Proceedings of the 2007 Winter Simulation Conference
S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, eds.

AN EXPLORATION-BASED TAXONOMY FOR EMERGENT BEHAVIOR ANALYSIS IN SIMULATIONS

Ross Gore
Paul F. Reynolds, Jr.

Modeling and Simulation Technology Research Initiative
151 Engineer's Way, P.O. Box 400740
Department of Computer Science, University of Virginia
Charlottesville, VA 22904-4740, U.S.A.

ABSTRACT

Emergent behaviors in simulations require explanation, so that valid behaviors can be separated from design or coding errors. We present a taxonomy, to be applied to emergent behaviors of unknown validity. Our goal is to facilitate the explanation process. Once a user identifies an emergent behavior as a certain type within our taxonomy, exploration can commence in a manner befitting that type. Exploration based on type supports narrowing of possibilities and suggests exploration methods, thus facilitating the exploration process. Ideally, a taxonomy would be robust, allowing reasonable variation in behavior type assignment without penalty in cost or correctness during the exploration process. The taxonomy we present is robust, comprehensive and suitable for use with our established emergent behavior exploration methods. In addition to the taxonomy, we present our design rationale, and a summary of results from a test application of our taxonomy.

1 INTRODUCTION

Understanding emergent behavior poses an interesting challenge. Emergence can represent a valid behavior arising from seemingly unrelated phenomena, or it can reflect an error in a model or its implementation (Davis 2005). Behavior is emergent if it is unexpected and stems from the interactions of the underlying components of the model (Johnson 2006). Emergent behavior can be beneficial, for example, if the unexpected behavior allows users to adapt the model to support tasks the designer never intended. Emergent behavior can be problematic if it reflects an error in the design or implementation of a model.

Our goal is to characterize an emergent behavior in a manner that supports isolation of its root causes efficiently. The process associated with this goal is typically called validating the emergent behavior, which is different from validating a simulation. Simulation validation is a demonstration that a simulation meets expected behaviors. Emergent behavior validation concerns a demonstration that the unexpected behavior is valid (or not) for a given set of conditions, or experimental frames (Zeigler 2000). Methods to validate simulations exist (Balaci 1997). Emergent behavior validation is still an active area of research. We have proposed "Explanation Exploration" (EE) in previous work for demonstrating that a given emergent behavior is valid (Gore et al. 2007). EE allows a subject matter expert (SME) to test hypotheses about the emergent behavior as a simulated phenomenon is driven towards conditions of interest. The work presented here improves the EE process by identifying a taxonomy that improves the efficiency of testing SME hypotheses that must be considered during emergent behavior validation.

It is important to distinguish a taxonomy that is meant to enhance a process from a taxonomy that is meant to provide a definitive categorization of all observable behaviors. In the case of the former, occasional disagreements among experts about the classification of a behavior should not have significant consequences. The goal of a process taxonomy is to provide a taxonomy that is robust, one that supports the validation process despite occasional disagreements. A robust taxonomy is what we seek, and present. Our intention is to provide a taxonomy that allows us to meet our goal: characterizing emergent behaviors in a manner that supports isolation of its root causes efficiently. We have designed this taxonomy to work well with EE.

In the remainder of this paper we describe previous work for classifying emergent behavior and describing the different types of unexpected behaviors that can arise in the laboratory setting. Then we present our taxonomy, describe the exercise we performed to test its utility, and discuss future work.

2 PREVIOUS WORK

2.1 Forensic Engineering

Forensic Engineering is the application of engineering principles and methodologies to answer questions of fact associated with catastrophic events. Forensic engineers do not treat each catastrophe in the same manner. Instead, they group catastrophes based on the description of the catastrophe with other catastrophes that have similar descriptions. Based on how a catastrophe is identified a certain set of hypotheses associated with the catastrophe category is tested to attempt to determine the cause of the catastrophe (Noon 2001).

2.2 Philosophy of Science

Thomas Kuhn, Imre Lakatos and Lawrence Laudan have considered the problem of unexpected behavior while conducting experimentation. Kuhn was the first to study the matter and prescribes the scientific method to explore unexpected results. Kuhn, Lakatos and Laudan require researchers not to change accepted theories, assumptions or methods to explain unexpected behavior in an experiment. However, Lakatos does not require unexpected behaviors to be investigated at all, as long as empirical progress is continuing to be made. Laundan believed unexpected behavior became worthy of investigation only when the resulting explanation has a certain level of importance. Kuhn, Lakatos and Laundan all differ on how unexpected behaviors which cannot be explained by current theories and methodologies cause new revolutionary shifts in science (Riggs 1992). However, those differences are beyond the scope of our work.

2.3 Previous Taxonomies of Emergent Behavior

Different researchers have attempted to identify different types of emergent behavior in order to understand and classify forms of emergence in a variety of systems. David J. Chalmers presented one of the earliest emergent behavior taxonomies, which distinguishes between *weak* and *strong* emergence (Chalmers 2002). Strong emergence is not deducible even in principle from the laws of the low-level domain, while weak emergence is only unexpected given the properties and principles of the low-level domain.

