# Cognitive Projection of Future States by Autonomous Entities

**Jared K. Snyder, Susanne Morse, Jay Potts, Todd Griffith, Ph.D,**
**Discovery Machine Inc.**
**Williamsport, PA**
**jsnyder@discoverymachine. com, smorse@discoverymachine. com, jpotts@discoverymachine. com, tgriffith@discoverymachine.com**

## ABSTRACT

As simulation plays an increasingly central role in training events, the requirements imposed upon autonomous entities within those simulations increase. In order to facilitate realistic training experiences, we expect the entities within our simulations to behave in a complex and human-like manner. One aspect of simulating human activities in complex simulated environments centers around enabling the projection of future states of a simulated world, in order to determine the relative merits of potential courses of action. This projection involves introspection of not only the entity's own state, but the state and anticipated actions of other entities within the environment.

This paper presents work sponsored by the Office of Naval Research (ONR) which demonstrates a general framework for how projection could be performed, with a focus on one method of imagining and evaluating future states. A strategy will be explained for how relevant information can be extracted from a simulation environment and converted into an abstract representation of the current state of the world. This representation will in turn be utilized by an abstract qualitative simulator, the NAUTical Intent and Location Utility Simulator (NAUTILUS), which will simulate the course of events by applying Subject Matter Expert (SME) created behaviors. These SME created and validated behaviors also contain a strategy for evaluating these courses of action which will be made use of by the Naval Evaluator of Missions and Objectives (NEMO) in appraising the state of the world. The NAUTILUS and NEMO provide the information necessary for an autonomous entity to determine whether the merit of the intended course of action validates its enactment. This projection framework will be displayed in an example Anti-Submarine Warfare (ASW) training exercise where an autonomous entity will project whether an intended course of action merits execution.

## ABOUT THE AUTHORS

**Jared K. Snyder** is a Software Engineer for Discovery Machine Inc., whose work focuses on behavior modeling and developing core technology offerings. After graduating with honors from Bucknell University with a Bachelor's of Science degree in Computer Science, Mr. Snyder has been designing and developing software and behavior models for Discovery Machine Inc. since 2009. His work has focused on creating behavior models for simulated entities within constructive military training environments that behave and interact in a human-like manner.

**Susanne R. Morse** is a Software Engineer for Discovery Machine Inc. She started working for Discovery Machine, Inc. in 2012 after graduating from Bucknell University with a Bachelor's of Science degree in Computer Science. Her work has focused on the creation of models with realistic behaviors in multiple simulation environments, including models whose behavior dynamically changes with human-entity radio communications.

**Jason R. Potts** is the Vice President of Software Development for Discovery Machine Inc. Since graduating with honors from Worchester Polytechnic Institute with a Bachelor's of Science degree in Computer Science, Mr. Potts has been designing and developing software for Discovery Machine Inc. since 2001. This has included development of tools facilitating knowledge capture, and representation of human expertise in a machine executable form.

**Dr. Todd W. Griffith** is the founder and CTO of Discovery Machine. He has been a principal investigator in 25 funded projects, working in the area of intelligent systems research for over twenty years and has published papers in the areas of cognitive science, human-computer interaction, and intelligent systems. Prior to founding Discovery Machine, Dr. Griffith taught computer science and artificial intelligence at Georgia Tech. and Bucknell.

# Cognitive Projection of Future States by Autonomous Entities

**Jared K. Snyder, Susanne Morse, Jay Potts, Todd Griffith, Ph.D.**
**Discovery Machine Inc.**
**Williamsport, PA**
**jsnyder@discoverymachine. com, smorse@discoverymachine. com, jpotts@discoverymachine. com, tgriffith@discoverymachine.com**

## INTRODUCTION

As simulation-based training continues to expand, the need for intelligent, automated entities for use in simulation-based training events increases in kind. With the inclusion of intelligent automated entities, training exercises can increase in fidelity and complexity, without increasing the costs associated with human operators (Potts, 2012). This expansion in the use of automated entities must however, be coupled with ensuring these entities make decisions in realistic and human-like ways.

In order for simulated entities to act intelligently, they need to understand the state of their environment including what their adversary is doing. Unlike the positions of allies, adversaries do not willingly share their intentions. Thus, simulated entities need to project or predict what the most likely next move or series of moves will be for that adversary. While there are numerous computational ways to project using acquired data, here we are proposing a technique that has been applied in an area of artificial intelligence called qualitative physics modeling. Specifically, we propose providing each entity the ability to perform projection or to simulate its mental model of the situation in order to come up with possible future states.

The following presents work funded by the Office of Naval Research (ONR) to implement the ability for projection inside of automated entities. Given a specific training event for an Anti-Submarine Warfare (ASW) watch-team, automated entities representing the individuals in the ASW watch-team were developed. Next, a generalized definition of the process of performing projection was conceived. Then each stage of this process was implemented and injected into one of these automated entities, the Anti-Submarine Warfare Evaluator (ASWE).

