

Technical description of an agent-based model for testing the effect of communication, trust and belief revision methods in a collaborative traffic scenario

Marcelo Souza, Andrew Koster and Ana Bazzan

The model description follows the ODD (Overview, Design concepts, Details) protocol for describing individual- and agent-based models (Grimm et al. 2006, 2010¹).

1. Purpose

Question: What is the purpose of the model?

Answer:

The model's purpose is twofold. Firstly it aims at accurately modeling traffic conditions in a simplified representation of a real road network. Secondly, it aims at modeling the effect of communication, potentially false, and countermeasures to false communication within this network.

2. Entities, state variables, and scales

Questions: What kinds of entities are in the model? By what state variables, or attributes, are these entities characterized? What are the temporal and spatial resolutions and extents of the model?

Answer:

Road network: the model consists of a simplified road network with two edges: GPU and GSU that both lead from G to U. The travel time along each edge is dependent on the traffic volume along these roads. This is calculated using the formula $t = t_0 * [1 + \alpha * (V/C)^\beta]$. C , t_0 , α and β are parameters that need to be extracted from real world data, with t_0 being the travel time in free flow conditions, C the capacity of the road, and α and β are tuning parameters. In the simulated network, we choose $t_0 = 11$ minutes, $\alpha = 0.2$ and $\beta = 10$ for both edges, $C_{gsu} = 10000$ and $C_{gpu} = 3000$, in accordance with information from the local traffic authorities. The variable V is computed by the model as the sum of all the cars traveling along the edge. This is influenced by two additional parameters, the cars that are not explicitly simulated as agents. This background traffic is configured through two further parameters of the simulation: $volume_{gsu}$ and $volume_{gpu}$. Thus, the parameters representing the road network can be represented as the vector:

(t_0 , α , β , C_{gsu} , C_{gpu} , $volume_{gsu}$, $volume_{gpu}$)

Agents: the agents in this model represent cars that can decide for themselves which of the two edges to pick to go from G to U. Each agent belongs to one of three different groups: *private drivers*, *professional drivers* or *authority drivers*. These different groups are used to initialize the simulation with heterogeneous behaviors, but other than that, all agents function in the same manner.

¹ This is a reference to the ODD “first update” manuscript.

In each iteration, an agent decides, based on its beliefs about the roads, to drive along either GSU or GPU. There are three beliefs which influence its behavior: `gsu_congested`, `gpu_congested` and `gpu_dangerous`. Each agent then decides what road to take according to the following table:

IF			THEN
<code>gsu_congested</code>	<code>gpu_congested</code>	<code>gpu_dangerous</code>	
*	*	True	gsu
True	*	Unknown	random
True	True	False	random
True	Unknown	False	gpu
True	False	False	gpu
Unknown	*	Unknown	gsu
Unknown	True	False	gsu
Unknown	Unknown	False	random
Unknown	False	False	gpu
False	*	Unknown	gsu
False	True	False	gsu
False	Unknown	False	gsu
False	False	False	random

If the agent chooses randomly between the two roads, he picks either one with a probability of 0.5. An asterisk in the table means that the rule is valid for any value of that variable.

In order to choose, the agent must thus compute its beliefs. This is done according to a belief change operator. Which belief change operator is used, is configurable, with choices from: *Basic*, *Pereira et al.* and *Koster et al.* described in detail in the respective papers. Here we summarize the operators.

The *Basic* operator does not use a trust model to evaluate sources, and simply chooses to believe what the majority of agents communicated. If equal numbers of agents communicate a proposition and its negation, the agent believes neither. If an agent using the *Basic* operator does not receive any communication, it maintains the same beliefs as in the previous iteration.

Pereira et al.'s operator uses the trust model (described below) to compute the trustworthiness of all sources, for each of the three propositions. Let a be a proposition and $\text{maxtrust}(a)$ be the maximum of the trustworthiness over all sources that communicated a , then the agent believes a iff $\text{maxtrust}(a) > \text{maxtrust}(\sim a)$. Otherwise, the agent believes $\sim a$ iff $\text{maxtrust}(\sim a) > \text{maxtrust}(a)$. Finally, if $\text{maxtrust}(a) = \text{maxtrust}(\sim a)$ the agent believes neither a nor $\sim a$. If an agent using *Pereira et al.*'s operator does not receive any communication, it maintains the same beliefs as in the previous iteration.

