Probabilistic Graphical Model for TTRPG combat outcome estimation

Manuel Mariani,
manuel.mariani2@studio.unibo.it

April 2021

UNIBO
Course: Fundamentals of Artificial Intelligence - module 3
Professor: Paolo Torroni

1 Introduction

1.1 Objective of the project

The objective of this project is to design and build a Probabilistic Graphical Model for estimating the outcome of combat in a tabletop role-playing game. The resulting software should provide a clear and understandable description of the state of the combat, turn by turn, and must allow the user to define simple queries describing the evidence of a particular instant of the combat, inferring the subsequent evolution of the combat's state given the user-defined evidence.

1.2 Problem's definition

The game

The problem to model is a simplified combat encounter of a tabletop role playing game. In a combat encounter, two factions, composed by multiple characters, face each other in alternating rounds of combat, until one faction is defeated.

The combat is structured in "rounds" and "turns". A round is composed of two phases: first a faction acts and then the other. During a round, each character can act in its turn by attacking an enemy and moving. When all characters have performed their turn, the next round starts.

Dice rolls

The game uses dice to determine the outcome of the characters' actions. The dice used are multi-faced, can be rolled multiple times, summed with a constant and or summed with other dice results. To represent dice formulas, for conciseness, a variation of the Common Dice Notation will be used:

$$NdF + M@B$$

where F is the number of faces on the die, N is the number of dice to be rolled and summed, M is an integer value to be added or subtracted to the result of the dice rolls, and B is the number of "Boons" if B is positive and "Banes" if B is negative.

Boons and Banes are respectively bonuses or maluses applied to a dice roll. If B Boons are present in a roll, $Bd6$ are rolled and the highest value from this set of dice is kept and the other discarded. After determining the highest, the value is added to the roll's result. Banes work in the same manner, but instead of adding the highest result, it is subtracted. This type of roll will be referred ad $Bd6kh$, where the kh means "keep highest". For the project's interest, this mechanic offers a nice and simple way to have interesting probability distributions that are neither uniform nor gaussian.

Characteristics

A faction is composed by multiple characters, each described by a set of attributes:

- **Health** [HP]: the amount of damage a character can endure. If it reaches 0, the character is out of the combat.
- **Attack** [ATK]: attacks against other characters are described by a roll formula. After the roll is done, the result is compared against the target's defense. If the result is higher or equal, the attack succeeds and the Damage can be calculated, otherwise the attack misses and nothing happens. Usually attacks are done using a single twenty sided die and some bonuses or maluses ($1d20 \pm M@B$)
- **Defense** [DEF]: the target number that an opponent must match or surpass with an attack roll for the attack to be successful.
- **Damage** [DMG]: the roll formula used to determine the amount of damage inflicted to a target character, given a successful attack. After the roll is done, the result is subtracted to the target's Health. Usually damage rolls do not use boons and banes ($NdF \pm M$)
- **Number of attacks** [N_{ATK}]: the number of attacks that a character can do in one turn.

In the game, characters can also take a great variety of actions, such as move, heal itself or an ally, cast spells that greatly influence the combat, interact with the environment and do other kinds of special actions. For the sake of brevity and simplicity, these actions have not been considered. In the software, if one wishes to express one of those special actions, the attributes defined in this project can be modified accordingly. For example, if a character can heal itself, increasing its health appropriately can be considered.

2 The model

2.1 Dice modeling

Dice are represented as probability distributions. Each dice formula gets converted into its Probability Mass Function (pmf) to be used in the probabilistic graphical model. To rapidly calculate the pmf, instead of enumerating all possible outcomes of the roll ($NdF + M@B$), convolutions are used:

1. The probability mass function $p_{\text{1dF}}(x)$ of a single die $1dF$ is determined. Since a single die has an uniform probability distribution, it is simply

$$p_{\text{1dF}}(x) = \frac{1}{F} \quad \forall x \in [1, F]$$

2. Summing the result of multiple dice is a sum of independent events, thus we can use convolutions to determine the value of p_{NdF}. To do achive this, p_{1dF} is convolved with itself N times

$$p_{\text{NdF}}(x) = \underbrace{(p_{\text{1dF}} * \cdots * p_{\text{1dF}})}_{\text{N}}(x)$$