Mark A. Bedau distinguishes between three kinds of emergence: *nominal*, *weak* and *strong* (Bedau 2002). He uses weak and strong in the same sense as Chalmers. Nominal emergence is the appearance of a macro property in a system that cannot be a micro property. William Seager emphasizes two kinds of emergence: benign and radical. If one can find a descriptive or explanatory scheme, which provides a useful kind of shorthand notation for describing the behavior of a system, the behavior is benign otherwise it is radical (Seager 2006).

Yaneer Bar-Yam (Bar-Yam 2004) distinguishes between four types of emergence: Type 0, Type 1, Type 2 and Type 3. His taxonomy is based on *particles* and *ensembles*. A particle is a single acting agent or entity, while an ensemble is a group of particles. In Type 0 behaviors there are not any interactions between particles and behavior is seen strictly on the particle level. All other types of behaviors involve ensembles. Type 1 behaviors are only unexpected given the properties and principles of the low-level domain while Type 2 behaviors cannot be found in the properties of the system's lower level domains or the interactions of the lower level domains. Type 3 classifies the emergent behavior of systems that arise out of the interaction with the environment.

Fromm proposes a taxonomy that classifies 4 types of emergence based on different feedback types and causality. Type I emergent behaviors are created without feedback, only *feed forward* relationships. The major characteristic of Type II is simple feedback. The feedback is characterized as a top-down relationship from the macroscopic to microscopic level. Multiple feedbacks, learning and adaptation are defining characteristics of Type III emergent behavior. Type IV emergence is characterized by multi-level emergence and a huge of variety of possible states in the created system. Type IV emergence is responsible for structures which cannot be reduced, even in principle, to the direct effect of the properties and laws of elementary components.

3 A TAXONOMY FOR EXPLORING EMERGENT BEHAVIOR

As reflected in the design of our EE work, we seek to validate emergent behaviors employing sets of experimental frames that extend beyond a model's original intended use. EE provides a process for doing so, however it lacks a supporting taxonomy. We find that extant emergent behavior taxonomies do not support the needs of the exploration process. They have been focused on distinguishing the explainable from inexplicable and on the causality and reducibility of emergent behavior. A process-oriented taxonomy should be designed to make the exploration process more efficient, it should be robust, providing positive support even when reasonable alternatives to optimal categorizations are chosen, and it needs to allow for exploring invalid emergent behaviors. The taxonomy we present here is designed to address these needs.

We base our taxonomy on three orthogonal dimensions: reproducibility, predictability and temporality. The orthogonality of our taxonomy is addressed later in this section. Reproducibility concerns the repeatability of a simulation for a given set of inputs. If a simulation is deterministic (perfect reproducibility) then exploration can be narrowed considerably in most cases. A simulation that is not deterministic is stochastic in our taxonomy. No other taxonomy for emergent behaviors considers the reproducibility dimension.

Exploration of emergent behaviors often requires testing of user-created hypotheses, under conditions of interest. Establishing that an emergent behavior is predictable can increase the efficiency of the exploration process. Predictable behaviors enable selective sampling, and constant behaviors enable one-time sampling to test hypotheses under conditions of interest. Others have noted the importance of emergent behavior predictability (Kim 99), but not with a focus on the exploration process.

Consider a photograph of a car partially across a finish line, with other cars behind it. One could describe the scene as “a car winning a race” or alternatively as “the car that won the race.” The two views represent different interpretations of state: process vs. final. The temporality dimension distinguishes between the process of achieving a final state and residing in the final state. It supports different exploration methods for the same depiction of a behavior. No other taxonomies for emergent behavior have considered this dimension.

Our taxonomy consists of these three independent dimensions: reproducibility, predictability and temporality. An emergent behavior should be described as a three-tuple, selecting one attribute from each dimension.

3.1 Reproducibility of Behaviors

An understanding of the behavior’s reproducibility can facilitate identification of efficient methods leading to acceptance or rejection of a hypothesis regarding an emergent behavior. In our taxonomy, a behavior that is reproducible on every occasion for a given simulation and a given input is *deterministic*. We define deterministic behaviors:

deterministic – for fixed input to a given simulation, all observable behaviors of the simulation are unchanging in all respects. Once the input has been specified the observable behaviors of the simulation are determined. Different inputs may lead to different observable behaviors.

If a behavior is not deterministic we regard it in the conventional sense as *stochastic*:

stochastic – for fixed input to the simulation, the observable behaviors of the simulation are not deterministic.

Deterministic behaviors allow for more efficient methods to test hypotheses. A deterministic behavior does not need to be observed for multiple trials with the same input set; instead the behavior only needs to be observed once for each input set of interest. A stochastic behavior requires more computationally expensive hypothesis testing methods. The behavior must be observed for multiple trials for each input set to allow the user to view the different observable behaviors of the simulation that can be produced for the same input set.