This paper starts by presenting a general definition of the process for performing projection. Next a general description of how each part of this four-stage process is implemented is described. Finally, a use case of the training event involving the ASWE, an aircraft carrier, and an enemy submarine is presented. This example case demonstrates how an automated entity executing in the Joint Semi-Automated Forces (JSAF) simulation is able to gather information from the simulation, imagine how events will play out in faster than real time, and interpret multiple courses of action into a Subject Matter Expert (SME) validated answer.

### Defining Projection

Projection can be described as the ability to imagine the consequences of performing a specific action. Projection is an important tool humans use to decide how to behave. Being able to postulate multiple courses of action, projecting how events may unfold for each proposed course of action, and evaluating which course of action is best, allows a human to make a better decision as to what action will achieve an optimal result.

In order for automated intelligent entities to use projection in their decision making, they must try and replicate the process by which humans perform projection. This work presents a process of projection that is decomposed into a step-by-step process.

Figure 1 below illustrates how our approach breaks down the projection process into the following four steps:

1. Determine Initial World State
2. Evaluate Initial State
3. Simulate and Reevaluate
4. Interpret Projection Results

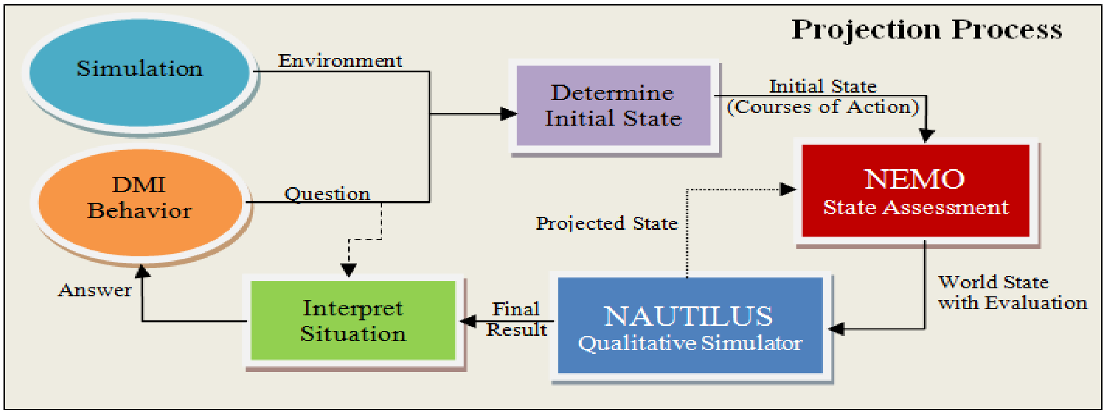


**Figure 1. Projection Process Diagram**

The first stage of *Determine Initial World State* deals with evaluating which entities in an environment are relevant to answering the posed question and what they are trying to accomplish. The first stage of projection is also where the imaginer generates any possible courses of action to solve the question that was asked. The next stage, *Evaluate Initial State*, deals with evaluating the initial layout to base whether or not future states have progressed to a better or a worse state than the original state. Next the *Simulate and Reevaluate* phase imagines the interaction of all the relevant entities' courses of action and evaluates this new state to determine whether things are remaining in status quo, getting better, or worsening. Finally, the *Interpret Projection Results* stage interprets the result of each course of action played out and selects the best of these courses of action.

### Qualitative Simulator

At the core of our projection is a qualitative simulator that we use to imagine out each course of action. The field of Qualitative Physics (Hayes, 1979) has been around for several decades and attempts to provide a cognitive explanation for how people reason about the world in terms of cause and effect. Qualitative simulation identifies what state would happen next and is applied recursively to identify all sequential states that follow from the initial qualitative state (Forbus, 2007). In this way, previous events and interactions between entities affect future states, something that is necessary for accurately predicting the consequences of performing certain tactics in an engagement. The ideas utilized by the qualitative simulator are based on the Qualitative Process theory introduced by Kenneth Forbus, which allows us to make several qualitative deductions including what is happening in a physical situation, what are the combined effects of several entities, and determining when the projection should end (Forbus, 1984). All of these concepts were used in implementing our projection process. Before explaining how the projection was implemented, some of the benefits of injecting projection into automated entities and some background information should be presented.

### Benefits

The ability for simulated entities to use projection when making decisions offers the following advantages:

- More dynamic interactions between simulated entities and users
- More nuanced decision making for simulated entities
- Less scripted and more robust scenarios
- Intelligent teammates who naturally use projection can be accurately modeled
- Simulated entities can act independently for longer, freeing trainers to focus on the training mission

## BACKGROUND

Throughout the described research effort, we have employed a methodology and toolset designed specifically for capturing knowledge from SMEs. The methodology works with the SMEs through five phases: expert domain scoping, conceptualize, formalize, operationalize, and test/deploy. Each of these phases leverages a graphical modeling interface to encourage and facilitate introspection and articulation by the expert. The modeling environment utilizes a visual task decomposition language called Task-Method Knowledge (TMK) (Murdock, 1998) which allows for the easy creation of models that are human-readable and understandable by users who lack computer programming skills (Potts, *et al.*, 2012).

