Chained Strategy Generation: A Technique for Balancing Multiplayer Games Using Model Checking

William Kavanagh Dr Alice Miller

W.Kavanagh.1@research.gla.ac.uk, Alice.Miller@glasgow.ac.uk

University of Glasgow

Abstract: Game balancing is the task of ensuring that a game is both fair to its player and interesting to play. Many games offer players a choice of disparate game material (such as cars, characters or weapons) and ensuring that these materials are all *balanced* is a notoriously complex task. We introduce a new technique called Chained Strategy Generation (CSG) that uses probabilistic model checking and strategy synthesis to model high-level competitive play to represent games being played over a long period of time. We then apply CSG to a case study to show how it can be used to help describe a game that is cyclically balanced, ensuring players have a number of impactful decisions, which will lead to more interesting gameplay.

1 Introduction

An abstraction of a game known as the *metagame* [1], describes the evolving state of play between the strategies and materials offered to players. Games are often designed around a set of game material – such as heroes, cards, vehicles or weapons – which players can choose from at the start of the game. A game is considered well balanced if it has a *healthy* metagame, where a variety of strategies for different materials are popular over a long period of time, with new strategies becoming prominent when they are effective against the current *meta*. In order to achieve this, the ways of playing need to form an intransitive relationship, similar to Rock Paper Scissors. Rock Paper Scissors is frequently used as the foundation for game design where all materials are organised in small cycles with each material unit able to beat the next with a high probability. A relationship where one unit is designed to beat another may make a game seem imbalanced, but as part of a network of similar relationships where all of a unit's drawbacks are compensated by advantages the whole game is balanced. In a system like this quantifying which materials are best is not possible, one must consider what materials are best at a given point in time. Materials are judged based on what strategies are popular (or what is the current meta) and how well they can perform against them.

We have devised a mechanism for representing a full metagame representation which can be analysed to inform design decisions about which materials are too powerful and which are too weak. To do this, we use the probabilistic model checker Prism [3] to generate strategies using strategy synthesis [2]. By repeatedly doing this we inductively define a set of *effective strategies*, a process we call chained strategy generation (CSG). By studying these effective strategies, the materials that are used by them and how well other materials can perform against them we can make judgements on the comparative strengths of the materials. We also identify dominant strategies and dominated material, without the need to consider the full strategy sets of all material units.

2 Methodology

Let **G** be a two-player game between players $\{p1, p2\}$ and M the set of material. We define a strategy in terms of players and material such that $\theta(p, m)$ is a strategy for player p using material $m \in M$. Define a particular strategy $\theta_{naive}(p, M)$ which chooses material randomly with a uniform distribution at the start of the game and chooses actions randomly with a uniform distribution of all actions available. Let $\theta_{p \to p'}$ be a function that maps a strategy for player p into the equivalent strategy for p' and let $P_{win(1)}(\theta(p, m), \theta'(p', m))$ be the probability that strategy $\theta(p, m)$ beats $\theta'(p', m)$.

By modelling the game as a Markov Decision Process where one player has a fixed strategy and the other is represented as having a nondeterministic choice of actions, we can use model checking to generate the strategy with the highest probability of winning against the fixed strategy, i.e. the optimal *counter*. For example, given some strategy $\theta(p, m)$, we can use model checking to find the strategy $\theta'(p', m)$ for which $P_{win(2)}(\theta(p, m), \theta'(p', m))$ is a maximum. This is equivalent to finding the adversary which maximises the probability of reaching a state in which $p2$ wins. We denote the strategy generated by this process $counter_{p'}(\theta(p, m))$.

CSG is described by:

1. **For** $m \in M$:
   $\theta^*(p1, m) := counter_{p1}(\theta_{naive}(p2, M))$
2. *The meta* at iteration 0 (θ^0) is $\theta^*(p1, m)$ for which $P_{win(1)}(\theta^*(p1, m), \theta_{naive}(p2, M))$ is maximum
3. $k := 1$
4. **For** $m \in M$:
   $\theta^*(p1, m) := counter_{p1}(\theta^{(k-1)}_{p1 \to p2})$
5. $\theta^*(p1, m)$ for which $P_{win(1)}(\theta^*(p1, m), \theta^{(k-1)}_{p1 \to p2})$ is maximum is *the meta* at iteration k (θ^k)
6. **If** $\theta^k \neq \theta^j$ for all $j < k$: k++; **Goto** step 4
   Else: **Quit**

CSG terminates under one of two conditions, either a dominant strategy has been identified – a strategy best played against by itself – or a cycle of effective non-dominant strategies have been found. A dominant strategy suggests that a game is poorly balanced and easily solved, players would soon converge upon the dominant strategy and repeated plays of the game are likely to involve both players employing the dominant strategy. In a well-designed game it would take a long time for CSG to terminate and most material units would at some point be used by a meta strategy.