Examples help elucidate the difference between deterministic and stochastic emergent behaviors. Consider a simulation that allows agents to interact on a landscape of two commodities: sugar and spice. Agents have variable lifespan, are able to reproduce, accumulate wealth by harvesting sugar and spice on a landscape, and die if they exhaust their supply of either resource. The genetic characteristics, initial wealth, reproductive lifespan, and maximum lifetime of each initial agent on the landscape are based on the outputs of uniform random number generators (Epstein and Axtell 1996).

The population of agents is tracked as time passes in the simulation. Two trials run from time=0 to time=500 with the same input parameters are shown in Figure 1 and Figure 2.

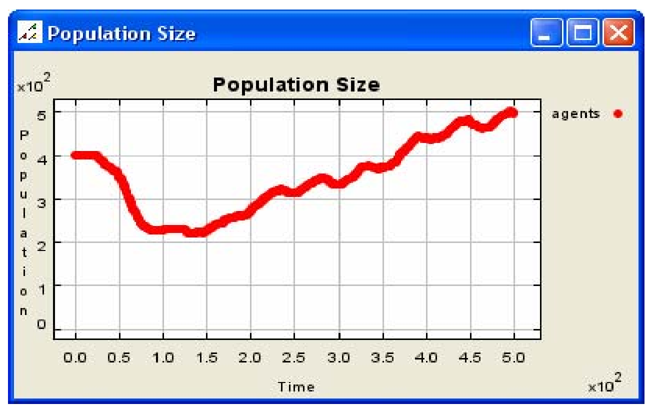

Figure 1: Population Graph for trial 1. At time t=500 population is 497.

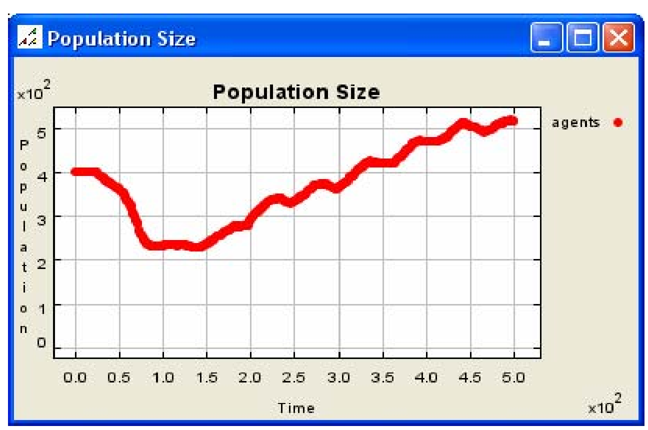

Figure 2: Population Graph for trial 2. At time t=500 population is 517.

Figures 1 and 2 show different observable behaviors for the population of the artificial society for the same fixed input; the behavior is classified as stochastic. Stochastic behaviors require the user to run the simulation with the same input for multiple trials to determine the range of observable behaviors. Running the previously described simulation of an artificial society for 100 trials, we determine that for an initial population of 400, the population at time t=500 can range from 477 to 523. When the behavior is stochastic, running the simulation for multiple trials provides an accurate characterization of the behavior of the simulation for a given input, which cannot be provided by a single simulation run.

However, consider the following simulation of a magnetic pendulum. The magnetic pendulum has a blue magnet as the bob on the end of a rigid rod, which is free to swing on a supporting bar. Two other red magnets are fixed in position on either side of the equilibrium position, in the absence of the red magnets, of the blue magnet. The red magnets are placed along the x-axis, which bisects the two red magnets. The pendulum can rest in equilibrium with the blue magnet directly above either of the fixed red magnets. The setup of the magnetic pendulum is shown in Figure 3 (Duit et al. 1997).

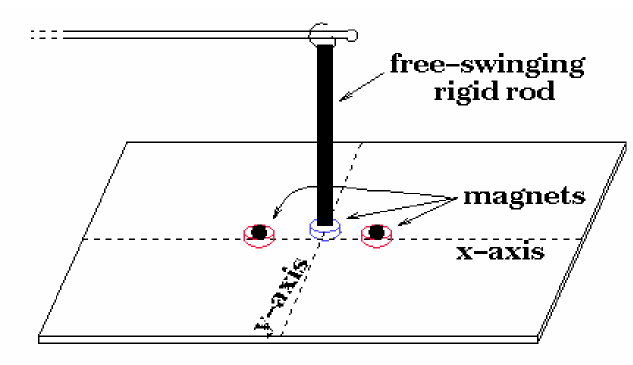

Figure 3: The Magnetic Pendulum Simulation.