In general, the TMK based approach to behavior modeling offers the following advantages (Potts, *et al.* , 2010):

- Rapid development of new behaviors for customized training scenarios
- Expert domain knowledge captured within the modeling tools to assist building more intelligent behaviors
- Transparency of behaviors for subject matter experts during and after development

## PROJECTION PROCESS

The following sections describe how the four steps to our projection process illustrated in Figure 1 were implemented.

### Determine Initial World State

The projection process begins with a question being asked that, in order to be answered, requires a future state to be evaluated. After a question of this nature is asked two things must be done, first, create an abstraction of the current geospatial layout of the world with any entities whose presence or actions will affect the answer to this question and, second, determine what objective these entities are trying to accomplish.

Creating an abstraction of the world requires being able to get the current state of entities from whatever simulation the automated entity is operating in. This is an easy task in most cases. The more difficult part is deciding which entities are relevant to add to the abstraction of the world. For some questions, not all entities may be relevant when projecting what the future world will be like. This selection process relies on talking to a SME who regularly does this type of projection and can explain how they determine which entities they determine have an impact on the question asked.

After determining which entities need to be included in the abstraction of the current world state, each of these entities is assigned an objective based on what the projector, hereafter referred to as the "imaginer", believes they are trying to accomplish. The way in which the imaginer assigns these objectives will also require insight from a SME. A SME may base how they assign objectives to entities based on positioning, past actions, direct knowledge, or any other strategy they use to determine this. Once all the entities have been added to this abstract picture of the world and assigned an objective, one final but crucial determination must be made, what objective should be assigned to the entity which represents the imaginer?

The objective assigned to the imaginer represents one possible course of action to project in order to answer the question posed to the imaginer. Some questions may allow or force multiple courses of action to be projected. If this is the case the imaginer will duplicate the abstraction of the world and simply change the objective assigned to themselves. These projections are independent of the simulation and therefore can be projected in parallel. Thus as soon as all the possible courses of action are created, evaluation of the initial world state can start.

### World Abstraction and Objectives

In order to gain qualitative view of a quantitative simulation environment we must have a way to represent the quantitative data in a qualitative way. Our world abstraction is broken down into two parts, the current world state and each entity's current goal. The world state consists of all the entities deemed important in the simulation. It allows us to keep track of the details for each entity, such as their projected location, speed, and heading. The system categorizes all entities based upon their importance to the imaginer into four groups, the imaginer, hostile entities, allied entities, and other entities. These categories allow us to alter the projected movement of an entity based on its importance. The world state also keeps track of the evaluation of whether this state is Good, Safe, or Bad for the imaginer.

We base this model on the fact that each entity in a simulation has a goal or course of action to complete. These goals enable us to determine proper courses of actions for each entity as well as provide an evaluation of the world state. We introduce the notion of an "Objective", represented as a data object which embodies an entity's goal. At the core of our system there is a base Objective, which is a generic representation of the characteristics that must be shared by all possible courses of actions. The base Objective provides a framework that can be extended to create specific custom objectives. Each sub-objective then represents a specific goal for an entity to complete (Figure 2).

In order to isolate the complexity of the simulation, there are two methods or behaviors associated with each objective class.

1. Perform Actions
2. Evaluate Completion

The Perform Actions behavior determines the next movement/action for an entity to complete. The Evaluate Completion behavior determines the status of an entity in the world. The execution and specific details of each behavior will be discussed further in later sections. The use of these behaviors allows us to customize each objective with SME-validated models. Using the TMK language, knowledge from SMEs is easily captured and transformed into these behavior models. These models are then executed during the projection and evaluation processes in order to project each entity as accurately as possible based on the expected behavior of each course of action.

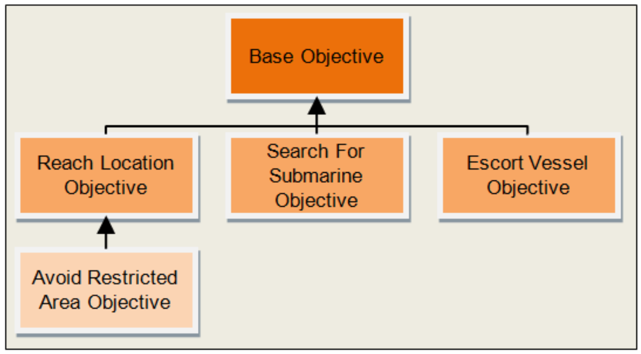


**Figure 2. Example Objective Hierarchy**

In addition to gaining the ability to accurately project each entity, the use of objectives removes the complexity from the rest of the projection process. Since each objective is responsible for both performing the correct courses of actions and evaluating the current state of an entity, both the evaluation and simulation processes only need to be concerned with iterating through each entity and determining when to stop projecting. Both processes are described in the following sections.

