Computer Modeling in Philosophy

Jeremiah A. Lasquety-Reyes*

# Towards Computer Simulations of Virtue Ethics

https://doi.org/10.1515/opphil-2019-0029
Received April 30, 2019; accepted September 09, 2019

**Abstract:** This article presents two approaches for computer simulations of virtue ethics in the context of agent-based modeling, a simple way and a complex way. The simple way represents virtues as numeric variables that are invoked in specific events or situations. This way can easily be implemented and included in social simulations. On the other hand, the complex way requires a PECS framework: physical, cognitive, emotional, and social components need to be implemented in agents. Virtue is the result of the interaction of these internal components rather than a single variable. I argue that the complex way using the PECS framework is more suitable for simulating virtue ethics theory because it can capture the internal struggle and conflict sometimes involved in the practice of virtue. To show how the complex way could function, I present a sample computer simulation for the cardinal virtue of temperance, the virtue that moderates physical desires such as food, drink, and sex. This computer simulation is programmed in Python and builds upon the well-known *Sugarscape* simulation.[1]

**Keywords:** virtue ethics, ethics, philosophy, computer simulation, social simulation, agent-based modeling, Python

## 1 Introduction

Though social scientists have used agent-based modeling for many topics with ethical dimensions or concerns, there are only a few agent-based models that explicitly refer to an ethical theory and only a handful of philosophers who have created computer simulations for ethics.[2] Among the few cases, there is currently no agent-based model for virtue ethics. However, I suggest that agent-based modeling is a promising technological instrument to use for virtue ethics research since there is a functional parallelism

[1] The Python source code for the complex way is available at https://github.com/JeremiahLR/Temperance. All the code snippets in this article are also written in Python. I have chosen Python because it is one of the easiest programming languages to learn and to run, especially for philosophers with no programming experience. However, all the code can be rewritten in other programming languages.

[2] The first notable pioneer in this regard would be Peter Danielson with his computer simulation of "instrumental contractarianism" based on the social contract theory of David Gauthier (Danielson, *Artificial Morality*; Gauthier, *Morals by Agreement*). His computer simulation was a computer tournament similar to the famous computer tournament of Robert Axelrod for the iterated prisoner's dilemma (Axelrod, *Evolution of Cooperation*). There are two computer simulations for utilitarianism, the first created by Mascaro et al. exploring the ethical status of suicide, abortion, and rape within an evolving world (Mascaro, Korb, and Nicholson, "Suicide," "ALife"; Mascaro, "Abortion"; Mascaro et al., *Evolving Ethics*) and the second a comparison between act utilitarianism and rule utilitarianism by Vincent Wiegel (Wiegel, "SophoLab"). Finally, the work done by Rainer Hegselmann and Oliver Will simulates Hume's idea of justice as an artificial virtue or a virtue produced by convention (Hegselmann and Will, "Modelling"; Will, "Hume$_{1.0}$").

***Corresponding author: Jeremiah A. Lasquety-Reyes,** MECS, Leuphana Universität Lüneburg, Germany;
E-mail: jlasquetyreyes@gmail.com

between the two. In virtue ethics, a person can possess qualities (virtues or vices) that lead to similar repeated behaviors of an ethical nature; in agent-based modeling, an agent can possess properties and variables that result in more or less predictable behavior as the computer simulation is run.

Person + Qualities → Behavior
Agent + Properties → Behavior

As an ethical theory, virtue ethics is essentially "agent-based."[3] It focuses on the person's overall qualities, the person's character, that lead to similar and repeated moral actions. A just person, i.e. someone who possesses the virtue of justice, is inclined to do just acts such as telling the truth, returning things borrowed, or paying a fair price. An unjust person, i.e. someone who lacks the virtue of justice and instead possesses the vice of injustice, is inclined towards unjust acts such as stealing, cheating, lying, and so forth. In comparison, agent-based modeling is able to give virtual agents different variables, rules, or strategies that dictate their behavior. It can be a single simple rule such as the one found in Schelling's segregation model or can be a complex set of properties such as those found in Epstein and Axtell's *Sugarscape* simulation.[4] This functional parallelism between virtue ethics and agent-based modeling suggests the possibility of using agent-based modeling as a tool to simulate and observe virtue ethics in action inside a controlled environment, which could lead to a greater appreciation and understanding of the ethical theory.[5]

One of the things that attract social scientists and philosophers to agent-based modeling is the possibility of conducting social and ethical "experiments" that are impossible to do in real life. Ethical theories often assume and make generalizations about human behavior that cannot be tested or replicated as one can do with physical experiments in a laboratory. However, a computer simulation representing certain behaviors of human beings can be run an indefinite number of times with different conditions and variables. The results can then be compared with each other and also quantitatively analyzed. In this respect, computer simulations could be seen as more complex, robust, and precise counterparts to the thought experiments that philosophers sometimes employ. For this reason, Mascaro et al. called their project a "new experimental ethics" and Wiegel called his project "experimental computational philosophy."[6] According to them, computer simulation adds a new opening for experimentation and understanding previously unavailable to philosophical ethics.

But how does one simulate virtue ethics? I suggest that there is a simple way and a complex way. The simple way conceives of virtue as a numeric variable invoked during specific events or situations in a simulation; the complex way conceives of virtue as the result of the interaction between physical, cognitive, emotional, and social components in an agent. I describe these two ways below.

## 2 The simple way

The simple way treats virtue as a numeric variable. This numeric variable will have an effect on the agent's behavior in certain events or situations. For example, in some social simulations such as *Sugarscape*, agents are

[3] I focus primarily on classical virtue ethics which traces its origins to Aristotle's *Nicomachean Ethics*, received significant development in the medieval period, and experienced a revival in the 20th century through the work of Alasdair MacIntyre (MacIntyre, *After Virtue*).

[4] Schelling "Models," *Micromotives and Macrobehavior*; Epstein and Axtell, *Growing Artificial Societies*.