The simulation has two possible equilibrium positions: the blue magnet directly above either of the fixed red magnets. The pendulum is pulled to 20 cm above the red magnets and a nonzero distance above or below the x-axis. It will eventually come to rest at one of the two equilibrium positions. Figure 4 shows the path of the blue magnetic pendulum as projected on the plane with the fixed red magnets for the same initial release point for two trials: 20cm above the red magnets and x =-1cm from the x-axis. The white dot represents the position of one fixed red magnet while the blue dot marks the position of the other fixed red magnet. The colored path shows the path that the pendulum follows to its eventual equilibrium position around one of the fixed red magnets (Duit et al. 1997). The final position of the blue magnet is determined by equations often taught in an advanced undergraduate physics course. Since the observable characteristics do not change over multiple trials with the same input the behavior is deterministic. A deterministic behavior gives the user an accurate characterization of the behavior of the simulation for a given input in one trial. Identifying a behavior as deterministic allows a user to perform significantly fewer trials for a given input to test a hypothesis. Performing fewer trials for hypothesis tests improves the efficiency of exploration and thus EE.

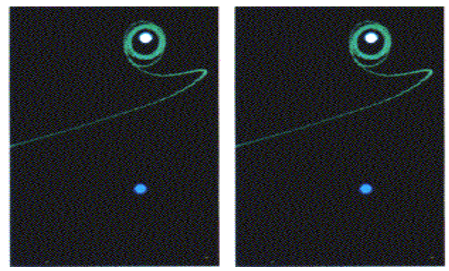

Figure 4: Trials 1 and 2 of the magnetic pendulum path with the same initial position: 20 cm above the red magnets and x= -1.0 cm from the x-axis.

3.2 Predictability

Measuring the predictability of a behavior corresponds to determining the inputs for which the behavior of the simulation cannot be forecasted without running the simulation, and running the simulation for only those inputs. Thus we view predictability as an important dimension in making the exploration process efficient. Predictability is generally relative to the degree of familiarity a user has with a simulation, and so is likely to increase with experimentation. The methods a user chooses to explore an emergent behavior will be largely influenced by the perceived degree of predictability of that behavior. Later in this section we will provide an example of how identifying the predictability of a behavior in a simulation benefits exploration.

A constant emergent behavior does not change the simulation behavior for any simulation input, where the simulation meets the requirements of the conditions of interest. For the simulation meeting the conditions of interest, only a single input needs to be tested to determine the behavior of the simulation for all inputs. However, the emergent behavior still must be explored for understanding and validity. The user explores the emergent behavior by testing new hypotheses under new conditions of interest. The new conditions of interest are created by searching across different model alternatives for the identified abstractions in the model to meet the conditions of interest (Gore et al. 2007). A behavior is constant if:

constant – the observable behaviors of the simulation are the same for every input.

The observable behaviors of most simulations change depending on input. Informally, if the observable behaviors of the simulation can be forecasted for most given inputs without running the simulation, then the behavior is *predictable*. Predictable behaviors allow the user to test hypotheses at critical input points and to extrapolate about the manifestation of the behavior between the tested critical input points. Critical input points still must be tested to determine the observable behaviors of the simulation, but the number of critical input points is much smaller than the number of the inputs that can be forecasted. Predictable behaviors allow for a sparse number of inputs to be tested. Constant behaviors are a subset of predictable behaviors; however it is important in terms of efficiency of exploration to distinguish constant and predictable behaviors. A behavior is predictable if:

predictable – for a sufficient number of given inputs to the simulation the observable behaviors of the simulation can be accurately predicted within a predefined error threshold ε without running the simulation.

Observable behaviors can change significantly with small changes to inputs. This output sensitivity makes observable behaviors very difficult to forecast. A behavior that is not constant or predictable is unpredictable:

unpredictable – for a sufficient number of given inputs to the simulation the observable behaviors of the simulation cannot be accurately predicted within a predefined error threshold ε without running the simulation.

Methods for exploring unpredictable systems exist. These methods are computationally expensive and should be used when a behavior is identified as unpredictable. These methods are not necessary for predictable or constant behaviors.

The following examples demonstrate the difference in sampling the input space of predictable, and unpredictable behaviors. Recall the simulation of agents interacting on a two commodity landscape. Here we consider the population of agents over time with different initial numbers of agents on the landscape. Three trials with different initial numbers of agents are shown in Figures 5-7. All trials run from time=0 time steps to time = 1500 time steps.

The three trials span the initial number of agents from 10 to 550. From the trials it is evident that for initial populations of at least 20 agents the simulation has a carrying capacity of approximately 500 agents. Given any initial agent population of at least 20 agents a population of approximately 500 agents is sustained (Epstein and Axtell 1996). Without running the simulation for 100 agents or 1000 agents we can predict that the population at time = 1500 will be 500 agents +/- 50 agents. Similarly, it is evident that a population of 10 initial agents or fewer will cause the agent population to become extinct. Without running the simulation for 2 or 8 agents we can predict that the population at time t=1500 will be 0 agents.