### Evaluate Initial State

After the initial world state has been determined it must be evaluated as a good, bad, or safe state. This evaluation provides an indicator of when to stop the qualitative simulation described later. The evaluation occurs in the state assessment module which executes the Evaluate Completion behavior for each entity according to their objective to determine the world state to be either a good, bad, or safe state. Because each entity's objective is customized, each individual entity is evaluated in determining the overall world state. By default each state is assumed to be a safe state. A bad state can be reached if an objective fails or when the situation is determined as dangerous. A good state can be reached when an objective is completed. In addition, the state assessment module has the ability to remove entities from the simulation if they are determined to be no longer important for the projection, based on the courses of action. Removing entities increases the efficiency of the simulation by reducing the number of entities to project.

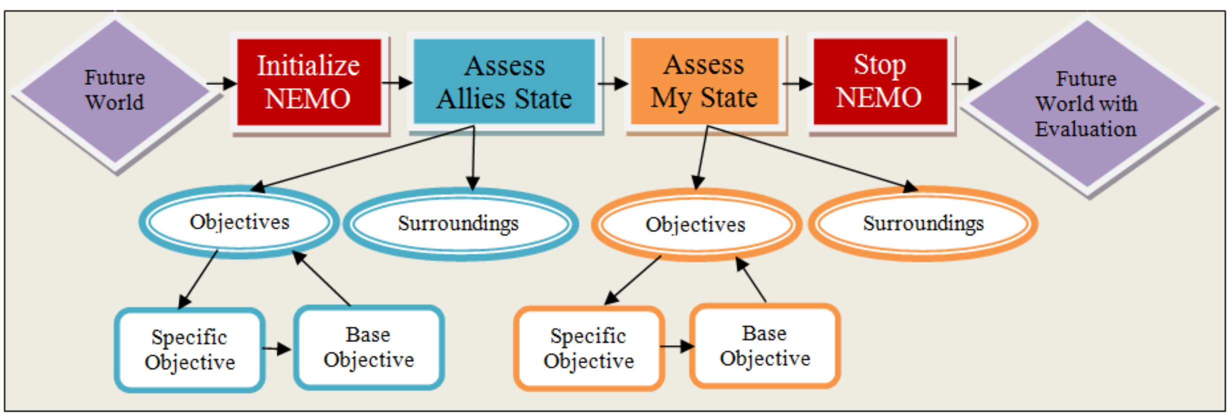


**Figure 3. World Evaluation Process**

**World Evaluation Process**

The whole evaluation process is shown in Figure 3. The input is a world state described earlier. The evaluation of a state starts by assuming the state is safe. It then proceeds to evaluate the state of all high value units. The Evaluate Completion behavior for each high value unit's objective is executed or its surroundings are evaluated if the entity has no objective. If at any time the state is determined to be bad, through a failed objective or dangerous surroundings, the evaluation stops. After checking all high value units, if the state is not bad, the main entity's state is evaluated. As a result, the end state will either be bad or the state of the main entity (i.e., if the main entity has completed its objective and all high value units are safe, the state will be good).

**Entity State Evaluation**

Each Evaluate Completion behavior contains a base evaluation. This evaluation determines if any hostile entities are within firing range of the given entity. It does this by comparing the distance of the hostile entity to the given entity and the maximum firing/detection range of the hostile entity. The state is determined to be safe if no hostile entities are within firing range and bad if there are hostile entities within firing range of the given entity. If an entity does not have an objective the same evaluation as the base objective is used.

The evaluation of each specific objective uses their custom Evaluate Completion behavior. If the objective is not completed or failed it is by default safe. The specific objective can also execute the base objective evaluation to see if it is in danger, or if danger is allowed based on the objective (e.g., the Hunt Sub objective) it can choose to ignore the base objective evaluation.

**Simulate and Reevaluate**

After evaluating the initial state, progressing this state into an imagined future begins. This progression is accomplished by a qualitative simulator (See Figure 4). The qualitative simulator is responsible for iterating through all the entities and updating their situation and position based on their surmised objective. What follows is a breakdown of how the qualitative simulation functions.

Before the work of actually progressing forward can begin, the qualitative simulator must be initialized with a delta time and a maximum projection time. These inputs are necessary for the following reasons. If the projection cannot be accomplished in less time than the events will actually play out, it is rendered useless, thus a delta time must be given such that the projection can occur at faster than real time. If a projection has the possibility of not reaching a change in status before it is rendered useless it must be allowed to terminate when this maximum allowable time is reached.

Once the delta time and maximum projection time have been specified the process of progressing entities forward in time begins. As previously discussed, entities are classified by their importance to the entity imagining. In our approach, entities are progressed forward in the following order: Others, Hostiles, Allies, and Imaginer. This is to emulate the imaginers tendency to base the movement and decision of the latter on the former. The process of progressing any of these entities forward is relatively the same except for the fact that the world is changing as the entities are being iterated over.

