

Military Simulator-A Case Study

Statistical Intrepretation of data

Shruti Jadon

*Department of Computer Science
Jaypee Institute of Information Technology
Noida, India*

Anubhav Singhal

*Department of Computer Science
Jaypee Institute of Information Technology
Noida, India*

Abstract— In this paper we show how the statistical outcome of combination of Behaviour Tree and Utility Based AI architecture can be used to interpret the probable results of certain situation & designing more realistic platform for Military Simulators. In this work, we have designed a situation based simulator system in unity 3D using RAIN, an addon for Unity developed by Rival Theory, whose outcomes helps in analyzing the results and finding out the various spaces on which our favorable situation might exist, this is done geometrically using data mining algorithms. Along with that a linear & non linear model has been suggested having its parameters determined by using statistical general terms & thus its accuracy being compared to each other & predefined algorithms.

Keywords— Military Simulator, Behaviour Tree, clustering, statistical analysis

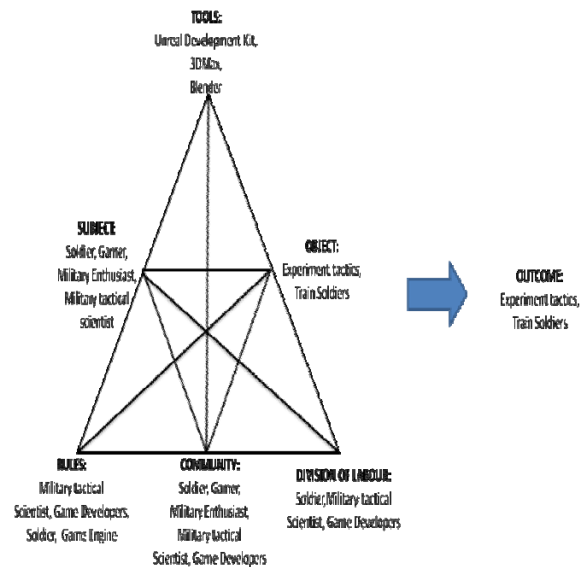
I. INTRODUCTION

Military simulations, war games, are simulations in which different tactics & situations can be tested and refined without the need for actual hostilities and their results can be analyzed by parameterizing numerous variables. Simulations exist in many different forms, with varying degrees of realism. In recent times, the scope of simulations has widened to include not only military but also geo-, political and social factors, which may be seen as inextricably entwined in a realistic warfare model.

For the creation of a realistic model which can collect all results of different human reactions while decision-making, we need a simulator which can give win lose situation based on numerous parameters. As per Chaos theory, even a small parametric change can lead to a lot difference in the result & to make the result more accurate & realistic, it is better to consider as much variable as possible for which we come to need of designing an artificial intelligence (AI) system.

II. SIMULATION ARCHITECTURE

A. Architecture as an activity system



Subject: He is the person who will be using the system.

Object: The objective of the subjects

Outcome: The final outcome of using the system.

Tools: The tools used to build the system.

Rules: The people or system who will decide the rules based on which the system will work.

Community: Any person who affects the system in any way or the system is affected by him.

Division of Labor: People who develop the system.

B. Behavior Tree Architecture

In video games, Artificial Intelligence is used to simulate human-like intelligence, intelligent behaviors or subsets of intelligent behaviors primarily in non-player characters (NPCs). The techniques used typically draw upon existing methods from the field of Artificial Intelligence. However, the term game AI is often used to refer to a broad set of algorithms that also include techniques from control theory, robotics, computer graphics and computer science in general. The AI

algorithm can be described through this Behavior Tree.