[5] The kind of agent-based modeling I propose for virtue ethics is different from, but at the same time potentially complementary to, another agent-based model often associated with ethics, Axelrod's computer tournament for the iterated prisoner's dilemma (Axelrod, *Evolution of Cooperation*). The prisoner's dilemma requires two players who each either seek their maximum advantage at the expense of the other or who cooperate for a lesser reward. Axelrod collected many different strategies from around the world that agents could play in this game. The most effective strategy, TIT FOR TAT (cooperate in the first move then copy everything that the other player does after that) has been described as "nice," "retaliatory," and "forgiving" all at the same time (Hegselmann, "Moral Dynamics," 5680). It seems that a chosen strategy may reflect virtue/vice qualities. However, the prisoner's dilemma, as well as other games used in game theory, require at least two players. Virtue ethics, on the other hand, deals with interactions between persons as well as situations where a person is alone. Therefore, a computer simulation of virtue ethics would ideally be more general than a computer simulation of game theory. They can be usefully combined in a computer simulation if, for example, an agent's development in virtue through non-game situations contributes to its strategy for game situations.

[6] Mascaro et al., *Evolving Ethics*; Wiegel, "SophoLab."

able to engage in trade with each other. Let us suppose that whenever agents engage in trade, there is always a chance to cheat during the transaction. Whether or not the agent decides to cheat is determined by the agent's level of virtue compared to a randomly generated number. In this case, the relevant virtue is the virtue of justice.

```
import random
agent_justice = 75
learning_rate = 1
if (trading):
        random_number = random.randint(1, 101)
        if (random_number >= agent_virtue):
                cheat()
                agent_justice -= learning_rate
        else:
                do_not_cheat()
                agent_justice += learning_rate
```

In the code snippet, *import random* ensures that we can obtain random numbers. The virtue (*agent_justice*) is a number between 0 and 100, in this case, 75. A *learning_rate* is set to determine the increase or decrease of the virtue after a moral choice is made. This is because according to virtue ethics theory, a habit, whether a virtue or vice, is strengthened by repeated actions of a similar nature.[7] Therefore, the act of cheating will decrease the virtue of justice while the act of not cheating (i.e. resisting the temptation to cheat) will increase the virtue of justice by the chosen learning rate. Each time the agent engages in trade, a *random_number* is generated to be compared against *agent_justice*. If the *random_number* is higher than *agent_virtue* then the agent cheats; if it is lower, then the agent does not cheat. What these two functions *cheat()* and *do_not_cheat()* actually do is left to the programmer.

This is the simplest way of simulating virtue. It is similar to role-playing games where a character possesses statistics like wisdom and intelligence with certain values, a player rolls the dice at special situations, and the character either performs an action or not. In spite of its simplicity, this mechanism captures one interesting aspect of virtue ethics, namely the aspect of likelihood. When we say that a person is just, we do not mean that the person does just actions *all* the time, but rather that the person is expected to do just actions *most* of the time. It is still possible for a just person to commit an unjust action once in a while. However, this should not completely detract from the person's general state of justice.

Another implementation of the simple way is to treat virtue not as a probability value but rather as a level of skill in confronting certain situations. We can also introduce a higher level of abstraction here. For example, let us represent an event or situation as a mathematical function such as the one in the following graph:

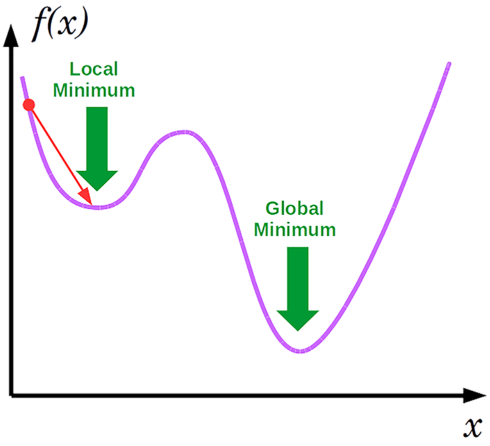

**Figure 1.** A mathematical function to abstractly represent an event or situation.

[7] As Aristotle says, "we become just by doing just actions, temperate by doing temperate actions, brave by doing brave actions" (*Nicomachean Ethics*, 1103b).

This mathematical function (Figure 1), which we could call an “event-function,” can represent any event or situation in the simulation that can be addressed by a virtue, e.g. buying a car with prudence, eating cake with self-control, fighting in a war with courage, and so on. Whenever an agent encounters an event-function, it performs *gradient descent* on the function. Gradient descent is an algorithm commonly used in machine learning that enables one to go from any starting point in the function towards one of the lowest points (a *minimum* among several *minima*) of the given function. How far one is able to move from the starting point to a minimum is determined by the number of steps multiplied by the size of the step. For example, let us say that in Figure 1 the movement from the red starting point to the nearest local minimum takes 100 steps if the step size is 1 or 50 steps if the step size is 2. The programmer can fix the step size as a constant, while the level of virtue can equate to the number of steps that the agent will make. To see how this works, let us once again take the trading example.

```
trade_event = generate_mathematical_function()
agent_justice = 75
if (trading):
        starting_point = check_encountered_before(trade_event)
        new_point = gradient_descent(starting_point, agent_justice)
```

First, we generate a mathematical event-function to represent the trading situation in the simulation. Then we have a value for *agent_justice*, in this case, 75. When the agent engages in trade, it finds a starting point in the event-function. If the agent has traded before, then the starting point is based on its most recent experience. If it is the agent’s first time to trade, then we assign the agent a semi-random starting point (I suggest semi-random because we do not want the agent to be extremely far from any minima). Finding the starting point is done through the function *check_encountered_before()*. The agent then performs gradient descent from the starting point to move down to a new point in the event-function. The number of steps the agent takes towards a minimum is determined by the value of *agent_justice*. Assuming each point of virtue is equivalent to one step, then the agent in the above example would be able to take 75 steps down towards a minimum. The new position after this process is then recorded as *new_point* and this variable will determine the agent’s starting point the next time it engages in trade. This movement from one point to another is how an agent learns within this context.

The closer an agent comes to a local minimum of the event-function, the more it can be said that it has addressed the situation in a virtuous way. A local minimum represents an ethically optimal response to the given situation. If the agent reaches the global minimum, i.e. the absolutely lowest point in the event-function, then the agent can be said to have acted in the most perfect ethical way in the situation.

The advantage of this mechanism is that it is more flexible. One can generate an indefinite number of event-functions and decide later on what kinds of events and situations they ought to represent. One is not restricted by the need to hard code situations in the program. In addition, these event-functions can have resemblances with each other. The chosen event-functions for buying a house and buying a car may look more similar in terms of their shape and contours when graphed than with the event-function for fighting in a war. One could easily simulate the same virtue being applied to different but similar situations; the virtue of prudence could practically play the same role in buying a house and in buying a car. Furthermore, one can have situations that are objectively more difficult than others. The difficulty of an event-function is determined by its height as well as how far the agent’s semi-random starting point is from any local minimum. We could perhaps select a more difficult event-function for fighting in a war than for buying a car.