In order to keep the qualitative simulator as simple as possible, the way in which the simulator will progress the entity forward in time is encapsulated in the objective assigned to the current entity. As previously mentioned all objectives contain a Perform Action strategy which defines how it should behave. The base objective's Perform Action strategy applies the entities specified speed and heading to update its position in the set delta time. All custom objectives that derive from this base objective are thus expected at the very least change the position of the entity. The Perform Action strategy of the custom objectives can range in complexity from trivial to sophisticated. For example, a complex Perform Action strategy may base its behavior on its surroundings, such as moving to avoid other entities it deems threatening, or creating new subordinate entities with their own objectives. An important difference is that the Perform Action strategy may also differ based on whether the entity being progressed is the imaginer or one of the imagined entities. As the imaginer, all the information can be accessed, while imagined entities should only have access to information that would realistically be available to them via their sensory perceptions.

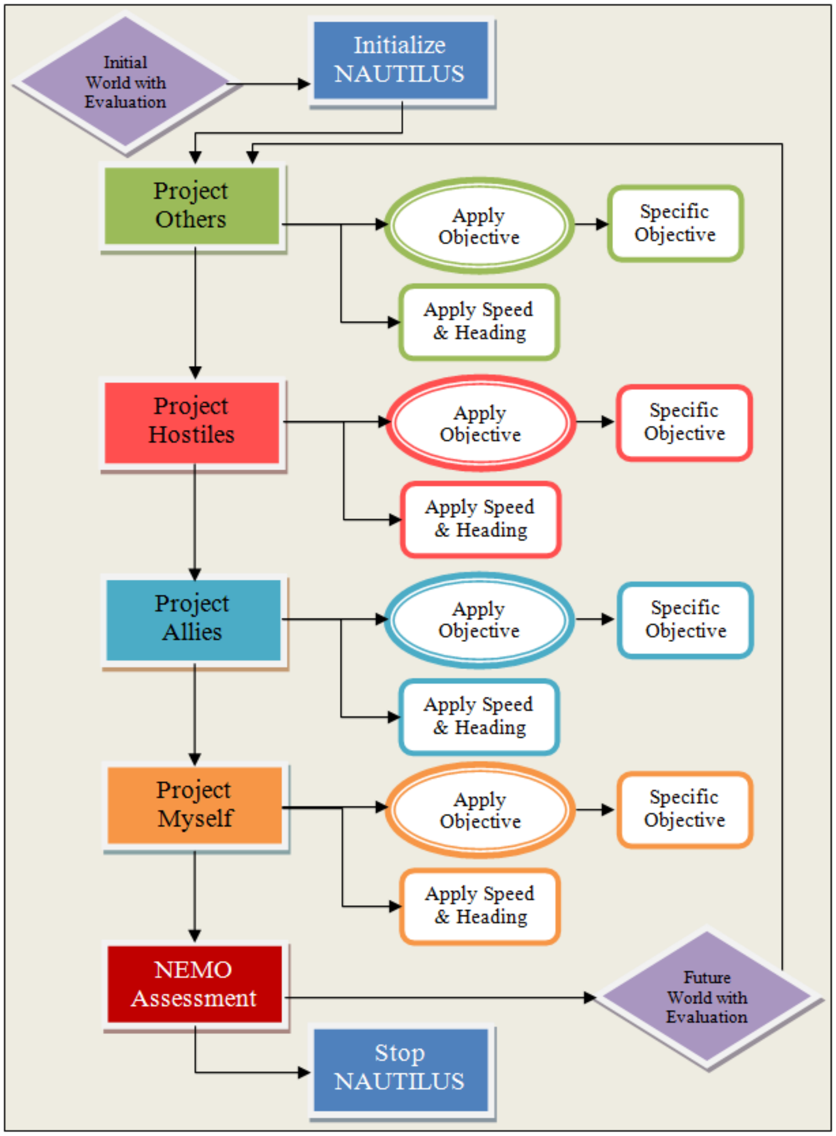


**Figure 4. Qualitative Simulator Process**

After progressing every entity forward by applying their specific objective to the current world state an imagined future state is created. This future state must then be reassessed. After evaluating this new state, the decision as to whether we should continue projecting needs to be answered. Our approach is roughly based on the equality of change law (Forbus, 1984), where the projection should continue unless one of two events occurs. The qualitative simulator should stop if the future state was evaluated to be different than the initial state, that is, we have gone from safe to good or bad, from good to safe or bad, or from bad to safe or good. The other condition that will cause the qualitative simulator to stop is if the maximum projection time has been reached. If neither of these conditions are met, the qualitative simulator will take the newly generated future state as the state from which to generate a new future state.

By using the newly generated future state as the state to project farther into the future from, the qualitative simulator continually builds upon its own projections much in the same way humans do when imagining how a course of events will play out. This building upon past projections also allows for more dynamic behavior from the imagined entities as they will interact with each other and how they behave as the projection is created. Once we have reached one of the aforementioned end conditions, the qualitative simulator conveys this imagined future state to the next stage of the projection process, interpretation of this imagined future state.