3. After the convolutions, the domain of the pmf of N rolls of F-sided dice is $D(p_{\text{NdF}}) = [1, N \times F]$. To add the integer modifier M, the domain and its values can be shifted according to M

$$p_{\text{NdF+M}}(x) = p_{\text{NdF}}(x + M)$$

therefore $D(p_{\text{NdF+M}}) = [1 + M, N \times F + M]$

4. Then, if B boons or banes are present, their probability mass function is calculated:

   (a) Considering $B > 0$, to determine the pmf of $Bd6kh$, we can say that the probability of the total result having a value $X \leq y$ is the probability of all $Bd6$ rolls having a value $X_1 \leq y \wedge \cdots \wedge X_B \leq y$, thus

$$\begin{aligned} P_{\text{Bd6kh}}(X \leq y) &= P_{\text{Bd6}}(X_1 \leq y \wedge \cdots \wedge X_B \leq y) \\ &= \underbrace{P_{\text{1d6}}(X \leq y) \times \cdots \times P_{\text{1d6}}(X \leq y)}_{B} \end{aligned}$$

   (b) So, by defining the Cumulative Density Function of a roll $F_{\text{NdF}}(y) = P_{\text{NdF}}(X \leq y)$, and by exploiting the fact that the rolls are independent the previous equation can be rewritten as

$$F_{\text{Bd6kh}}(y) = \prod_{i=1}^{B} F_{\text{1d6}}(y) = \prod_{i=1}^{B} \frac{y}{6} = \left(\frac{y}{6}\right)^B \quad \forall y \in [1, 6]$$

   (c) Then the Probability Mass Function of $Bd6kh$ can be calculated from the Cumulative Density Function

$$\begin{aligned} p_{\text{Bd6kh}}(y) &= F_{\text{Bd6kh}}(y) - F_{\text{Bd6kh}}(y - 1) \\ &= \left(\frac{y}{6}\right)^B - \left(\frac{y-1}{6}\right)^B \quad \forall y \in [1, 6] \end{aligned}$$

   (d) If B is negative, the distribution is simply "mirrored" from the positive distribution

$$p_{\text{Bd6kh}}(y) = p_{\text{-Bd6kh}}(6 - y + 1)$$

5. Finaly to add or subtract the B boons or banes, given that it is a sum of independent events, a convolution between $p_{\text{NdF+M}}$ and $p_{\text{Bd6kh}}(x)$ is performed, returning the final probability mass function of the roll

$$p_{\text{NdF+M@B}}(x) = (p_{\text{NdF+M}} * p_{\text{Bd6kh}})(x)$$

2.2 Character grouping

To make the graphical model more general as possible, the characters' of each faction get combined into a single one, resulting in only two characters to consider.

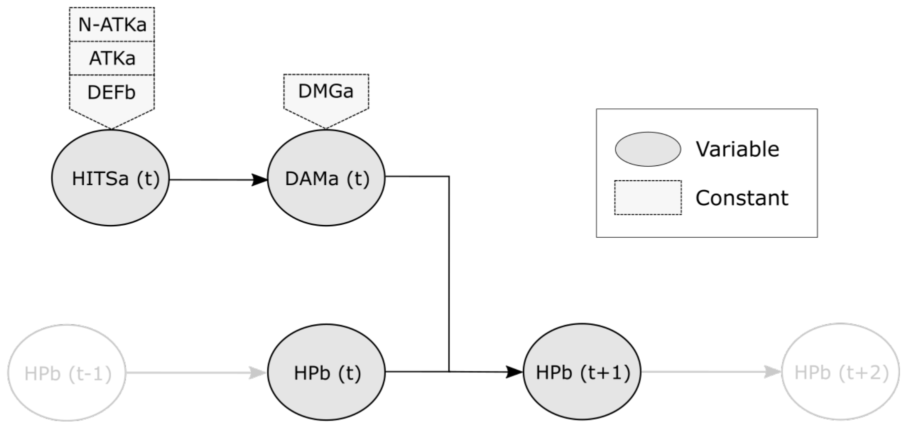
For a faction F composed of characters c, its characteristics are computed with the following formulas:

$$\begin{aligned} HP_F &= \sum_c HP_c \\ DEF_F &= \frac{\sum_c DEF_c \times HP_c}{HP_F} \\ N_{ATK,F} &= \sum_c N_{ATK,c} \end{aligned}$$

For the attack and damage dice, the combining is done by summing the probability distributions of the characters' dice and then normalizing.