It is also not necessary to address an event-function with just one virtue. We could require a combination of virtues to determine the progress of gradient descent. For example, instead of only the virtue of justice, we could require both the virtues of justice and temperance in trade situations. This allows for more nuanced interpretations according to the goals of the simulation. Vices can also be implemented with negative values. We could employ gradient ascent instead of gradient descent and maxima instead of minima. The consequences for an agent reaching any minimum (through virtue) or maximum (through vice) of an event-function are left to the programmer. One could perhaps incorporate certain rewards or punishments if the agent reaches any of these points.

Though this second mechanism has interesting features and can be developed even further, it nevertheless has its shortcomings in representing virtue ethics. In classical virtue ethics, a virtue is really a virtue if it allows the higher rational part of a human being to control the lower appetitive part which can sometimes *resist* this control. Aristotle and Aquinas compare the power of the rational part over the lower appetitive part to the power of a monarch over his subjects.[8] Though subjects generally obey their monarch, they still have minds of their own and can choose to rebel against authority. In a sense, the lower appetitive part of a human being also has a mind of its own and can go against the dictates of reason.

To simulate this, we require at least two components in an agent: a rational component and an appetitive component. With regards to the appetitive component, Aquinas talks about "passions" that propel or incline us towards certain objects.[9] These passions are roughly the same as emotions in modern terms, and therefore the appetitive component can be interpreted as an emotional component. But this appetitive/emotional component is not disembodied. For Aquinas, emotions are also bodily phenomena that are grounded in physical needs (e.g. being grumpy because of hunger) and they can have physical manifestations (e.g. the rapid beating of the heart when in love, the blushing of the face when ashamed). Therefore, a physical component is required as well.

Virtue ethics also recognizes that virtues are learned from others. Aristotle, for example, recognizes the importance of childhood upbringing in the development of virtue. "It is not unimportant, then, to acquire one sort of habit or another, right from our youth."[10] Social influence in the form of praise and criticism for specific behaviors can determine which virtues or vices flourish in a particular group. There is an important social component in how virtues work.

## 3 The complex way

There is in fact a computer simulation framework that recognizes the importance of these various components. The PECS reference model was proposed by Bernd Schmidt and Christoph Urban for use in social simulations.[11] PECS stands for "physical conditions, emotional state, cognitive capabilities, and social status."[12] Schmidt and Urban insisted that to more accurately model human behavior, all four components should be in place. "It is a fundamental conviction of the PECS research program that an understanding of human behaviour can be achieved only if all 4 aspects and their interaction are taken into account."[13] The reference model was proposed as an alternative to the more popular BDI (Belief, Desire, Intention) model.[14] The BDI model conceives of agents as intending/executing plans according to specific beliefs in order to fulfill certain desires or goals. Though the BDI model has its merits, it unfortunately does not include any emotional or social components, and neither does it have any mechanism for agent learning or agent-to-agent communication. Despite these shortcomings, the BDI model is popular and has many implementations, while the PECS model has never gone beyond the conceptual stage.[15]

More recently, Joshua Epstein's work on *Agent_Zero* promotes a similar idea to PECS by including cognitive, emotional, and social modules in agents.[16] Epstein's design of these modules is grounded in certain findings of neuroscience. He presents a functioning computer simulation that captures irrational behaviors such as mass panic and contagious violence. Though the theoretical assumptions for *Agent_Zero* are different from those of virtue ethics, Epstein's work shows the potential of using interacting components in agents to simulate more realistic human behavior.

[8] Aquinas, *Summa*, I, q. 81, a. 3, r. 2.
[9] Aquinas, *Summa*, II-I, q. 22.
[10] Aristotle, *Nicomachean Ethics*, 1103b24.
[11] Schmidt, *Modelling*; Urban, "PECS."
[12] Schmidt, *Modelling*, 1.
[13] Schmidt, *Modelling*, 20.
[14] Rao and Georgeff, "BDI-Agents."
[15] Balke and Gilbert, "How Do Agents."
[16] Epstein, *Agent_Zero*; Epstein and Chelen, "Advancing Agent_Zero."

## 3.1 Simulating the virtue of temperance

There are many ways to implement PECS in a simulation of virtue ethics. I present only a simple example of how this could work. I will focus on only one virtue, the virtue of temperance. Temperance is the virtue of moderation in matters of food, drink, and sex.[17] According to a long philosophical tradition, it is one of the four cardinal virtues along with prudence, justice, and courage. The word "temperance" is no longer commonly used. Perhaps the more suitable modern term would be "self-control," as long as it is qualified as the self-control that controls natural physical desires and not other kinds of self-control such as being able to control anger.

The assumption behind temperance is that there are things in the world that draw us that are normally good and beneficial. At the same time, these things can be desired excessively. In the case of food, one obviously needs food to survive and most foods are healthy in moderation, but past a certain point, food can be detrimental to one's health. For example, oily and fatty foods are dangerous to someone already struggling with heart problems. Yet even when one *knows* that certain foods are detrimental to one's health, one can stubbornly persist in eating such foods because the craving is too strong. The person who resists the craving but in the end gives in to it is called incontinent, while the person who deliberately pursues the craving without any restraint is called intemperate.[18]

In addition to damage to self, there can also be damage to others. For example, excessive drinking of alcohol can result in abusive behavior towards others; a married person who is sexually unfaithful can cause serious emotional injury to the spouse; and when there is a scarcity of food, eating more than one's fair share can cause others to starve. Such negative behaviors against others can provoke negative responses from those affected or from the larger social group. For example, a hunter-gatherer tribe may ostracize or expel a selfish member who regularly eats more than his or her due share of the food. In this light, we also see a social facet to the virtue of temperance.

## 3.2 Sugarscape

To simulate temperance, I have chosen to modify a part of the computer simulation called *Sugarscape*.[19] I have chosen *Sugarscape* because it is already well-known in the agent-based modeling community and because it clearly shows how one can build complex simulations starting from basic mechanisms.