Koster et al.'s operator follows the algorithm below to compute the beliefs:

In this model, **constructargs** always constructs the following arguments:

{({gsu_congested}, gsu_congested), ({~gsu_congested}, ~gsu_congested), ({gpu_congested}, gpu_congested), ({~gpu_congested}, ~gpu_congested), ({gpu_dangerous}, gpu_dangerous), ({~gpu_dangerous}, ~gpu_dangerous)}

Given that there are 3 pairs of mutually attacking arguments, the second part of the algorithm (line 16 onwards) simply selects one of each such that the level of this belief is higher than the level of the other. In the case where the level of both a proposition and its negation are exactly equal, this proposition is unknown to the agent.

At the start of each iteration, the prior opinion a for each proposition is set to the *normalized level* of the belief in the previous iteration:

In the case that the agent does not receive any communication in an iteration, this thus results in it choosing the road it did in the round before (because b and d will be 0, and u will be 1 for each argument)

The initial state for agents using the *Basic* or *Pereira et al.* operator is that $gsu_congested$ and $gpu_congested$ are unknown, whereas the value for $gpu_dangerous$ is a parameter of the model. For the *Koster et al.* operator, the initial state is that we set the binomial opinion for $gsu_congested$ and $gpu_congested$ to $\langle 0, 0, 1, 0.5 \rangle$ and for $gpu_dangerous$ to $\langle 0, 0, 1, discomfort \rangle$ with $discomfort$ a parameter of the model.

The *trust model* used by two of the belief change operators functions as follows: agents maintain a trust value for each group of agents (private, professional or authority drivers) and for each type of information ($gsu_congested$, $gpu_congested$, $gpu_dangerous$). Initially, each value is set to 0.5 (meaning the agent knows nothing about the trustworthiness). After an iteration in which an agent received communication at the start of an iteration, the trustworthiness is updated at the end of that iteration, and passed to the next iteration. At the end of an iteration, after moving, agents receive full information about the true state of the road network in that iteration. They can then evaluate for each message whether it was truthful or not. Let C be a class of agents and T be a type of information then:

$$\text{Trust}(C, T) = (|\text{Truthful}(C, T)| + 1) / (|\text{Total}(C, T)| + 2)$$

The trustworthiness of an individual agent is simply the trust in the group it belongs to, and dependent on the content of the message that it sends.

Truthful and Liar agents: each agent is either truthful or a liar. A truthful agent always communicates the true state of the roads that it traveled along. If it traveled along the GSU edge it communicates truthfully whether that edge was congested or not. If it traveled along the GPU edge it communicates truthfully whether that edge was congested or not, and whether it encountered danger or not (in our simulation there is never any danger). Liar agents simply always communicate falsely (if a road is congested, it communicates the road is clear and vice versa. The same occurs for communicating danger). Moreover, because liar agents are out to break the system (potentially for some strategic motive, potentially just for the sake of vandalism), we simulate the possibility of a sybil attack occurring, similar to the one described by Ben Sinai et al. (2014). For this reason we include a parameter $sybils$: for every liar agent in the system that communicates, there are multiple messages sent, equal to the number of sybils. In other words, if $sybils = 4$, every liar agent sends out 4 messages.

At any iteration, an agent can thus be represented by a vector containing:

- The agent's class
- Whether the agent is truthful or a liar
- The belief change operator it uses
- The trust values computed in the previous round
- The messages received in the previous round
- The level of the beliefs for each proposition in the previous round (or in the case of *Pereira et al.* or *Basic* operator, simply the route chosen is sufficient).

Environment: as the environment, we understand, the global parameters that are not related to the road network. Thus the environment defines how many iterations the simulation will run for, how many agents there are of each class and what percentage of the agents are liars. We also configure what percentage of agents sends communication each round, and what percentage of agents receives it. These together are an approximation of the reality where not all drivers participate in the use of a collaborative traffic app, and not all of those using such an app actively report situations they encounter. Moreover, there is a Boolean parameter for whether agents communicate about the danger they perceive on road GPU or not. This can be seen as a new feature in the collaborative traffic app. Moreover, it serves to see whether (trustworthy) communication can help to overcome false, but commonly held beliefs, in the environment. In the environment we also set the distribution for the initial discomfort level (the prior belief that the rout GPU is dangerous). This is a normal distribution with a configurable mean and standard deviation.