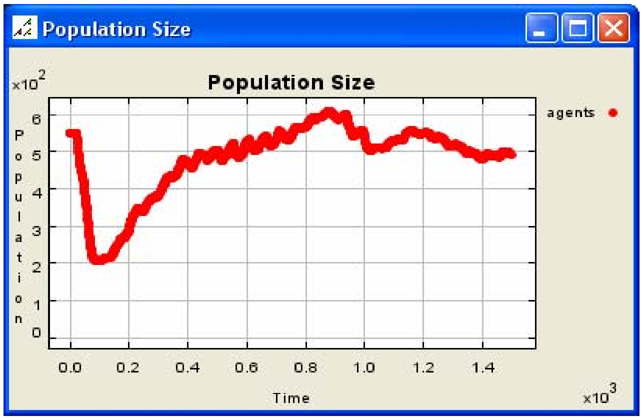

Figure 5: Population Graph for initial number of agents = 550. Number of agents at time = 1500 is 498.

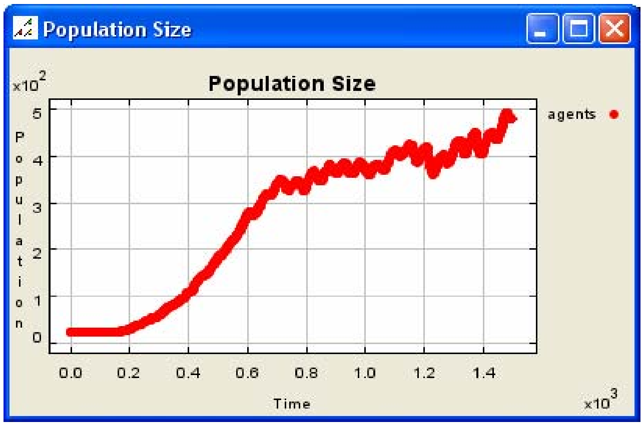

Figure 6: Population Graph for initial number of agents = 20. Number of agents at time = 1500 is 487.

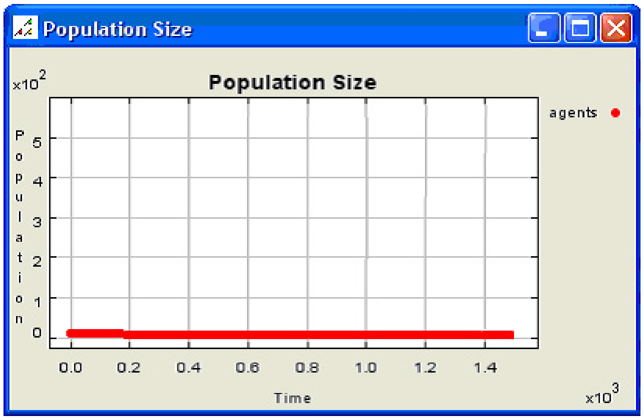

Figure 7: Population Graph for initial number of agents = 10. Number of agents at time = 1500 is 0.

The sample space between 10 initial agents and 20 should be fully explored to determine the initial number of agents that cause the population to change from becoming extinct to reaching the carrying capacity. Since we do not need to run the simulation to accurately forecast the output for any initial number of agents <= 10 and any initial number of agents >= 20, the number of agents at time = 1500 is predictable. Identifying a behavior as predictable significantly reduces the number of inputs for which a simulation needs to be run for a given hypothesis test. Again, fewer simulation runs for a given hypothesis test improves the efficiency of exploration and EE.

Recall the simulation of the magnetic pendulum. Previously we considered the final resting position of the pendulum given the same release point. In this section we consider the final resting position of the pendulum given different initial release points for the pendulum. In Figure 8 pendulum is always dropped from the same height, 20 cm, above the two equilibrium positions but the distance from the x-axis which bisects the two equilibrium points varies.

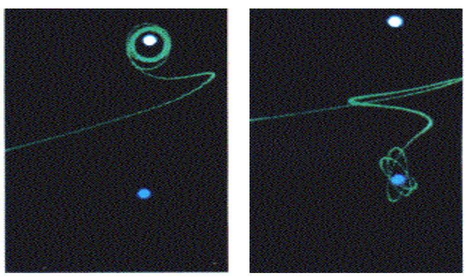

Figure 8: Projection of path of the pendulum with initial position: 20 cm above the red magnets and x= -1.0 cm from the x-axis on the left and initial position: 20 cm above the red magnets and x= -.75 cm from the x-axis on the right.

The final equilibrium position of the pendulum depends on the initial position. When the pendulum starts its motion closer to one magnet than the other, its path will become highly perturbed. The complexity of the pendulum's motion makes it nearly impossible to determine the final equilibrium position of the pendulum without running the simulation (or solving the simulation's equations) and observing the output. Since the simulation's output cannot be predicted without running the simulation the change in path of the magnetic pendulum for different initial positions is unpredictable. For unpredictable behaviors we must sample the entire input space to determine the behavior of the simulation for all inputs. This is computationally expensive, but necessary to determine an accurate characterization of the behavior.

3.3 Temporal Dimension

Most behaviors manifested by a simulation materialize over the course of simulation execution. Often, characteristics of a materializing behavior differ from characteristics of the same behavior fully manifested. Informally, a behavior *E* is *materializing* if its categorization in our taxonomy includes consideration for how the behavior has come about. A behavior E is *manifested* if the categorization focuses on what properties the behavior possesses in its most recently observed state. Definitions for temporal dimension properties are:

materializing – if the process by which the behavior has evolved is taken into account when placing it in a taxonomy.