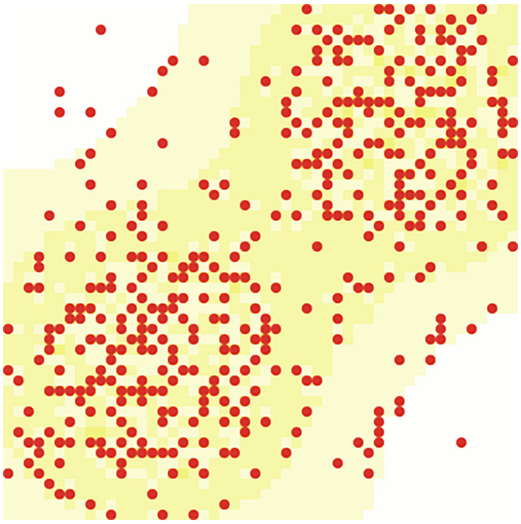

**Figure 2.** A *Sugarscape* implementation from the NetLogo Models Library.

[17] Aristotle, *Nichomachean Ethics*, 1118a25-35; Aquinas, *Summa*, II-II, q. 141, a. 4.

[18] Aristotle, *Nichomachean Ethics*, 1148a15-23; 1119a22-35.

[19] Epstein and Axtell, *Growing Artificial Societies*.

The basic premises of *Sugarscape* are as follows:

1. There are agents and sugar in a large grid world. In Figure 2, the red spots represent the agents and the yellow patches represent the sugar.
2. Agents start with a certain amount of sugar.
3. The agents metabolize at every turn, i.e. they consume a certain amount of sugar in their possession at every turn (their metabolic rate, the amount of sugar consumed, is variable).
4. If an agent does not have any sugar left to consume, it dies.
5. In order to survive, an agent moves around the grid world to collect sugar.
6. The sugar patches can have different amounts of sugar (variable). In Figure 2, the dark yellow patches contain more sugar than the light yellow patches.
7. The agents' choice of which sugar patch to go to is restricted by their range of vision (variable), e.g. one agent might see only up to one square around itself, while another might see up to three squares around itself.
8. After the sugar has been consumed in a sugar patch, the sugar can grow back later according to a predetermined regrowth rate (variable).

The above list constitutes the most basic stage of *Sugarscape*. Epstein and Axtell build over this basic stage to add complex features such as mating and the bearing of offspring, the evolution of genetic traits, trading of resources, conflict between different groups, and transmission of diseases. However, this basic stage of *Sugarscape* is sufficient for our purposes. We merely require a simulation where agents pursue and consume food in order to survive.

## 3.3 Modifying Sugarscape

We modify this basic stage of *Sugarscape* to develop a simulation of temperance. First, we need to introduce the possibility of unhealthy and excessive consumption. An agent may consume a large amount of food (we prefer to use food instead of the fanciful sugar) which is detrimental to the agent's health. Since the agents in *Sugarscape* collect different amounts of sugar but have a fixed metabolism rate, i.e. the rate at which they consume the sugar, excessive consumption is not possible. An agent in *Sugarscape* consumes a constant amount of sugar at each time step regardless of how much sugar it collects. A modification is to introduce a new variable for agents, *health*, and then to collapse their collecting and consuming actions into one. So instead of agents possessing sugar which they consume at every time step, agents will have a health value which decreases at every time step; instead of agents collecting all the sugar in sugar patches to add to their inventory of sugar, agents will consume all the food at food patches to increase their health. In this way, we can have uneven levels of consumption based on the amount of food in a food patch.[20] This is also a reasonable modification. It is more plausible to imagine agents with a certain level of health than to imagine agents carrying their entire store of sugar/food everywhere they go. It is also reasonable to imagine agents consuming food as they encounter it rather than being able to keep their food indefinitely (food is perishable more often than not; perhaps sugar was an apt choice for *Sugarscape* because of sugar's preservability).

After introducing a health variable for agents and the direct consumption of food in food patches, we can determine the possible amounts of food. For the sake of simplicity, we will permit only 1 to 3 units of food in food patches. 1 unit of food is good for an agent (it adds 1 unit of health), 2 units of food is even better (it adds 2 units of health), while 3 units of food is unhealthy (instead of adding health it deducts 1 unit of health). 3 units of food is, therefore, excessive for agents.

In addition to this, we also introduce excess according to social criteria. Let us suppose that there is a standing social rule that dictates that it is wrong to eat more than 1 unit of food at a time. We may imagine

[20] The downside of this is that agents will no longer have a wealth of sugar which is important for the more advanced features of *Sugarscape* such as trading and economic calculations. However, we do not need any of these features in our computer simulation of temperance.

that this rule originated from a distant past when the community experienced scarcity but is now held as a custom even without scarcity. When an agent eats 2 or 3 units of food, it is punished by any other agent that sees it (based on the range of vision). If there is no other agent nearby, then the agent can eat 2 or 3 units of food without punishment. This social punishment introduces a dilemma concerning 2 units of food: 2 units of food is beneficial for an agent but is at the same time condemned and punished by the community. There is now a tension between individual and social interests. However, there is no dilemma with respect to 3 units of food; an agent will still suffer physical detriment from eating 3 units of food whether or not it is punished by the community. Given all these modifications, we can introduce five cognitive rules that the agent can learn through experience:

1. 1 unit of food is good for me.
2. 2 units of food is very good for me.
3. 2 units of food is bad for the community.
4. 3 units of food is bad for me.
5. 3 units of food is bad for the community.

These rules will be placed in the cognitive component of the agent and will be weighted based on the agent's individual experience. We will explain this further below when we discuss the simple PECS framework. For now, it is enough to know that whenever an agent sees a food patch inside its range of vision, it does not automatically pursue the food in that food patch but first undergoes a decision-making process that involves the four PECS components. The decision-making process will determine whether an agent pursues or avoids the food at that food patch. In comparison to the basic premises of *Sugarscape*, these are the new basic premises for our simulation:

1. There are agents and food in a large grid world.
2. Agents start with a certain amount of health (variable).
3. The agents metabolize at every turn, i.e. they lose a certain amount of health at every turn (variable).
4. If an agent does not have any health left, it dies.
5. In order to survive, an agent moves around the grid world to consume food.
6. The food in food patches can be 1, 2, or 3 units of food.
7. The agents' choice of which food patch to go to is restricted by their range of vision (variable).
8. Before an agent goes to a certain food patch, it initiates a decision-making process that involves the four PECS components. This decision-making process will determine whether the agent will pursue or ignore the food.
9. After the sugar has been consumed in a sugar patch, the sugar can grow back later according to a predetermined regrowth rate (variable).