The environment can thus be represented as a vector:

(Iter, N_private, N_prof, N_auth, perc_liar_private, perc_liar_prof, perc_liar_auth, perc_send, perc_receive, comm_danger, mean_discomfort, stddev_discomfort)

3. Process overview and scheduling

The simulation process consists of a setup, and then a run phase. In the setup phase, all agents are initialized: they are assigned, probabilistically according to their class whether they are a liar or not, and their discomfort level is set. Their trust in everybody in all situations is set to 0.5 and their beliefs about the road conditions are set to unknown, except for gpu_dangerous, which is true if discomfort > 0.5 and false if discomfort < 0.5.

In the run step the agents run for a fixed set of iterations. In each iteration, the following happens. Let t be the current round.

1. Each agent receives the truth about congested state of roads and danger in round $t - 1$. This information is only used to compute the trust values.
2. The agents that received communication in round $t - 2$ compute the trustworthiness of all agent classes and message types based on the info of round $t - 1$. The other agents do nothing.

3. The agents that will communicate in round t are chosen (up to the designated percentage).
4. Each agent chooses what road to drive on.
5. Compute the travel time along both roads.
6. All agents arrive and know how long it took. If it took longer than double the free flow time (so in the simulation, longer than 22 minutes), then the road is considered congested.
7. The agents that were chosen to communicate in step (1) send their messages (either truthfully or lies, depending on their configuration).
8. The agents designated to receive communication receive the messages
9. Round t ends.

4. Design concepts

Questions: There are eleven design concepts. Most of these were discussed extensively by Railsback (2001) and Grimm and Railsback (2005; Chapter. 5), and are summarized here via the following questions:

Basic principles. Which general concepts, theories, hypotheses, or modeling approaches are underlying the model's design? Explain the relationship between these basic principles, the complexity expanded in this model, and the purpose of the study. How were they taken into account? Are they used at the level of submodels (e.g., decisions on land use, or foraging theory), or is their scope the system level (e.g., intermediate disturbance hypotheses)? Will the model provide insights about the basic principles themselves, i.e. their scope, their usefulness in real-world scenarios, validation, or modification (Grimm, 1999)? Does the model use new, or previously developed, theory for agent traits from which system dynamics emerge (e.g., 'individual-based theory' as described by Grimm and Railsback [2005; Grimm et al., 2005])?

Answer: The main modeling approach used is agent-based modeling (ABM). This model is composed by the entities mentioned before. Their main inter relationships occur due to the use of a shared resource, the edges of the transportation network. To replicate this shared use, the aforementioned formula

the formula: $t = t_0 * [1 + \alpha * (V/C)^\beta]$

is used to compute the travel time of agents in the road network. Note that if $V \gg C$, time t increases rapidly. This is meant to model overcapacity. We remark that overcapacity does occur because C is just the nominal capacity.

The system dynamics emerges from the fact that when agents make decisions about route choices, then some edges will have over capacity leading agents to shift their choices and so on, until a kind of equilibrium is reached.

Emergence. What key results or outputs of the model are modeled as emerging from the adaptive traits, or *behaviors*, of individuals? In other words, *what* model results are expected to

vary in complex and perhaps unpredictable ways when particular characteristics of individuals or their environment change? Are there other results that are more tightly imposed by model rules and hence less dependent on what individuals do, and hence 'built in' rather than emergent results?

Answer: The emergent property that is modeled is the distribution of vehicles in the routes. This is affected by communication about the traffic status of routes, and in particular by the fact that false information is eventually communicated.

Depending on the initial configuration there are a few potential built-in properties, such as congestion on the route GSU, or congestion on the route GPU.

Adaptation. What adaptive traits do the individuals have? What rules do they have for making decisions or changing behavior in response to changes in themselves or their environment? Do these traits explicitly seek to increase some measure of individual success regarding its objectives (e.g., "move to the cell providing fastest growth rate", where growth is assumed to be an indicator of success; see the next concept)? Or do they instead simply cause individuals to reproduce observed behaviors (e.g., "go uphill 70% of the time") that are implicitly assumed to indirectly convey success or fitness?