### Interpret Projection Results

The final result of the qualitative simulator gives us an abstraction of the geospatial layout of all the entities and an evaluation, based on assigned objectives, of whether this future state is good, bad, or safe. This geospatial layout and evaluation must be interpreted in the context of the original question to form an appropriate answer. In this way the results of our qualitative simulator are not interpreted based on the contents of the final state, but rather the relationship between the initial and final states (de Kleer and Brown, 1982). The interpretation stage relies on the objective assigned to the imaginer and the SME knowledge used in the Determine stage. Another key consideration of the interpretation stage is whether multiple courses of actions were projected.

When only one course of action was projected, interpreting the resulting future state can be done simply by comparing the initial evaluation with the final evaluation. This is based upon the fact that if an appropriate objective was assigned to the imaginer, its evaluation will do much of the interpretation of the geospatial layout during the evaluation stage. If the state remains good, remains safe, changed from bad to safe, changed from bad to good, or changed from safe to good the answer can be the affirmative. If the state remains bad, changed from safe to bad, or changed from good to bad answer can be the negative. The harder projection result to interpret is when the state changes from good to safe. The answer in this case will require a SME-validated method of discerning a quantifiable level of goodness from the final layout of the vessels in question.

If multiple courses of action were projected, interpreting the results can be done in a multitude of ways. The way in which the courses of action are selected to return as the answer should be based on SME knowledge. For example in one case, a SME may specify that the projected results of the courses of action should be checked in a specific order, representing a preference for accomplishing a mission in one way before trying another. Another method may require that any projection which returns in the affirmative be judged for its relative "goodness" by a SME-validated evaluation strategy. In earnest, a question may require a unique SME-validated way to interpret the results depending on the difficulty of the question which was the cause of the projection process to begin.

## USE CASE

A depiction of our use case is found in Figure 5. It involves a carrier ship (at the center of the image) and a hostile submarine (in the top right corner). The carrier ship is trying to cross the finish line at the top of the image. The circle in the image represents the point at which the sub will be in firing range of the carrier. The rectangle represents a restricted area the carrier may not travel through. Our Anti-Submarine Warfare Evaluator (ASWE) automated entity must determine the best course of action for the carrier to cross the line without leading it into a compromising situation. We define a compromising situation to be the sub moving to firing range of the carrier or the carrier entering the restricted area.

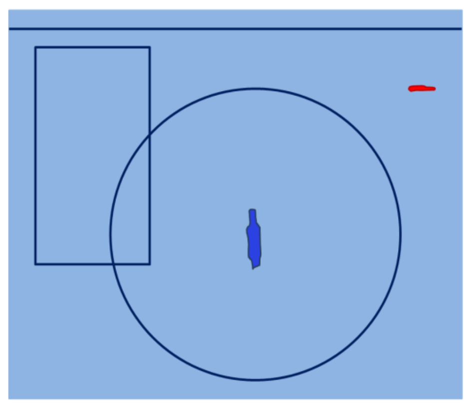

**Figure 5. Use Case Geospatial Layout**

To start the projection process the ASWE addresses the question "How can the carrier cross the finish line without getting into a compromising situation?"In determining the initial world state, the carrier is the imaginer and the submarine is a hostile entity. There are no other entities in the initial scenario. The world state details for each entity are retrieved from JSAF. Through geospatial reasoning and SME knowledge, the ASWE decides the sub's objective is to hunt the carrier while evading detection. The carrier's objective is to cross the line. Based on SME knowledge the ASWE determines three potential courses of action:

1. Turn to Avoid Sub
2. Force Sub to Leave Area
3. Increase the Carrier Speed

The SME knowledge indicates that the order of preference for each course of action is the order they are listed. The Turn to Avoid Sub course of action is applied by changing the carrier's heading before the qualitative simulation begins. For our use case we will evaluate turning the carrier 15, 30, and 45 degrees away from the sub. The next course of action, Force Sub to Leave Area, adds a helicopter to the scenario as an allied entity. There are two possible objectives for the helicopter, one the helicopter drops sonobuoys to deter the sub and the other the helicopter attacks the sub with a torpedo. Both objectives will be considered. The last course of action increases the carrier's speed. All courses of action are simulated in parallel. Since all runs are similar, we will describe the evaluation and simulation of the first course of action.

After the initial state is determined it is sent to the state assessment module, called the Naval Evaluator of Missions and Objectives (NEMO) in our test case. Since there are no allied entities, only the imaginer's state is considered. The imaginer's objective is retrieved and its corresponding Evaluate Completion strategy is executed. For the cross line objective, the Evaluate Completion strategy will return a bad state if the entity has entered the restricted area. The strategy will return a good state if the entity has not entered the restricted area and has crossed the finish line. All other situations will result in a safe state. Since the carrier has not crossed the finish line or entered the restricted area, the strategy returns a safe state. After the Evaluate Completion strategy for the cross line objective has completed, the Evaluate Completion strategy for the base objective is executed. This strategy will change the state to a bad state if a hostile entity is within firing range of the given entity. Since the sub is not within firing range, the base objective's Evaluate Completion strategy does not change the initial state. As a result NEMO will evaluate the initial state as a safe state.