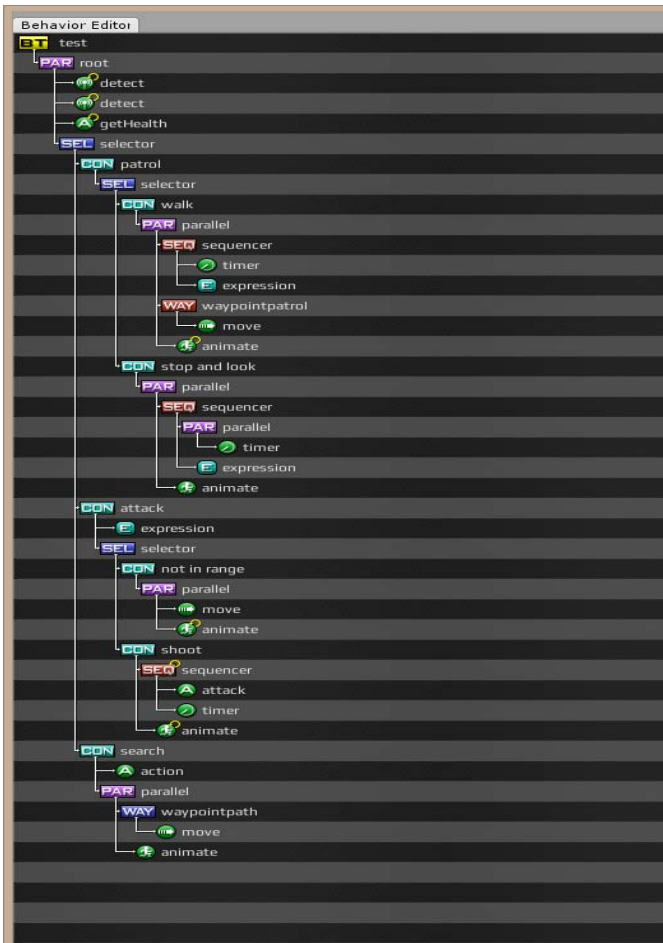


Figure 10: Behavior Tree

The tree consists of three states.

Walk- In this state the player moves along a patrol path continuously and stops and waits for a random time interval between 3-20 seconds and walks again for a random time interval of 3-20 seconds

Attack- In this state there are two sub-states.

Not in Range: In this state the enemy is in visual range but not in weapons range. The player runs towards the enemy in this state.

In Range: In this state the enemy is in weapons range and the player fires on the enemy. The damage is calculated based on various parameters. The equation and parameters used for damage calculation are discussed in the section III Search

If the enemy goes out of visual range, then the player searches the whole terrain randomly for the enemy.

III. DATA ANALYSIS

Many different parameters could be chosen for the analysis; here we have chosen only 13 parameters which are giving values for win-lose situation.

A. Situation plot scheme

In this simulator, we have designed a graphical situation that has 2 teams having equal number of soldiers, and team a is trying to enter in to area of team b, while team b's members are patrolling their paths.

B. Parameters

We have chosen 13 parameters, they are

- Number of attackers & defenders
- Weapon Range
- Visual Range
- Weapon Damage
- Fire Rate
- Walk Speed
- Run Speed
- Search Speed
- Weather
- Gun Condition
- Light condition
- Terrain
- Accuracy of shooting

C. Damage Calculation

Damage done is based on the following parameters.

- Weather
- Terrain
- Gun Condition
- Light condition
- Accuracy of shooting
- Weapon Damage

Weapon Damage is the amount damage a weaon will do if all the other conditions has no effect on the weapon.

The four parameters weather, terrain, gun condition and light condition. These condition affect the damage in the following way.

Parameter value = 0 : No effect

Parameter value < 0 : reduce damage

Parameter value >0 : increase damage

If all the parameters have a value of 50, damage done would be equal to 100, that is, the bullet will kill in one shot.

If all parameters have a value of -50, damage done will be approximately equal to 0.

The equation for damage calculation is:

$$external\ factors = \left| \frac{-50 - weather}{400} + \frac{-50 - guncondition}{400} + \frac{-50 - terrain}{400} + \frac{-50 - light}{400} \right|$$

Accuracy has a range of 0 to 100. A random variable in the range 0-100 is generated each time a bullet is fired. If it is less than or equal to accuracy, only then the bullet will cause any damage based on the above mentioned parameters.

With help of above mentioned parameters and AI, any kind of soldier and weapon can be simulated barring explosives as Area of Effect of ammunition has not been taken into account.

D. Output

All the above parameters are set in the simulator and the simulator is started. Using these parameters the simulator runs the scenario multiple times and gives us the following values:

- Total damage taken
- Ammo used
- Casualties
- Result

If all the soldiers of a team die at the end of simulator, then the result is lost.

The sample data looks like:

E. Tools used:

Unity 3D-Unity is a game development ecosystem: a powerful rendering engine fully integrated with a complete set of intuitive tools and rapid workflows to create interactive 3D and 2D content. It has been used to give simulator visual animation of all the possible results of a particular scenario.

3D max modeling-

3ds Max provides a number of different modeling toolsets and workflows, each with its positive and negative sides. These include Procedural Modeling (parametric objects and non-destructive modifiers), Editable Mesh and Editable Poly explicit modeling, Surface Tools Spline to Patch modeling and NURBS. It has provided the soldier model for the simulator.