Answer: Agents adapt to the environment by selecting the route that they believe is not congested. In a sense, there is an emergent phenomenon in which a number of agents will shift routes. The individual measure of success is given by the fact that they experience a lower travel time if they select a route that is not congested.

Objectives. If adaptive traits explicitly act to increase some measure of the individual's success at meeting some objective, what exactly is that objective and how is it measured? When individuals make decisions by ranking alternatives, what criteria do they use? Some synonyms for 'objectives' are 'fitness' for organisms assumed to have adaptive traits evolved to provide reproductive success, 'utility' for economic reward in social models or simply 'success criteria'. (Note that the objective of such agents as members of a team, social insects, organs—e.g., leaves—of an organism, or cells in a tissue, may not refer to themselves but to the team, colony or organism of which they are a part.)

Answer: The system-wide objective is to balance the load, i.e., obtain a fair distribution in which some agents select one route, while others select another one. Individual agents have the objective of travelling through the system as fast as possible.

In the particular configuration that we discussed in the paper (the default configuration discussed in this document in the section on initialization), because there is a background traffic that has no route choice, the behavior that is expected is that all agents select one particular route (the peripheral). There is no proper ranking of the alternative actions. Rather, actions are selected according to the table shown in Section 2.

Learning. Many individuals or agents (but also organizations and institutions) change their adaptive traits over time as a consequence of their experience? If so, how?

Answer: There is no explicit learning by the agents, but the refinement of the trust model and updating agents' knowledge about the status of world can be seen as a type of implicit learning: by better modeling the world, the agents achieve better results.

Prediction. Prediction is fundamental to successful decision-making; if an agent's adaptive traits or learning procedures are based on estimating future consequences of decisions, how do agents predict the future conditions (either environmental or internal) they will experience? If appropriate, what internal models are agents assumed to use to estimate future conditions or consequences of their decisions? What tacit or hidden predictions are implied in these internal model assumptions?

Answer: Similarly, there is no explicit estimation of future consequences of decisions. However every iteration agents estimate which route will get them to their destination faster (and safer).

Sensing. What internal and environmental state variables are individuals assumed to sense and consider in their decisions? What state variables of which other individuals and entities can an individual perceive; for example, signals that another individual may intentionally or unintentionally send? Sensing is often assumed to be local, but can happen through networks or can even be assumed to be global (e.g., a forager on one site sensing the resource levels of all other sites it could move to). If agents sense each other through social networks, is the structure of the network imposed or emergent? Are the mechanisms by which agents obtain information modeled explicitly, or are individuals simply assumed to know these variables?

Answer: Agents are able to sense : a) the state of the selected route (i.e., whether or not it is congested), and b) the dangerousness of the peripheral route. Both are environment variables. This knowledge is used both in the decision-making process for the next step, as well as for communicating with other agents.

Apart from this, an important feature of the model is that agents are able to communicate information about these two variables. This communication can be wrong/false due to both inaccurate knowledge or intentional malicious communication (to divert other agents from the selected route).

Interaction. What kinds of interactions among agents are assumed? Are there direct interactions in *which* individuals encounter and affect others, or are interactions indirect, e.g., via competition for a mediating resource? If the interactions involve communication, how are such communications represented?

Answer: Interactions are both explicit (when agents communicate using a formal language based on logic) and implicit, i.e., the fact that they share a limited resource implies that they compete for such resource.

Stochasticity. What processes are modeled by assuming they are random or partly random? Is stochasticity used, for example, to reproduce variability in processes for which it is unimportant

to *model* the actual causes of the variability? Is it used to cause model events or behaviors to occur with a specified frequency?

Answer: The only stochasticity in the environment is in the initialization of the model. Agents are assigned a discomfort level from a normal probability distribution. In addition, at runtime agents decide randomly (for a route) in case they have no (clear) information to decide.