2.3 The Bayesian Network

Since we are considering the evolution of a system over time, the chosen Probabilistic Graphical Model for representing the problem was a **Dynamic Bayesian Network**. The network, considering only a single turn of an attacking faction a against a defending faction b, is:



Note that the term "constant" is used to describe a parameter or function that does not depend on a turn t but is dependant on the faction characteristics.

Then in turn $t+1$ the PGM continues with the same network structure, but with b as the attacker and a as the defender. When the two factions have both done their turn, the round is over and the next one starts, repeating the network with a as an attacker and b as a defender

In the network three variables for each faction are used:

- $HITS_f(t)$: Number of successful hits of the faction f against the opposing faction during turn t. It depends on the constants $N_{ATK,f}, ATK_a, DEF_b$.
- $DAM_f(t)$: The damage dealt by faction f to the opposing. It depends on the number of successful hits $HITS_f(t)$ and on DMG_a (constant)
- $HP_f(t)$: Health points of the faction f during turn t. It depends on the damage dealt by the opposing faction o, $DMG_o(t)$ and at the beginning of the combat $HP_f(0) = HP_f$

2.4 Aleatory Variables and Operations

In the model, a custom set of operators is defined to mathematically determine the probability mass functions of a variable A in relation to an other variable B or numeric parameter n:

- Aleatory sum $\oplus$:
$$A \oplus B(i+j) = A(i) * B(j) \quad \forall i \in D(A), j \in D(B)$$
- Aleatory subtraction $\ominus$:
$$A \ominus B(i-j) = A(i) * B(j) \quad \forall i \in D(A), j \in D(B)$$
- Aleatory power $\uparrow$:
$$A \uparrow n = \begin{cases} \underbrace{A * \cdots * A}_{n} & \text{if } n > 0 \\ 1 \text{ for x=0, 0 elsewhere} & \text{if } n = 0 \end{cases}$$
- Aleatory product $\otimes$:
$$A \otimes B = \sum_{j \in D(B)} B(j) \times (A \uparrow j)$$

2.5 Mathematical system

Using the previously defined operators, a mathematical system representing a turn t of combat can be described as follows:

1. First the number of successful hits of faction a against faction b is determined by the probability that an attack done by a has a result equal or higher than the defense of faction b to the power of the number of attacks of a

$$HITS_a(t) = ATK_a(x \geq DEF_b) \uparrow N_{ATK,a}$$

2. Then the amount of damage dealt by faction a can be calculated by multiplying the number of successful hits with the damage distribution of faction a

$$DAM_a(t) = DMG_a \otimes HITS_a(t)$$

3. Finally, the health distribution of faction b at turn t is found by subtracting the damage dealt by faction a to the health of faction b in the previous turn $t-1$

$$HP_b(t) = HP_b(t-1) \ominus DAM_a(t)$$

This system represents the turn taken by faction a attacking faction b. Since in a round the two factions alternate, the turn where b is attacking and a defending is symmetrical to the one described above.

2.5.1 Evidence

Since the system is probabilistic, the distributions representing its evolution in time (rounds) get more spread out as the combat progresses. The Dynamic Bayesian Network developed allows the user to define evidence for each variable of the model during whatever round. During a round, if evidence is present it is used instead of the above-defined calculations, and the relative variable's distribution "collapses" into a single certain value. This is particularly useful in the case of the tool being used in real-time estimation of the combat, where the state of the system is known during each round.

3 Conclusions

3.1 Improvements

Further improvements can be made, in particular the model would be more accurate if multiple characters were represented individually and if the model included more of the possible actions included in the game.

These improvements, however, greatly increase the complexity of the model and the gains in accuracy are hypothesized to be marginal compared to the cost in complexity, considering the aleatory nature of the system.

3.2 Final remarks

This project shows promising results. The tool developed is capable of displaying the estimated evolution of a TTRPG combat encounter in a simple and understandable way, while also providing the feature of evidence for real-time applications.