## 3.4 A simple PECS framework

The PECS framework is what will determine whether an agent pursues the food at a certain food patch or not. Since this simulation is for illustration purposes, it is better to keep the PECS framework as simple as possible. The primary intent is to show how different components in an agent can interact and possibly conflict with each other and how these components can be configured to reflect virtue.

The decision-making process is a function that takes the agent and the food as its arguments. It starts with 0 scores for all the four components. These scores are changed as the decision-making process moves forward.

```
def decision(agent, food):
        physicalScore = 0
        emotionalScore = 0
        cognitiveScore = 0
        socialScore = 0
```

We begin with the physical component in the decision-making process. The physical component is based on the agent's health. If the agent has health equal to or greater than 10, then the agent is satisfied. However, below 10, it begins to get hungry. The lower the agent's health, the greater its hunger. We can think of the physical component as representing the agent's primary survival instinct.

```
if (agent.health >= 10):
    physicalScore = 0
elif (agent.health < 10 and agent.health > 6):
    physicalScore = 1
elif (agent.health < 7 and agent.health > 3):
    physicalScore = 2
elif (agent.health < 4):
    physicalScore = 3
```

If the physical component is based on the agent's health, the emotional component is based on the object (i.e. the food). Aquinas' theory of emotion is an appraisal theory of emotion. According to him, emotions are based on the perceived goodness or badness of an object. In our case, the greater the amount of food, the more desirable it seems and the stronger the emotion of desire is. By default, 3 units of food will elicit a greater emotion of desire (3 times as much) than 1 unit of food. However, negative experiences with 3 units of food may eventually change this. The desire for 3 units of food is reduced by the number of times the agent was made sick from it. The process of losing desire is incremental; it will take 3 occasions of illness before the agent completely loses its emotional desire for 3 units of food.

```
if (food.amount == 1):
    emotionalScore = 1
elif (food.amount == 2):
    emotionalScore = 2
elif (food.amount == 3):
    emotionalScore = 3 - agent.timesSick3
```

The cognitive component is based on the five cognitive rules that we mentioned above. Each rule is given a weight according to how many times the agent learns the rule through experience. For example, every time the agent eats 2 unit of food, it learns rule2 ("2 units of food is very good for me") and rule2weight is increased. Every time the agent is punished for eating 2 units of food, it learns rule3 ("2 units of food is bad for the community"). The positive and negative rule weights contribute to the final cognitive score. This represents a very simple mechanism for learning rules through experience.

```
if (food.amount == 1 and agent.rules[“rule1”]):
    cognitiveScore = agent.rules[“rule1weight”]
if (food.amount == 2 and agent.rules[“rule2”]):
    cognitiveScore = agent.rules[“rule2weight”]
if (food.amount == 2 and agent.rules[“rule3”]):
    cognitiveScore -= agent.rules[“rule3weight”]
if (food.amount == 3 and agent.rules[“rule4”]):
    cognitiveScore -= agent.rules[“rule4weight”]
if (food.amount == 3 and agent.rules[“rule5”]):
    cognitiveScore -= agent.rules [“rule5weight”]
```

The score of the social component is based on social approval or condemnation multiplied by a social pressure value. When an agent consumes 1 unit of food in the sight of others, it is not punished and the social score is 1. We can interpret this as a subdued form of social approbation.[21] However, the agent is

[21] It is also possible to implement positive benefits or rewards for adhering to social expectations. However, for the sake of simplicity, we do not include any.

punished when others see it consume 2 or 3 units of food. The number of times the agent is punished is recorded as timesPunished2 and timesPunished3; either of these variables brings the social score down to a negative value. For example, if the agent has been punished two times in the past for eating 2 units of food, then the social score for 2 units of food will be -2.

Afterward, this score is multiplied (or one could say, amplified) by a social pressure value. Social pressure represents the number of times the agent has socially interacted with other agents in the past. In our context, an interaction occurs simply when an agent consumes food in the presence of another agent. For every interaction, the social pressure value increases by a small rate (the rate is ideally smaller than 1, for example, 0.2). This mechanism simulates the gradual social training of the agent. The more interactions the agent has with other agents, the more socially conscious it becomes and the more it pays attention to what others say.

```
if (food.amount == 1):
    socialScore = 1
if (food.amount == 2):
    socialScore -= agent.timesPunished2
if (food.amount == 3):
    socialScore -= agent.timesPunished3
socialScore = socialScore * agent.socialPressure
```

Finally, we add the four scores to obtain the decisionScore. If the decisionScore is more than 0, then the agent will pursue the food at the food patch (if there are several food patches inside the agent's range of vision with positive decisionScores, the agent will choose one of them at random). If the decisionScore is equal to or less than 0, the agent will ignore the food.

```
decisionScore = physicalScore + emotionalScore + cognitiveScore + socialScore
```

With these four components in place in the decision-making process, we can now consider how agents act in a simulation.

## 3.5 Agents in action

We start with the most straightforward case, an agent considering 1 unit of food. Assuming the agent is at full health of 10, then the physical score is 0. Based on this amount of food, the emotional score is 1. Assuming this is the agent's first time to encounter 1 unit of food, then the cognitive score is 0. Finally, since the agent has not previously interacted with other agents and has no social pressure yet, the social score is 1 * 0 = 0. The total decision score is 1, which makes the agent pursue the food.

After several steps and experiences, the agent's decision-making process for 1 unit of food becomes different. During a new deliberation, let us suppose that the agent's health is 5 and so its physical score is 2. Since it is still 1 unit of food being considered, the emotional score remains 1. The agent has by this time consumed 1 unit of food several times so it knows rule1 well ("consuming 1 unit of food is good for me"). Assuming it has consumed 1 unit of food five times in the past, then rule1weight is significantly higher at 5 and the cognitive score is 5. Because it has by now interacted many times with other agents, its social score is higher. Assuming it has previously interacted ten times with other agents, then the social pressure value is 0.2 * 10 = 2. This social pressure value is multiplied with the preliminary social score, 1 * 2 = 2. The final decision score is 2 + 1 + 5 + 2 = 10, which is much greater than the decision score of 1 in the first case. We could interpret this to mean that, over time, the agent has become much more inclined towards consuming 1 unit of food. Whereas it pursued the food weakly in the beginning (with decision score 1), it now pursues it very strongly (decision score 10).[22]

[22] Since there are no disadvantages to eating 1 unit of food, this difference in scores might not seem significant. An agent will always pursue 1 unit of food regardless of the decision score since the decision score for 1 unit of food can never be negative in this simulation. However, as soon as we introduce any disadvantage (for example, if consuming 1 unit of food is suddenly socially punished), then the intensity of the inclination could make a difference.

