set to "homogeneous", the discretization thresholds are the same for all agents. The N values will be the sequence of continuous values which partition the range [0,1] into as many regular partitions as the num-

ber of categories in the "evaluation scale". For example, if "evaluation scale" == 2, then the discretization thresholds will consist of only one value: 0.5. If "evaluation scale" == 5, then there will be N = 4 evenly spaced thresholds: (0.2, 0.4, 0.6, 0.8).

- If "grade language" is "heterogeneous", then each agent's N discretization thresholds will be the ordered list of N values drawn from a uniform distribution in [0, 1]. To clarify, this means that with a heterogeneous grade language each scholar is very likely to have a unique set of discretization thresholds – in other words, a unique interpretation of the grade language.

6. Input data

The model does not use input data to represent time-varying processes.

7. Submodels

The submodels in this ABM are the functions which are called in each time point during the simulation run (see the second row of the table in section "Process overview and scheduling"). Here the functions are explained sequentially, following the order in which they are executed during the simulation of a time step.

- Update scholar resources. The resources of each scholar are calculated as:

  Resources in the previous time step +
  Resources gain +
  Gain by publication (assumed to be zero if unspecified) –
  Cost of authoring (assumed to be zero if unspecified) –
  Cost of reviewing (assumed to be zero if unspecified

- Define scholar role. Half the scholars (chosen at random drawing their index number from a uniform distribution) are set to be "author"; the rest will be "reviewer".

- Calculate scholar expected quality. For each scholar, the expected quality is defined as:

  resources * 0.1 / (resources * 0.1 + 1)

- Define review network. Each author is assigned a randomly chosen reviewer (sampling with uniform probability without replacement).

- Define scholar behavior. Behavior refers to whether the scholar will be "reliable" or "unreliable" and is defined for each agent following a set of rules. The set of rules varies between scenarios (see global variable "scenario"); the rules for each scenario are stated in section "Design concepts / Basic principles".

- Prepare submissions. For each scholar with role = "author", we update the attributes:

- A "multiplier". This is set to 1 by default but is set to 0.1 in case authors submit a low quality submission (which can happen in the scenarios "self-interested authors" and "fair authors").

- Submission quality: drawn from a normal distribution with mean = "expected quality" * multiplier, and s.d. = |"expected quality" * "standard deviation of submissions' quality"|.

- Cost of authoring: defined as: resources * multiplier. "

- Count submissions: count submissions at the previous time point, + 1.

- Rate submissions. For each scholar with role "reviewer" we do the following:

  - We define "reviewed quality" as the submission quality of an author under review.

  - We update the cost of reviewing: for scholars with role == "reviewer" and behavior == "reliable", the cost of reviewing is set to:

    0.5 * resources * (1 + reviewed quality - expected quality)

    For scholars with role == "reviewer" and behavior == "unreliable", the cost of reviewing is:
    0.25 * resources * (1 + reviewed quality - expected quality).

  - We calculate the count of reviews (adding 1 to the count of reviews at the previous time point).

Then, for each scholar with role "author":

- We determine whether the behavior of the author's reviewer is "reliable" or "unreliable".

- We calculate a multiplier. This is set to 1 if the author's reviewer is "reliable"; if "unreliable", then the value of the multiplier is picked at random (uniform) between 1.9 and 0.1.

- We draw a value from a uniform distribution with mean = resources * multiplier, and s.d. = |resources * "baseline evaluation bias" * multiplier|.

- If the value so obtained is lower than 0, we set it to 0.

  - The raw reviewer assessment is defined as: value * 0.1 / (value * 0.1 + 1).

- Discretize reviewer assessments. Each raw reviewer assessment (a real number) needs to be discretized according to the reviewer's interpretation of the grade language (i.e. by using the reviewer's discretization thresholds). Thus, we index the intervals between each reviewer's thresholds from 0 (the interval ≤ 1st threshold) to N (the interval > last threshold), where N will correspond to the number of categories -1 ("evaluation scale" - 1).

  Each author's reviewer assessment is then defined as the index of the interval corresponding to the raw reviewer assessment, divided by N. This will return values in the range [0, 1].

- Update scholar attributes. This entails looping through all scholars with role == "author", and updating different attributes:

  - Discrepancy. This is defined as the absolute difference between reviewer assessment and author's self-assessment. The self-assessment is equal to the submission quality, discretized using the method described above, but using the author's own discretization thresholds instead the reviewer's thresholds.

  - Got published. Scholars with role == "authors" are ranked by their "reviewer assessment". "Got published" is set to true for all authors whose reviewer assessment is among the K highest. K is calculated as "number of scholars" * "acceptance rate". Other authors do not get published, so got published = false.

  - Deserved publication. Similar to the above, we rank "author" scholars by their "submission quality". For the K (K = "number of scholars" * "acceptance rate") authors with the highest submission quality, "deserved publication" is set to true; false otherwise.

  - Count publications. For scholars with role == "authors" and got published == true, the "count publications" is updated by adding 1 to the "count publications" of the previous time point.

- Update outcome variables. We update the global variables "evaluation error", "resource loss", "reviewing expenses", "system productivity" and "Gini index". These variables were defined in section "Entities, state variables, and scales".

References

Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S. K., Huse, G., Huth, A., Jepsen, J. U., Jørgensen, C., Mooij, W. M., Müller, B., Pe'er, G., Piou, C., Railsback, S. F., Robbins, A. M., … DeAngelis, D. L. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, *198*(1–2), 115–126. https://doi.org/10.1016/j.ecolmodel.2006.04.023

Grimm, V., Berger, U., DeAngelis, D. L., Polhill, J. G., Giske, J., & Railsback, S. F. (2010). The ODD protocol: A review and first update. *Ecological Modelling*, *221*(23), 2760–2768. https://doi.org/10.1016/j.ecolmodel.2010.08.019

Squazzoni, F., & Gandelli, C. (2013a). Opening the Black-Box of Peer Review: An Agent-Based Model of Scientist Behaviour. *Journal of Artificial Societies and Social Simulation*, *16*(2). https://doi.org/10.18564/jasss.2128

Squazzoni, F., & Gandelli, C. (2013b). *Peer Review Model* (Version 1.1.0) [Computer software]. CoMSES Computational Model Library. https://www.comses.net/codebases/3145/releases/1.1.0/

Squazzoni, F., & Gandelli, C. (2012). Opening The Black-Box Of Referee Behaviour. An Agent-Based Model Of Peer Review. *ECMS 2012 Proceedings Edited by: K. G. Troitzsch, M. Moehring, U. Lotzmann*, 647–653. https://doi.org/10.7148/2012-0647-0653