After the initial state is evaluated, it is sent to the qualitative simulator, called in this use case as the NAUTical Intent and Location Utility Simulator (NAUTILUS). The NAUTILUS progresses this state forward by a preset delta time. This is accomplished by executing the Perform Action strategy of each entity's objective to decide its future location. The sub is simulated first since it is a hostile entity. Its objective is retrieved and executed. The Perform Action strategy for the hunt carrier while evading detection objective will move the sub to intercept the carrier unless the sub detects a threat nearby. Possible threats are active sonobuoys, ships with passive sonar, and torpedoes. If a threat is detected the Perform Action strategy will move the sub to avoid the threat. In this instance, since no threats are nearby, the strategy moves the sub towards the carrier. The carrier's objective is then retrieved and executed. The Perform Action strategy for the cross line objective simply moves the carrier forward on its current heading a distance determined by the speed of the carrier and the delta time. After both entities have been progressed forward, the resulting state is evaluated by NEMO. The first projected state is evaluated as a safe state the same way as described earlier. The NAUTILUS then decides whether it needs to continue progressing forward based on the evaluation's result. Since the state is still safe, the same as the initial state, the NAUTILUS loops to progress to the next state. The process continues until the sub gets within firing range of the carrier, before the carrier crosses the finish line, at which point the world state is evaluated to be a bad state which ends the simulation.

**Table 1. Evaluation of Multiple Courses of Actions**

| **Course of Action** | **Initial State** | **Final State** |
|---|---|---|
| Turn 15° to Avoid Sub | SAFE | BAD |
| Turn 30° to Avoid Sub | SAFE | SAFE |
| Turn 45° to Avoid Sub | SAFE | BAD |
| Force Sub to Leave Area with Sonobuoys | SAFE | SAFE |
| Force Sub to Leave Area with Torpedo | SAFE | SAFE |
| Increase Carrier's Speed | SAFE | BAD |

All other courses of action were executed in parallel to the describe course of action, so all the final resulting states are available in the interpret stage. Using a visualization tool, all courses of action were plotted based on their projection runs (see Figure 6). Table 1 shows all the courses of action that were evaluated with their initial and final states. The first step to interpreting the results is to throw out all results that ended in a bad state, which eliminates the first, third, and the last option. Then using the SME knowledge of the preference order for the courses of action, the best course of action is determined. In this case, the best course of action is to turn the carrier 30° away from the sub. The ASWE may then make a recommendation based on this course of action. Note for these runs the maximum projection time was 30 minutes. As a result some of the courses of action remained in a safe state. Instead of making a recommendation from these results the ASWE may also decide to increase the projection time and rerun the courses of action that ended in a safe state. The decision to rerun courses of action would be made by the ASWE behavior model based on the projection results.