Next, we have an agent considering 2 units of food. The beginning is not so different from the case of 1 unit of food. The only difference is that instead of an emotional score of 1, the emotional score is 2, which results in a total score of 2. It is only after several steps that the scores become interesting. Let us suppose that the agent's health is now 5, therefore the physical score is 2. Based on the amount of food, the emotional score is still 2. The cognitive score depends on how many times the agent has consumed 2 units of food in the past (which involves learning rule2) minus the number of times the agent has been punished by others for eating 2 units of food (rule3). If we assume that the agent has eaten 2 units of food three times in the past and has consistently been punished for doing so, then the cognitive score is 3 – 3 = 0. Finally, the social score is calculated by the number of times the agent was punished for 2 units of food (as just mentioned, 3 times, resulting in a negative value of -3) multiplied by a social pressure value. If we assume ten previous interactions at a social pressure rate of 0.2, then the social pressure value is 10 * 0.2 = 2. As a result, the social score is -3 * 2 = -6. The four component scores add up as 2 + 2 + 0 – 6 = -2. Since the total decision score is negative, the agent decides to ignore the 2 units of food.

This could be different. If the agent has less health (for example, 2 health) and for this reason is hungrier, then the physical score would be higher at 3. The emotional score remains 2. If the agent was punished two times in the past for eating 2 units of food but had one occasion when it was not punished (there were no other agents in the vicinity), then its cognitive score would be positive: rule2weight – rule3weight = 3 – 2 = 1. Finally, if the agent had fewer interactions with other agents in the past (for example, five interactions instead of ten), then its social pressure value would be weaker: 5 * 0.2 = 1. This social pressure value is multiplied by the number of times the agent has been punished for 2 units of food, resulting in the social score 1 * -2 = -2. The total decision score would be 3 + 2 + 1 – 2 = 4, which makes the agent pursue the 2 units of food.

What is important to note in these last two examples is the tension between components. The social component is negative while the other components are either positive or zero. In this way, the agent's social history can compel the agent to ignore 2 units of food. This depicts how certain "virtuous" behaviors are socially constructed and enforced. There is nothing in the natural constitution of the agent that makes it ignore 2 units of food. On the contrary, 2 units of food gives the maximum health benefit an agent can have in this simulation. The prohibition is therefore purely social. The internal conflict in the scores represents a conflict between individual advantage and a social norm.

Last but not least, we have an agent considering 3 units of food. The decision score from the first encounter is again similar to the previous examples, but this time with an emotional score of 3 and a total score of 3. The agent consumes the food. Let us imagine that in this first consumption, the agent becomes sick and is also punished by other agents. By the time the agent considers eating 3 units of food a second time, its health is lower, for example, 3. In this case, its physical score is 3 because of stronger hunger, its emotional score is 2 (3 – number of times sick = 3 – 1), its cognitive score is -2 (due to both rule4 and rule5), and its social score is -1 (let us assume the social pressure value is 1, so the social score is 1 * -1). The total decision score is 3 + 2 – 2 – 1 = 2. Despite having experienced sickness and punishment from 3 units of food, the agent decides to eat it again. In a simple way, this represents how physical conditions and/or emotions can lead one to irrational behavior, including pursuing something that is already known to be harmful and/or disapproved by society. Consider, for example, how some people continue to smoke even when they have serious lung problems and doctors and family members are urging them to quit.

In this last case, we observe a more pronounced internal conflict. Both the physical and emotional scores are positive while the cognitive and social scores are negative. The positive scores outweigh the negative scores so that the agent consumes 3 units of food a second time. This causes it to become sick again and, let us imagine, to be punished by others again. By the time it considers eating 3 units of food a third time, its physical score is 3 (strong hunger again), emotional score is 1 (3 – number of times sick = 3 – 2), cognitive score is -4 (due to rule4 and rule5 learned a second time), and social score is -2 (the agent was punished twice already). The decision score is 3 + 1 – 4 – 2 = -2, therefore the agent chooses to ignore the food. At this point, the agent has learned its lesson from previous experiences. We can also say that, with respect to 3 units of food, it has learned the virtue of temperance. In the midst of an internal struggle, it is able to say "no" to a desirable but unhealthy object.

The most important thing to note in this complex way of simulation is that the virtue of temperance is not a single variable. It manifests itself through the interaction of the four components in an agent. The complex way of simulating virtue ethics therefore captures something that the simple ways do not: the internal struggle often involved in virtuous behavior.

## 3.6 Some results

We can see the development of virtuous behavior in the course of running an actual computer simulation for many steps.[23] For settings, we have a 5 × 5 grid world with five agents and ten random food patches. The agent vision is 2, agent metabolism is 0.3, and the food regrowth rate is 1. We run the simulation automatically for fifty steps (this produces a manageable amount of data; the simulation can, of course, run for hundreds of steps). The following chart shows some results:

