Modeling behavior of units in combat with Monterey Phoenix

Alexander Nguyen

Mentored by Dr. Kristin Giammarco

Introduction

In the Marine Corps, live training is essential for soldiers to gain the experience necessary for real-life combat. However, live training is costly both in time and resources leading to value in the option of virtual training. Behavioral modeling can help build a database of various possible scenarios and highlight risks present in each scenario. The purpose of this project was to model the behavior of units within a Marines combat scenario and expose underlying risks using the behavioral modeling tool Monterey Phoenix (MP). MP utilizes lightweight formal methods to automatically generate all possible behaviors as traces within a user-defined scope (Giammarco et al., 2017). Traces are visual diagrams that allow users to see the behaviors acted out visually. Events within the model are coordinated with each other to define interactions and are then constrained so that only traces that meet the requirements of the model are displayed (Quartuccio et al., 2017). However, events that cause unexpected behaviors that were not considered by the user can be exposed, preventing a loss of resources in the future. This allows the user to easily identify potential sources of risk, providing valuable information to the Marine Corps about what they could add to their virtual training. Risk analysis of scope-complete behavior models is not common practice even though it would be beneficial in many scenarios. Using MP to perform risk analysis on Marines combat scenarios would demonstrate the viability of MP for performing quantitative risk analysis on real-world situations, potentially leading to more widespread application of risk analysis in behavioral models.

Materials and Methods

The final model was based on the Breaching Protective Obstacles scenario within the Marine Rifle Squad manual (U.S. Marine Corps, 2020). MP-Firebird software version 4 was utilized (https://firebird.nps.edu/). To model the different actors within the scenario, each of them were defined as a root event with their own

Figure 1 (right): Code within MP that defines the different attributes that are assigned to the events. The likelihood and impact attributes were coordinated to assign specific values to each of them. Other attributes were then used to store values in the global data table and report.

```
ATTRIBUTES { number likelihood, impact, trace_risk_score,
        sum_risk_score, max_risk_score,
        max_risk_trace_unique_number,
        risk_score, Risk_Score;
};

COORDINATE $employ_smoke_grenade: Toss_smoke_grenade
DO $employ_smoke_grenade.likelihood:= 0.8;
$employ_smoke_grenade.impact:= 1;OD;

COORDINATE $breach_requires_explosives: Initiate_breach_through_explosive_means
DO $breach_requires_explosives.likelihood:= 0.4;
$breach_requires_explosives.impact:= 7; OD;

COORDINATE $breach_mechanical_means: Initiate_breach_through_mechanical_means
DO $breach_mechanical_means.likelihood:= 0.6;
$breach_mechanical_means.impact:= 7; OD;

COORDINATE $Teammate_injury: Teammate_injured
DO $Teammate_injury.likelihood:= 0.4;
$Teammate_injury.impact:= 15; OD;
```

Materials and Methods (continued)

atomic events for simplicity. In an earlier iteration, using composite events to model the hierarchy of actors in the scenario more accurately resulted in visual clutter, making it less visually accessible shown in Figure 2. Interactions were established between the teams, obstacle, and enemy combatants using coordinate statements. Events that were performed by the same actors were connected using share statements. Reasonable alternate events not included in the manual were included within the model. Verification and validation of the model was based on feedback from subject matter expert (SME) U.S. Marine Major David Beard. Risk analysis was then performed by assigning numerical likelihood and impact attributes to events that were reasoned to be likely causes of failure for other events as shown in Figure 1. These values were then multiplied together to calculate a risk score, which was displayed using a data table within the global view of MP. Higher risk scores indicate a higher chance of mission failure for a scenario.

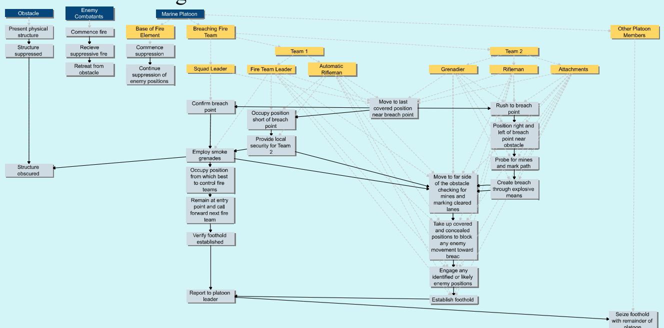

Figure 2 (above): Trace generated from a previous version of the model. Dashed lines represent shared process ownership between different events.

Results

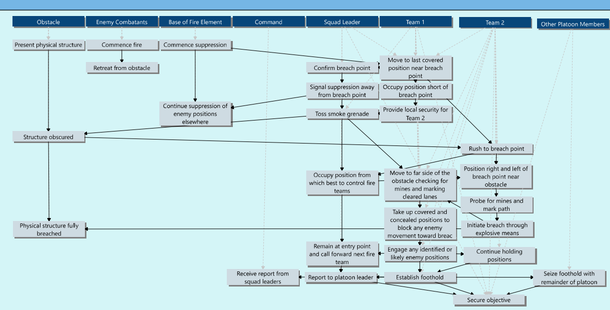

Figure 3 (above): Trace generated from final version of the model at a scope of one. Blue boxes represent root events.

Results (continued)

The final model had eight root events to model each actor in the scenario including the obstacle and enemy combatants. At scope one, the model generated 32 traces, scope two produced 192, and scope three produced 576 traces. The global average risk score at scope one was 7.3, scope two was 9.4, and scope three was 11.6. The run speeds from scope one to three were 617 events/sec, 395 events/sec, and 215 events/sec. The model had a runtime of 2.96 seconds at scope one, 26.17 seconds at scope two, and 148.68 seconds at scope three, as derived from the console window in MP. The model was successful in highlighting high risk events using the global report statement and organizing the total risk scores for all traces within a global data table.

Trace Number	Trace Risk Score
1	3.6
2	9.6
3	5
4	11

Table 1 (above): Table within the MP global view, displaying each trace's risk score for all traces in scope one.

Risk Report for Scope 2
Total risk over 192 traces (sum of trace risk scores): 1814.4
Highest Risk: 16 (trace 108)
Average Risk: 9.4
Sort by Marked to view traces with above average risk >= 4.

Table 2 (left): Global risk report generated when model was run at scope two. Displays summary of trace risk score data. The risk threshold was set at a risk score of four or higher.

Conclusions

The final model was able to accurately generate and visually display potential outcomes based on a known scenario from a Marines training manual. Due to only using root events to model the agent hierarchy from this training drill, information for separate members of the platoon were lost. Future work is needed to produce a model that can retain the accuracy of the hierarchy without creating visual clutter. Further development could also expand the model to include more variation of the enemy combatants or the obstacle itself, allowing the model to provide even more data for future training.

References

Giammarco, K., Giles, K., & Whitcomb, C. A. (2017). Comprehensive use case scenario generation: An approach for modeling system of systems behaviors. *2017 12th System of Systems Engineering Conference (SoSE),* 1–6. https://doi.org/10.1109/SYSOSE.2017.7994950

Quartuccio, J., Giammarco, K., & Auguston, M. (2017). Identifying decision patterns using Monterey Phoenix. *2017 12th System of Systems Engineering Conference (SoSE),* 1–6. https://doi.org/10.1109/SYSOSE.2017.7994952

U.S. Marine Corps. (2020). *Marine rifle squad* (MCRP 3-10A.4). https://www.marines.mil/Portals/1/Publications/MCRP%203-10A.4.pdf?ver=xMp-TvUOgArdOOnzhRRfkQ%3d%3d