If a behavior is not materializing then it is *manifested*. The majority of the time it is manifested emergent behaviors that are the behavior of interest to a user:

manifested **–** if the process by which the behavior has evolved is not taken into account when placing it in a taxonomy. Rather the behavior is treated as a final state, for the purposes of placing it in a taxonomy.

The descriptions of many emergent behaviors differ depending on whether the behavior is materializing or manifested. For example, a materializing behavior may be deemed <stochastic, unpredictable, materializing> whereas, following some epoch in the simulation, it is then considered <deterministic, predictable, manifested>. Both categorizations may be appropriate, depending on when the observations are made, and the goals of the user. As we have stated previously, a key goal is to design a taxonomy that is robust, supporting the exploration process even when a user categorizes a behavior in a manner that others might consider reasonable but less than optimal. We believe we have achieved this goal. The shape game provides some insight.

The shape game is set up as follows. Pick a positive integer n greater than 2. Place n points such that they are equally spaced on the 2-dimensional plane. The n equally spaced points are the vertices of an n-sided polygon, centered on the origin. Each of the n chosen points will be referred to as a vertex of the n-sided polygon. Choose n colors. Assign a unique color to each vertex. Take a 2n sided die where each of the n colors appears twice on the die. Choose any point inside the n-sided polygon; this point will be referred to as the seed. An example of the initial setup for the game with n = 3 is shown in Figure 10.

The shape game is played as follows. Roll the die. Depending upon which color comes up, move the seed half the distance to the similarly colored vertex. Repeat this procedure, each time moving the previous point half the distance to the vertex whose color turns up when the die is rolled. After 20 rolls, start marking the point indicated by moving half the distance to the vertex whose color turns up when the die is rolled. Repeat this process, marking the points at least 10,000 times. The result of the shape game using n=3 is the well known Sierpinski triangle. The Sierpinski triangle materializing and the Sierpinski triangle fully manifested are shown in Figures 11 and 12 respectively (Devaney 2004).

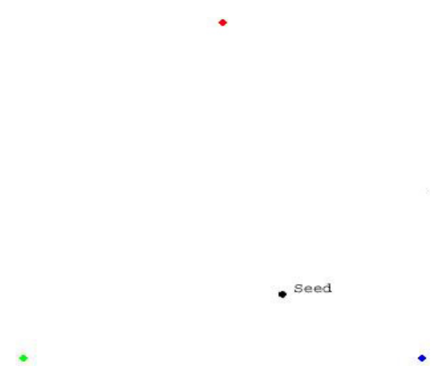

Figure 10: The setup for the Shape Game with n=3.

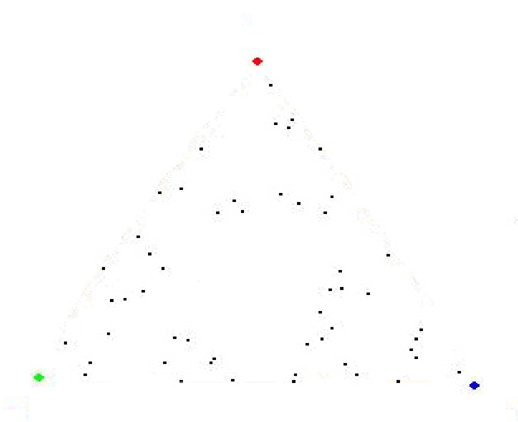

Figure 11: The Sierpinski triangle materializing.

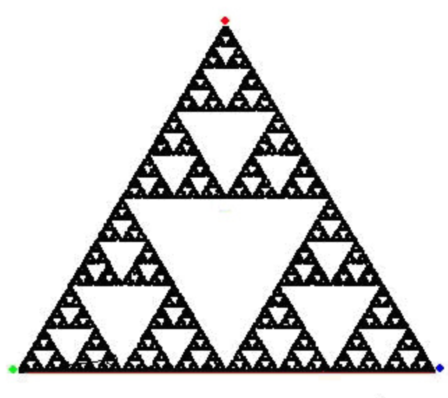

Figure 12: The fully manifested Sierpinski triangle.

The shape game emphasizes the importance of capturing the temporality of an emergent behavior. In Figure 11 when the Sierpinski triangle is materializing, the behavior is stochastic and unpredictable. It is impossible to predict the placement of the initial 30 marks in the output of the shape game for a given input. A user interested in testing hypotheses about the shape game for a given input as it is materializing would need to use the computationally expensive testing methods required for stochastic and unpredictable behaviors. However, when the output of the shape game is fully manifested (as it is for n=3 in Figure 12) the placement of the marks making up the output is stochastic and predictable for a given input. A user interested in testing hypotheses about the manifested output of the shape game could improve the efficiency of exploration by selectively sampling critical input points.