Collectives. Do the individuals form or belong to aggregations that affect, and are affected by, the individuals? Such collectives can be an important intermediate level of organization in an ABM; examples include social groups, fish schools and bird flocks, and human networks and organizations. How are collectives represented? Is a particular collective an emergent property of the *individuals*, such as a flock of birds that assembles as a result of individual behaviors, or is the collective simply a definition by the modeler, such as the set of individuals with certain properties, defined as a separate *kind* of entity with its own state variables and traits?

Answer: Collective behavior is central in this model. As mentioned, individuals of the collective are affected by each other in the sense that they compete for a limited resource. There is no explicit representation of the collective, nor are agents part of any organization. However, agents belong to one among three different groups. Information provided by such groups have different degrees of trustworthiness.

Observation. What data are collected from the ABM for testing, understanding, and analyzing it, and how and when are they collected? Are all output data freely used, or are only certain data *sampled* and used, to imitate what can be observed in an empirical study (“Virtual Ecologist” approach; Zurell et al., 2010)?

Answer: Data collected from the ABM is basically which route the agent has selected. The main measure of performance of the system is the number of agents selecting route GPU.

5. Initialization

Questions: What is the initial state of the model world, i.e., at time $t = 0$ of a simulation run? In detail, how many entities of what type are there initially, and what are the exact values of their state variables (or how were they set stochastically)? Is initialization always the same, or is it allowed to vary among simulations? Are the initial values chosen arbitrarily or based on data? References to those data should be provided.

Answer:

The initialization parameters are defined at start-up.

The parameters that we set at the beginning of our model are:

1. Amount of background traffic on each roads. The default values are 13000 for GSU and 1000 for GPU. These values were chosen in order to ensure that GSU is congested, while GPU is clear. Remember that the goal is that all decision-makers select GPU then.
2. Parameters of the volume-delay formula (t_0 , α , β , C_{gsu} and C_{gpu}). After consulting the local traffic authority, google maps (for real distances) as well as capacity

manuals, these are set to $t_0 = 11$ minutes, $\alpha = 0.2$ and $\beta = 10$ for both edges, $C_{gsu} = 10000$ and $C_{gpu} = 3000$.

3. Number of agents in the system, and their types. The default values are 200 private drivers, 70 professional drivers and 30 authority vehicles. While the exact values were chosen arbitrarily, they are intended to reflect a majority of drivers being private, with a minority of professional drivers and a smaller fraction of authorities. Moreover, each agent is initialized with a discomfort level regarding the GPU route, taken from a normal distribution with configurable mean and standard deviation. These mean and deviation are parameters to be set, in order to initialize the agents. The used values were $\text{mean}=0.7$ and $\text{std. dev.} = 0.2$.

4. Number of iterations. The default value of 100 was chosen after various empirical tests. This value guarantees enough time until convergence (all agents sticking to one route).

5. Percentage of agents that send messages (default: 40%), and percentage of those that receive messages (default: 30%). Additionally, the percentage of liars for each type of agent: private (20%), professional (30%) and authorities (5%), and the number of sybils (default 4). These values are chosen to ensure there is a majority of false information being sent in the system, in order to necessitate a trust model.

6. The belief change operator to be used (*Basic*, *Pereira et al.* or *Koster et al.*)

6. Input data

Question: Does the model use input from external sources such as data files or other models to represent processes that change over time?

Answer:

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

7. Submodels

Questions: What, in detail, are the submodels that represent the processes listed in ‘Process overview and scheduling’? What are the model parameters, their dimensions, and reference values? How were submodels designed or chosen, and how were they parameterized and then tested?

Answer:

There are no submodels.

References

Grimm, Volker, Uta Berger, Finn Bastiansen, Sigrunn Eliassen, Vincent Ginot, Jarl Giske, John Goss-Custard et al. "A standard protocol for describing individual-based and agent-based models." *Ecological modelling* 198, no. 1 (2006): 115-126.

Koster, Andrew, Marcelo Souza, and Ana LC Bazzan. "Liar liar, pants on fire; or how to avoid believing information from untrustworthy sources." *Submitted to Journal of Artificial Intelligence Review*.

Pereira, Célia da Costa, Andrea GB Tettamanzi, and Serena Villata. "Changing one's mind: Erase or rewind? possibilistic belief revision with fuzzy argumentation based on trust." *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence*. Vol. 1. 2011.