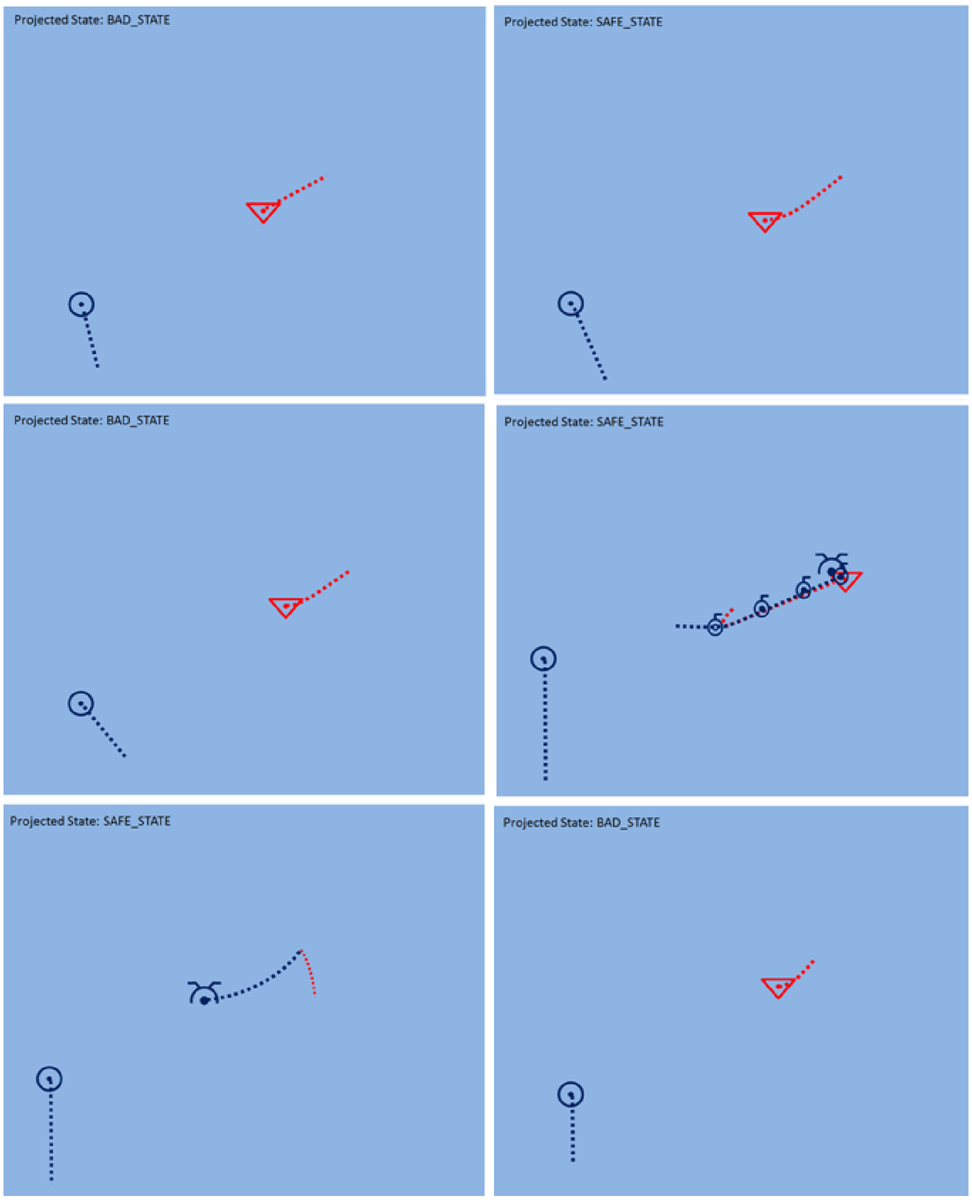


**Figure 6. Visualization of Projected Future State**

## FUTURE WORK

Moving forward with our projection process, we plan to increase our ability to project complex scenarios. Currently our projection process only addresses geospatial reasoning problems. We plan to expand our projection to go beyond geospatial reasoning problems to include scenarios where other reasoning methods are needed. We also plan to adapt our use of objectives to give each entity the opportunity to have multiple objectives. In many known cases the success of an entity is tied to the successful completion of a subordinate's objective. For example, a ship with an objective to hunt a submarine may launch a helicopter to destroy the sub. The ship will then rely on the helicopter's success. With slight changes to the objective structure, we hope to achieve this interconnectedness of entities. As we enhance our ability to handle different scenarios, our projection process will become increasingly useful in simulated entities' behavior models.

## CONCLUSIONS

This paper has presented our approach to implementing projection in automated entities performing an example simulated training event. This ability to project future states resulted in a much more nuanced decision making process in the ASWE automated entity and instilled a more human-like quality to its implementation. It also enabled an automated entity to make a much higher level decision in a short amount of time than other methods, such as a rule based system, might have been able to provide. This translates into an even more autonomous entity capable of making increasingly higher level decisions in a more human-like way. In striving to provide more realistic and engaging training, projection-enabled automated entities bring this goal closer to fruition.

## REFERENCES

Chandrasekaren, B. (1986) Generic Tasks in Know-ledge-Based Reasoning: High-Level Building Blocks for Expert System Design, *IEEE Expert*. Fall, 1986.

deKleer, J. & Brown, J. S. (1982) Foundations of envisioning. Proceedings of the Second National Conference on Artificial Intelligence. 434-437.

Forbus, K. (1984)Qualitative Process Theory. *Artificial Intelligence*, 24:85-168.

Forbus, K. (1989) Introducing actions into qualitative simulation. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, 1273-1278.

Forbus, K. (2007)Qualitative modeling. In Harmelen, F. , Lifschitz, V. , and Porter, B. (Eds) *Handbook of Knowledge Representation*. Elsevier.

Garcia, C. J. & Griffith, T. W. (2005) A Composable Behavior Modeling System for Rapidly Constructing Human Behaviors, *Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2005*.

Hayes, Patrick, J. (1979) “The Naive Physics Manifest”, in D. Michie, ed., *Expert Systems in the Micro-Electronic Age*, Edinburgh: Edinburgh University Press, 242-70, as repr. in Boden, ed., 171-205.

Hinrichs, T. , Forbus, K. , de Kleer, J. , Yoon, S. , Jones, E. , Hyland, R. , Wilson, J. (2011) Hybrid Qualitative Simulation of Military Operations. In *Proceedings of the Twenty-Third Innovative Applications of Artificial Intelligence Conference*. 1655-1661

Prolegomena to a Task-Method-Knowledge Theory of Cognition. J. William Murdock, pp. 746-751, Proceedings of the Twentieth Annual Conference of the Cognitive Science Society (Cogsci'98), Madison, WI, August 1-4, 1998

Potts, J. R. , Griffith, T. , Roth, K. , & Snyder, J. (2012) Customizable speech centers for automated entities within simulation. Proceedings of the Interservice/Industry Training, Simulation, and Education Conference.

Potts, J. R. , Griffith, T. , Sharp, J. J. , & Allison, D. (2010) Subject matter expert-driven behavior modeling within simulation. Proceedings of the Interservice/Industry Training, Simulation, and Education Conference.