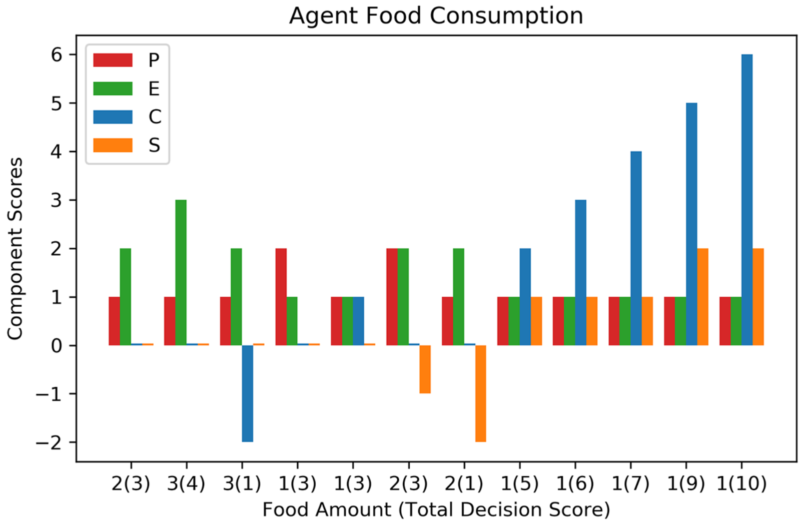

**Figure 3.** The complete food consumption history of an agent with respective scores.

In Figure 3, we have a complete history of the food consumption of one particular agent. Each instance of consumption has four bars which represent the scores of the four components (physical, emotional, cognitive, and social). These are the scores given to the food before it is consumed; the scores are measured on the y-axis. The labels on the x-axis show the amount of food that is consumed and, within parentheses, the total decision score. For example, in the first instance, we can see that the amount of food consumed is 2 units and the total decision score is 3. The four bars of the first instance also reveal that it has a physical score of 1, an emotional score of 2, a cognitive score of 0, and a social score of 0.

Early on, the agent consumes both 2 and 3 units of food. However, the agent immediately stops eating 3 units of food after the second time. It refuses to eat 3 units of food a third time because its decision score for 3 units of food has become negative. This is not surprising given the double penalty for eating 3 units of food.[24] We also see that the agent consumes 2 units of food only three times (the first, sixth, and seventh instances). In this case, the social score goes down in the sixth and seventh instances, representing the role

[23] The file temperance_automatic.py at https://github.com/JeremiahLR/Temperance runs the computer simulation automatically for a designated number of steps with the settings used here and produces the same kind of charts. However, given the random world generation, it will not produce the exact same results.

[24] A closer analysis of the second and third instances of consumption shows that on both occasions the agent gets sick and is punished by others. This is evident, for example, in how the cognitive score drops to -2 in the second attempt, meaning that both rule4 (“3 units of food is bad for me”) and rule5 (“3 units of food is bad for the community”), which are negative rules, are applied. That the social score during these two instances is 0 is due to the slow rate of social pressure (0.2) and the way the social score gets rounded to the nearest integer.

that social influence plays in the agent's decision-making process.[25] Similar to the 3 units of food, the agent refuses to continue eating 2 units of food because the decision score has become negative. From the eighth instance onward, the agent only eats 1 unit of food with gradually increasing decision scores. Since it has learned to ignore 2 and 3 units of food and exclusively concentrates on 1 unit of food, we can say that the agent has learned the virtue of temperance.

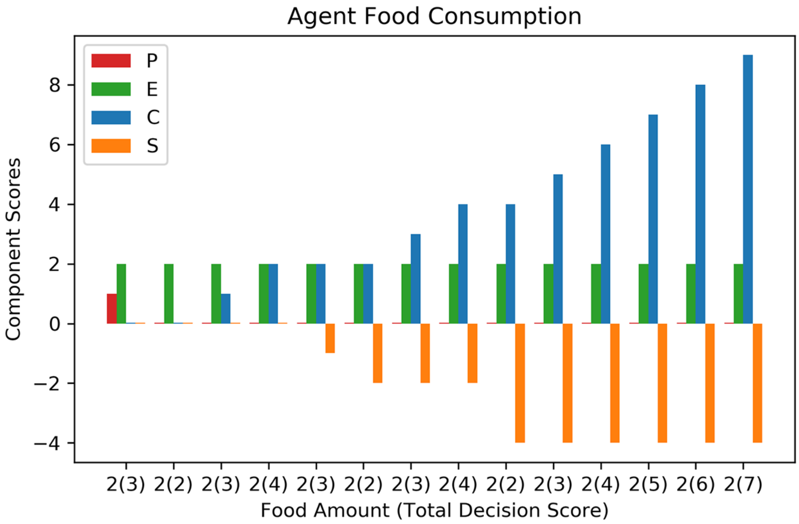

**Figure 4.** The food consumption history of an agent for 2 units of food with respective scores.

We run the simulation again but this time with a larger 10 × 10 grid world instead of 5 × 5. This way, agents can have enough space to not encounter each other all the time. Figure 4 above shows the food consumption history of a particular agent for 2 units of food (we exclude 1 and 3 units of food to make the data fit in the chart). As I mentioned previously, 2 units of food is interesting because it involves a dilemma: 2 units of food is beneficial for the agent but is condemned by the community. In this case, the agent is able to consume 2 units of food for a few times in the beginning without being punished because there are no other agents nearby.[26] By the time the agent starts experiencing social disapproval for eating 2 units of food (the social score goes down starting from the fifth instance), the agent is already well-accustomed to eating 2 units of food; it ignores the social disapproval and continues eating 2 units of food.[27] This case captures how early training or upbringing can make a significant difference in how one acts in the long term. The early missed opportunities to be socially corrected set the agent on a stubborn behavioral trajectory that is difficult to change.

It also leads us to the question, is the agent virtuous or not? It continues consuming 2 units of food because it knows with growing certainty that 2 units of food is beneficial for its health. Society, however, says that this action is wrong. How much should virtue be based on individual flourishing and how much on social norms? What takes precedence when these two factors collide? This is an interesting question in virtue ethics that we will not try to answer here.[28] But the fact that this tension can be drawn out in our computer simulation through the behavior of agents is a promising sign. What other interesting aspects of virtue ethics can be captured using computer simulations?

[25] The cognitive score is 0 in both instances because rule2 ("2 units of food is very good for me"), which is positive, and rule3 ("2 units of food is bad for the community"), which is negative, are applied and cancel each other out.

[26] This is evident in how the cognitive scores in the third and fourth instances of consumption increase. Rule2 ("2 units of food is very good for me"), which is positive, is applied, while rule3 ("2 units of food is bad for the community"), which is negative and which normally cancels rule2 out, is not applied.

[27] In theory, since the social score involves a multiplier (the social pressure value) and the cognitive score only increases incrementally, the social score can still catch up at some point in the future. However, this would take a long time and is not guaranteed to happen.

[28] This question is sometimes asked concerning the difference between Western and Asian conceptions of virtue given the Western stress on individualism and the Asian prioritization of the group. For example, see Reyes, "Loób and Kapwa."