3.4 Employing the Taxonomy for Exploration

The previous subsections have presented our taxonomy for emergent behaviors, in support of the exploration process. Here we provide an example of how an emergent behavior can be explored more efficiently after it has been categorized in our taxonomy. Recall the simulation of agents interacting on a two commodity landscape. In Section 3.2 we considered the population of agents over time with different initial numbers of agents on the landscape. Three trials with different initial numbers of agents are shown in Figures 5-7. The behavior is identified as <stochastic, predictable, manifested>.

The behavior is predictable; only a small amount of the sample space of the conditions of interest need to be explored. From Figures 5 and 6 we can predict the population at time = 1500 for all initial numbers of agents >=20; the population will be 500 +/- 50. Similarly, using Figure 7 we can predict the population at time=1500 for all initial numbers of agents <=10; the population will be 0. The critical input points are 11<= initial number of agents <=20. Each critical input point should be tested according to the following criteria.

The behavior is stochastic; the trials for each critical input point must be performed multiple times in a fashion similar to Monte Carlo Sampling. Monte Carlo Sampling is computationally expensive but necessary to determine the range of observable behaviors of the simulation for each critical input point (Mooney 1997).

The behavior is manifested; the population of agents can only be measured once the population has reached its carrying capacity. In this example, we assume a SME has determined that at time =1500 the population will be in a steady state and the carrying capacity can be observed for all critical input points.

Further work remains. Our stated goals are efficiency in the exploration process and robustness. The previous example shows how a user would proceed exploring an emergent behavior after identifying the behavior with our taxonomy. However, we need to conduct further assessment to determine if we have met our goals, and whether alternative methods might be better.

3.5 A Robust Taxonomy

We are motivated by the needs of a SME wishing to test hypotheses efficiently as an emergent behavior is driven towards conditions of interest. We do not attempt to guarantee that every user will classify every emergent behavior exactly the same. Different user objectives may result in different classifications of the same behaviors in the same simulation.

Two different users may classify the manifested behavior of the shape game differently. One may identify the behavior as <stochastic, predictable, manifested> because of the die used to determine the placement of the marks that form the output. This identification is accurate; the coordinates of the individual marks forming the output can differ stochastically for the same initial conditions. Exploration would include running multiple trials for the same initial points and seed for the shape game to determine different possible configurations of markings that form the output.

Another user may view the observable behaviors of the shape game differently. Ignoring individual marks forming the output, the user may be interested only in the outline of the output. This user would declare the shape game <deterministic, predictable, manifested>. The overall output remains the same for the same initial points and seed given to the simulation. The user would run the shape game simulation only once for each n (number of points) and seed to explore the behavior.

We believe our taxonomy is robust, because it appears to support efficient exploration of a given behavior even when different users classify the same behavior differently. However, one must separate robustness (advantageous) from ambiguity (generally disadvantageous). Future work includes more detailed analysis of the utility of the robustness we believe our taxonomy provides.

3.6 Orthogonality of the dimensions

We strove for orthogonality among the dimensions of the taxonomy. We believe we have achieved it. One apparent class of counterexamples concerns behaviors labeled constant in the predictability dimension and deterministic in the reproducibility dimension. We explore this case here. Consider a coin with memory. If the immediately preceding flip of the coin yielded heads, then the coin is fair for the current flip, for any given trial there is a 50/50 chance of heads or tails. However, the longer any immediately preceding sequence of tails results, the more likely the current flip will yield a heads. With certainty we can add that following an uninterrupted sequence of n tails results the next flip will be a heads. A user flips the coin until a heads comes up. The user, not aware of the coin's biases, wishes to predict the likelihood of heads coming up at least once in a set number of trials. The behavior of the simulation is stochastic: the sequence of heads and tails, the observable behavior of the simulation, change from run to run. However, the predictability of the simulation is constant; under our assumption of a biased coin, at least one heads result will occur within n+1 trials.

The distinction between different types of behavior demonstrates the utility on our taxonomy: even all constant behaviors should not be tested in the same manner. <stochastic, constant> behaviors, as in the previous example, need to be observed over multiple trials with the same input to determine the range of different times that the behavior is manifested. <deterministic, constant> behaviors only need to be observed once, the behavior is manifested at the same time in each trial.

4 EXERCISING AND EVOLVING OUR TAXONOMY

In order to determine the variability that could occur when users are asked to classify an emergent behavior with our taxonomy we conducted an identification exercise with ten different simulations. Each simulation included a behavior description and user objective. The user objective is the question(s) the user wishes to answer by exploring the emergent behavior. In constructing the exercise we strove to make the user objective unambiguous. The goal of the exercise was to determine whether a test group would identify behaviors with attributes from the taxonomy consistent with the expectations of the designers.

Each of the ten simulations, descriptions and user objectives were to be identified with one attribute from the reproducibility dimension and one attribute from the predictability dimension. The temporal dimension of the taxonomy had not been developed. For 9 out of the 10 simulations at least 75% of the test takers identified both attributes of each simulation as we expected. For 7 out of the 10 simulations at least 90% of the test takers identified both attributes of each simulation as we expected.