IV. STATISTICAL INTERPRETATION

After the data collection 2 data mining algorithms: Rain Forest & Naive Bayes has been applied & analyzed their accuracy based on training set. Along with that a linear model has been created by calculating the parameters, using statistical general terms.

A. Random Forest

Random forests are an ensemble learning method for classification (and regression) that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes output by individual trees.

The accuracy obtained from it is 81.2%

B. Naive Bayes

Naive Bayes is a popular (baseline) method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or ham) with word frequencies as the features. The accuracy obtained from this method is 79.8%

C. Linear Model

A linear model can be determined using linear regression algorithm, as per data the linear relation of weaponRange_a & result_a comes out to be:

$$result_a = 0.0682 \times weaponRange_a + 0.1225$$

and threshold value, which indicates the separation of loss to win condition i.e. if value>threshold then it's a win condition or else otherwise. It comes out to be 0.373,determined using scattered plotting of training set.By applying it on test set we determined the accuracy to be 66.768%

D. Non Linear Model

A non linear model can be determined using as many variables as possible, Its standard equation is:

$$W = px + qy + rz + ixy + jyz + kzx + lxy$$

Where p, q, r, i, j, k, l are real numbers

Similarly, here the chosen 4 variables are weaponRange_a, visualRange_a, firerate_a and weather_a.

Here, by putting the values in the standard equation & determining variables again using the equations determined by linear regression by plotting out the curve in 5 dimensions, one can imagine a prism kind structure, although not close, but the points on the structure lies on the result_a region.

Here, the threshold of the result comes out to be 0.95. By applying it on test set we get accuracy of 54.9222%.

S.no.	Classification algorithm	Accuracy
1.	Rain Forest	81.2%
2.	Naive Bayes	79.8%
3.	Linear model	66.768%
4.	Non linear model	54.922%

Figure 3: tabular comparison of algorithms

V. CONCLUSION

From this analysis we can conclude it's almost impossible to determine all the variables that can affect the result, and even if we find out it is very hectic to formulate it in to one equation & find out its accuracy. We had 26 variables, 13 for each team, & we conclude that random forest gives us the best accurate result, but as the junk of data will increase, it would be better to calculate them individually using linear model & then taking average of all the results.

REFERENCES

- [1] Jadon,S., and Singhal, A. 2014. Military Simulator - A Case Study of Behaviour Tree and Unity based architecture. International Journal of Computer Applications (0975 – 8887)Volume 88 – No.5, February 2014.
- [2] Mark, D., and Dill, K. 2010. Improving AI Decision Modeling Through Utility Theory. Game Developers Conference.
- [3] Laird, J. E. and van Lent, M. 2001. Human-Level AI's Killer Application Interactive Computer Games. AI Magazine Volume 22 Number 2.
- [4] Dill, K. and Mark,D. 2012. Embracing the Dark Art of Mathematical Modeling in AI. Game Developers Conference.
- [5] Dill, K. and Martin, L. 2011. A Game AI Approach to Autonomous Control of Virtual Characters. Interservice/Industry Training, Simulation, and Education Conference (IITSEC).
- [6] Mitgutsch, K. and Weise, M. 2011 Subversive Game Design for Recursive Learning, Singapore. MIT GAMBIT Game Lab. Massachusetts Institute of Technology.
- [7] van Lent, M. and Laird, L. 1999 Developing an Artificial Intelligence Engine, Artificial Intelligence Lab. University of Michigan.
- [8] Mikhalkin, G. 2005. Enumerative tropical algebraic geometry in R2. J. Amer.Math. Soc. 18,no. 2, 313–377.
- [9] Macindoe,O., Pack Kaelbling, L. and Lozano Perez, T. 2012 Assistant Agents for Sequential Planning Problems. Computer Science and Artificial Intelligence Laboratory, MIT.
- [10] Patrik O. Hoyer, "Non-negative Matrix Factorization with Sparseness Constraints", Journal of Machine Learning Research 5, 2004, 1457–1469.
- [11] Egenfeldt-Nielsen, S. and Heide Smith, J. 2003. Playing With Fire-How do computer games affect the player? The Media Council for Children and Young People.
- [12] Wang, S. 2012 Operations Research 3OR. School of Mathematics & Statistics, the University of Western Australia.
- [13] Rival Theory. "Rain User manual", Tutorials.