3 Results

We implemented CSG on a simple case-study of a 2-player turn-based game where players choose a pair of characters from a choice of 5: a Knight, an Archer, a Wizard, a Rogue and a Healer. These characters are all qualitatively differentiated. The Archer can target multiple opponents, the Wizard can stun an opponent, preventing them from acting on their next turn, the Rogue can execute low-health opponents and the Healer will increase either their current health value or that of an ally upon successfully attacking an opponent. All characters have have a maximum health value, an accuracy value and a damage value, the healer also has a heal value – the value by which they increase an ally's health – and the rogue also has an execute value – the value at which they can do damage equal to the opponent's current health. These 17 attributes are the *configuration* of the game. A coin is flipped to decide who goes first then players take it in turns to attempt one action from any of their alive characters targetting any of their opponent's alive characters. Our aim is to assess how balanced the materials are for a given configuration and, if needed, to suggest what to change.

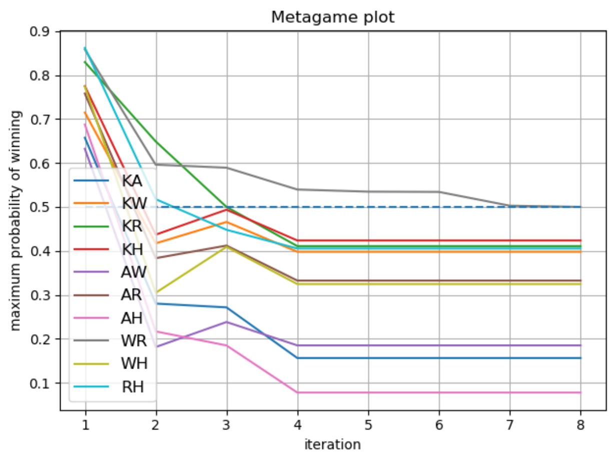

Figure 1: CSG performed on a poorly balanced game.

Character initial	K	A	W	R	H
Health	9	6	7	7	7
Accuracy	0.5	0.85	0.75	0.65	0.7
Damage	4	2	2	3	2
Execute/Heal	-	-	-	6	2

Table 1: A configuration for the case study

One way the results of CSG can be analysed is to plot the probabilities of each material played maximally against the meta strategies. This is shown in fig. 1 for the configuration in table 1. A dominant strategy is clearly identified for a Wizard-Rogue pair (WR). Because our case study uses *teams* of material, we study the aggregate win chance for each character when all 10 pairs play against each other using their final strategies. Ideally these values would be close to 0.5 to indicate that each pair is as likely to win as each other overall. The results for each character are shown in table 2. It is clear that the Archer is too weak and should be made stronger whilst the Rogue is too strong and should be brought more in line with the other characters. We reconfigured the Archer to have 7 health and 0.9 accuracy and the Rogue to have an execute range of 5 and accuracy of 0.6. The result of this change is that average win rates for all characters are more uniform as shown in table 2 and no dominant strategy is identified. This shows how effectively CSG informs game design in spite of the sensitivity of metagame development.

Char. initial	K	A	W	R	H
Former	0.497	0.404	0.5	0.602	0.497
Updated	0.478	0.521	0.484	0.514	0.504

Table 2: Table comparing the aggregate win probabilities for all characters between the two configurations to 3dp.

4 Conclusion

We have shown how CSG allows for analysis of a game's balance and can be used to inform better game configurations. By predicting the direction of the metagame, CSG allows game designers to compare material units in terms of how viable they are throughout the game's lifespan, rather than at a single point in time.

Future work will involved the development of an automated tool that analyses and reconfigures games itself, until it finds the optimal configuration within user defined bounds. This tool has the potential to be highly useful for game development and to furthering understanding of complex game systems. Ultimately, by ensuring the material is developed to be fair, CSG will help designers to make games which are more interesting and more fun to play.

References

[1] M. Debus. Metagames: on the ontology of games outside of games. In *Proceedings of the International Conference on the Foundations of Digital Games, 2017.*

[2] Ruben Giaquinta, Ruth Hoffmann, Murray Ireland, Alice Miller, and Gethin Norman. Strategy synthesis for autonomous agents using PRISM. pages 220–236, 2018.

[3] M. Kwiatkowska, G. Norman, and D. Parker. PRISM 4.0: Verification of probabilistic real-time systems. In *Proc. Int. Conf. Computer Aided Verification (CAV'11).*