The exercise revealed two issues with our taxonomy. First, most disagreement between our expected results and test takers' actual results stemmed from simulations we expected to be labeled as <deterministic, unpredictable>. These simulations corresponded to chaotic systems or deterministic systems where small changes in input cause significant changes in output. Test takers labeled these systems <deterministic, predictable>.

Chaotic behaviors are difficult to identify and explore. However, chaos theory is a well-established field with methods for exploring and identifying chaotic systems. In future work we plan to use chaos theory work to help users identify <deterministic, unpredictable> behaviors.

Test takers identified the need for a temporal dimension. They found that the exercise was ambiguous with respect to when a described behavior was under observation. Test takers mentioned that it was unclear as to whether they should identify the attributes of the materializing or the manifested emergent behavior. We realized a user interested in exploring emergence could be interested in exploring both the materializing and manifested emergent behaviors. The feedback resulted in creation of the temporal dimension of the taxonomy.

5 CONCLUSION

The goal of EE is to allow users to efficiently build confidence that a given emergent behavior is valid. It is not to develop a taxonomy of emergent behavior. Our taxonomy is not meant to resolve disagreements among SMEs about the classification of a behavior. However, in attempting to improve the efficiency of EE exploration we found it necessary to identify certain attributes of emergent behavior. SMEs can use the attributes to identify the minimum amount of exploration required by EE to determine the validity of an emergent behavior.

Our taxonomy is not complete, in the sense that it is not fully tested for best utility, and robustness. However, we do believe the dimensions appearing in our taxonomy reflect the kind of categorizations necessary to make emergent behavior exploration efficient. We will be examining utility and degree of robustness further in future work.

ACKNOWLEDGMENTS

We acknowledge support from NSF (ITR 0426971), and our colleagues in Modeling and Simulation Technology Research Initiative (MaSTRI) at the University of Virginia.

REFERENCES

Balaci, O. 1997. Principles of simulation model validation, verification, and testing. *Transactions of the Society for Computer Simulation International* 14: 3-12.

Bar-Yam, Y. 2004. Multiscale variety in complex systems. *Complexity* 9: 37-45.

Bedau, M. A. 2002. Downward causation and the autonomy of weak emergence. *Principia* 6: 5-50.

Chalmers, D. J. 2006. *The Re-emergence of Emergence*. Oxford: Oxford University Press.

Davis, P. K. 2005. New Paradigms and Challenges. In *Proceedings of the 2005 Winter Simulation Conference*, ed. M. E. Kuhl, N. M. Steiger, F. B. Armstrong and J. A. Joines Piscataway, NJ: Institute of Electrical and Electronic Engineering.

Devaney, R. L. 2004. Fractal Patterns and Chaos Game. *National Council of Teachers of Mathematics.* 98: 228-233.

Duit, R., M. Komorek, and J. Wilbers. 1997. Studies on educational reconstruction of chaos theory. *Research in Science Education*. 27: 339-357.

Epstein, J. M. and R. Axtell. 1996. *Growing Artificial Societies: Social Science from the Bottom Up*. Cambridge: The MIT Press.

Fromm, J.. 2004. *The Emergence of Complexity*. Kassel: Kassel University Press.

Gore, R., P. F. Reynolds, L. Tang, and D. Brogan. 2007. Explanation Exploration: Exploring Emergent Behavior. *Proceedings of the 21st International Workshop on Principles of Advanced and Distributed Simulation*, San Diego, CA: Society for Modeling and Simulation International

Johnson, C. W. 2005. What are emergent properties and how do they affect the engineering of complex systems?. *Reliability Engineering and System Safety* 91: 1475-1481.

Kim, J. 1999. Making Sense of Emergence. *Philosophical Studies* 95: 3-36.

Mooney, C. 1997. *Monte Carlo Simulation*. Thousand Oaks: Sage Publications Inc.

Riggs, P. 1992. *Whys and Ways of Science: Introducing Philosophical and Sociological Theories of Science*. Carlton: Melbourne University Press.

Seager, W. 2006. Emergence, Epiphenomenalism and Consciousness. *Journal of Consciousness Studies* 13: 21-38.

Zeigler, B. P., H. Praehofer, and T. G. Kim. 2000. *Theory of Modeling and Simulation*. 2nd Edition. Burlington: Academic Press

AUTHOR BIOGRAPHIES

ROSS GORE is a Ph.D. Candidate in Computer Science and a member of MaSTRI at the University of Virginia. His email address is <rjg7v@virginia.edu>.

PAUL F. REYNOLDS, JR. is a Professor of Computer Science and member of MaSTRI at the University of Virginia. He has conducted research in Modeling and Simulation for over 25 years, and has published on a variety of M&S topics, including parallel and distributed simulation, multi-resolution modeling and simulation. He is a plank holder in the DoD High Level Architecture. His email address is <reynolds@virginia.edu>.