## 4 Conclusion

I have presented a simple way and a complex way to simulate virtue ethics. The simple way represents virtue as a single variable while the complex way represents virtue as the result of interacting components in an agent. The simple way is easier to implement and could still be useful in social simulations that only partially incorporate virtue ethics. It is possible to develop the second simple way involving mathematical event-functions even further for flexible representations of virtue. However, for a more accurate rendering of virtue ethics theory, the complex way is preferable. It is able to represent the inner conflict and struggle that sometimes occurs in the practice of virtue. The four PECS components used in the simulation of temperance are quite simple, but one can certainly develop more sophisticated PECS components.[29] The simulation of temperance presented here is only meant to serve as a proof of concept. It shows the feasibility and advantages of simulating virtue through a PECS framework. In the process, I hope that this article encourages explorations in a new field of computational virtue ethics.[30]

## References

Aquinas, Thomas. *Summa Theologiae*. Edited by Commissio Leonina. Vol. 4-12, *Opera omnia iussu impensaque Leonis XIII P.M. edita*. Rome: Typographia Polyglotta S.C. de Propaganda Fide, 1888-1906.

Aquinas, Thomas. *The "Summa Theologica" of St. Thomas Aquinas*. Translated by Fathers of the English Dominican Province. 2nd and Revised ed. London: Burns Oates and Washbourne, 1920.

Aristotle. *Nicomachean Ethics*. Translated by Terence Irwin. 2nd ed. Indianapolis, IN: Hackett, 1999.

Axelrod, Robert. *The Evolution of Cooperation*. New York: Basic Books, 1984.

Balke, Tina, and Nigel Gilbert. "How Do Agents Make Decisions? A Survey." *Journal of Artificial Societies and Social Simulation* 17:4:13 (2014). doi: 10.18564/jasss.2687.

Danielson, Peter. *Artificial Morality: Virtuous Robots for Virtual Games*. London: Routledge, 1992.

Epstein, Joshua. *Agent_Zero: Toward Neurocognitive Foundations for Generative Social Science*. Princeton, NJ: Princeton University Press, 2013.

Epstein, Joshua, and Robert Axtell. *Growing Artificial Societies: Social Science from the Bottom Up*. Washington D.C.: Brookings Institution Press, 1996.

Epstein, Joshua, and Julia Chelen. "Advancing Agent_Zero." In *Complexity and Evolution: Toward a New Synthesis for Economics*, edited by David S. Wilson and Alan Kirman, 299-318. Cambridge, MA: MIT Press, 2016.

Gauthier, David. *Morals by Agreement*. Oxford: Oxford University Press, 1986.

Hegselmann, Rainer, and Oliver Will. "Modelling Hume's Moral and Political Theory—The Design of HUME$_{1.0}$." In *Norms and Values: The Role of Social Norms as Instruments of Value Realisation*, edited by Michael Baurmann, Geoffrey Brennan, Robert E. Goodin, and Nicholas Southwood, 205-232. Baden-Baden: Nomos, 2010.

Hegselmann, Rainer. "Moral Dynamics." In *Encyclopedia of Complexity and Systems Science 2009*, edited by Robert Meyers, 5677-5692. New York: Springer, 2009.

Laird, John. *The Soar Cognitive Architecture*. Cambridge, MA: MIT Press, 2012.

MacIntyre, Alasdair. *After Virtue: A Study in Moral Theory*. 3rd ed. Notre Dame: University of Notre Dame, [1981] 2007.

Mascaro, Steven. "Abortion, Rape and Suicide: Evolutionary ALife Investigations of Ethically Contentious Behaviour." Doctor of Philosophy, Clayton School of Information Technology, Monash University, 2008.

Mascaro, Steven, Kevin Korb, and Ann Nicholson. "Suicide as an Evolutionarily Stable Strategy." In *Advances in Artificial Life*, edited by Josef Kelemen and Petr Sosík, 120-132. Berlin: Springer-Verlag, 2001.

Mascaro, Steven, Kevin Korb, and Ann Nicholson. "ALife Investigation of Parental Investment in Reproductive Strategies." In *Artificial Life VIII: Proceedings of the Eighth International Conference on Artificial Life*, edited by Russell Standish, Mark A. Bedau, and Hussein A. Abbass, 358-361. Cambridge: MIT Press, 2003.

Mascaro, Steven, Kevin Korb, Ann Nicholson, and Owen Woodberry. *Evolving Ethics: The New Science of Good and Evil*. Exeter: Imprint Academic, 2010.

[29] It is theoretically possible to use more complex cognitive architectures to simulate PECS and the decision-making process. For example, the SOAR cognitive architecture which is used in both games and military applications has physical (perceptual), emotional, and cognitive components and could potentially include a social component as well. See Laird, *The SOAR Cognitive Architecture*.

[30] This article was made possible through a Research Fellowship at the Institute of Advanced Study for Media Cultures of Computer Simulation (MECS) at the Leuphana Universität Lüneburg, funded by the German Research Foundation (DFG).

Rao, Anand, and Michael Georgeff. “BDI-Agents: From Theory to Practice.” In *Proceedings of the First International Conference on Multiagent Systems, June 12-14, 1995*, 312-319. San Francisco: AAAI Press, 1995.

Reyes, Jeremiah. “Loób and Kapwa: An Introduction to a Filipino Virtue Ethics.” *Asian Philosophy: An International Journal of the Philosophical Traditions of the East* 25:2 (2015), 148-171.

Schelling, Thomas. “Models of Segregation.” *The American Economic Review* 59:2 (1969), 488-493.

Schelling, Thomas. *Micromotives and Macrobehavior*. New York: W. W. Norton, 1978.

Schmidt, Bernd. *The Modelling of Human Behaviour*. Ghent, Belgium: SCS-Europe BVBA, 2000.

Urban, Christoph. “PECS: A Reference Model for the Simulation of Multi-Agent Systems.” In *Tools and Techniques for Social Science Simulation*, edited by Ramzi Suleiman, Klaus Troitzsch and Nigel Gilbert, 83-114. Heidelberg: Physica-Verlag, 2000.

Wiegel, Vincent. “SophoLab: Experimental Computational Philosophy.” Doctor of Philosophy, Technische Universiteit Delft, 2007.

Will, Oliver. “Hume$_{1.0}$ - An Agent-Based Model on the Evolution of Trust in Strangers and Division of Labour.” In *Multi-Agent-Based Simulation X*, edited by Gennaro Di Tosto and H. Van Dyke Parunak, 123-134. Berlin: Spring-Verlag, 2010.