Using Multi-Arm Bandits to Optimize Game Play Metrics and Effective Game Design

Kenny Raharjo and Ramon Lawrence

***Abstract*—** Game designers have the challenging task of building games that engage players to spend their time and money on the game. There are an infinite number of game variations and design choices, and it is hard to systematically determine game design choices that will have positive experiences for players. In this work, we demonstrate how multi-arm bandits can be used to automatically explore game design variations to achieve improved player metrics. The advantage of multi-arm bandits is that they allow for continuous experimentation and variation, intrinsically converge to the best solution, and require no special infrastructure to use beyond allowing minor game variations to be deployed to users for evaluation. A user study confirms that applying multi-arm bandits was successful in determining the preferred game variation with highest play time metrics and can be a useful technique in a game designer's toolkit.

***Keywords*—**Game design, multi-arm bandit, design exploration and data mining, player metric optimization and analytics

I. Introduction

The game industry is a multi-billion dollar annual industry that combines software development with artistic creativity. Although there are known templates and best practices for game design, building a game that is a hit still has considerable challenges and often requires luck, as it is not easy to pre-determine player response to a game. Game designers use extensive testing, betas, player feedback, and a variety of other mechanisms to understand what to improve and if players will enjoy a game. It would be extremely valuable to have a technique to systematically explore some design choices to help settle on effective game designs. These design choices may be minor user interface changes such as colors or button placement to more dramatic changes in game play and rules.

Multi-arm bandits [1] were developed to solve the exploration-versus-exploitation problem where the goal is to use knowledge gained by exploring the space to make better choices in the future. They have found applications in medical trials, web site design, online marketing, and other decision support areas. In the online marketing space, multi-arm bandits have been applied to optimize user click-through rates by finding better advertisement placements. In many ways, games, especially games with in-app purchases, have a similar business model as online content delivery and advertising. The goal is to engage users to interact and pay for content, and it is critical to make design decisions that optimize for increased revenue. Further, it is possible to dynamically change a game via updates and downloadable content much more easily than in the past.

In this work, we modify an existing game to have multiple design variations and use a multi-arm bandit algorithm to determine the optimal variation based on user play time. The multi-arm bandit algorithm determines the variation given to each user when they start the game and over time will converge to the best variation. In a user trial, the algorithm was successful in determining the optimal variation in a very short period of time, and this successful result demonstrates the applicability of multi-arm bandits to game design.

The organization of this paper is as follows. Section 2 presents a short background on game design and multi-arm bandits. Section 3 describes the game used for the experimentation and how the user evaluation experiment was performed. Section 4 contains experimental results. The paper closes with future work and conclusions.

II. Background

The competitive game industry requires complex algorithms and critical choices to ensure that the latest games capture the attention of demanding consumers. According to the Entertainment Software Association (ESA), there are 150 million Americans that play games of which 42 percent play three or more hours per week [2]. If a game does not appeal to a player, or becomes repetitive, they will stop playing it entirely. Different people have their own preferences that developers often have limited knowledge about. To insure players like a game and keep playing, it is valuable to dynamically change game parameters to suit the users' taste and improve user experience.

Game analytics is the collection and analysis of game data for the purpose of discovering and understanding patterns in user interactions with the game. This analysis may be exploited to resolve issues, develop additional content, or increase revenue by encouraging more in-app purchases or advertisement clicks. Game monetization with analytics is as important as the game itself. For the game industry, it is crucial to find ways to develop games which appeal to the target audience, minimize frustration and keep their interest for as long as possible. Analysts model a player profile by their game time, spending habits, interaction with other players and game progression. The data allow analysts to recommend design

Kenny Raharjo and Ramon Lawrence are with the Department of Computer Science at the University of British Columbia, Canada. phone: 250-807-9390; e-mail: ramon.lawrence@ubc.ca.

choices to maximize retention rate. Analysts will also want to study trends or features which will affect the conversion rate of non-paying users to paying users.

The multi-armed bandit [1] is a statistical model which balances exploration and exploitation in order to solve recurring decision problems. The common real-world metaphor for this problem involves a gambler facing a collection of slot machines ("bandits") at a casino, with each machine having an “arm” to pull. Each machine has an unknown distribution and expected payout, and the goal is to select arms that maximize winnings through a sequence of lever pulls. After each attempt, the gambler must decide which slot machine to play given current knowledge about their payouts. The multi-armed bandit is a popular choice for experimental design since after each action is taken, valuable information is obtained about that action, and this data will be used as feedback to the algorithm to further improve its upcoming actions.

Formally, a multi-arm bandit problem is defined with k arms where each has a reward distribution $(v_1, v_2, \ldots, v_k)$. The reward distribution may be fixed or some other probability distribution. For example, a machine could payoff $2 every time (fixed) or have a payoff that is a normal distribution with mean value of $1. The fundamental challenge of the problem is that the player does not know the distributions of the arms *a priori*, so they must strategically play machines to both maximize reward while discovering information about arm payoffs. At a given time t, the player picks arm i and obtains a reward from distribution v_i.

A multi-arm bandit algorithm is a technique for selecting a given arm at each time t using the knowledge of past selections to determine the next choice. One of the most common algorithms is epsilon-greedy (ε-greedy [3]) that selects an arm by being greedy and selecting the best arm (1- ε)% of the time, and exploring by selecting a random arm ε% of the time. This algorithm requires no knowledge of the reward distributions and is straightforward to implement. One of the challenges is that even if the algorithm converges to the best solution, it will still select randomly ε% of the time. There are techniques to decay ε over time [4], but they are not used in this work.

Other algorithm variants include Upper Confidence Bound (UCB) strategies [5]. There are also algorithms that handle context [6] and distributions changing over time [7] (i.e. restless bandits). Contextual bandits make an arm selection based on prior selections and contextual knowledge of the domain. In online advertising, knowing a user's age, gender, browser, and demographics should influence the advertisement selections made. A contextual bandit uses this information to inform its decision.

Unlike A/B testing, which requires constant supervision and modifications, multi-arm bandit algorithms are adaptive and perform continuous optimization without further input or modification. Multi-arm bandits support any number of variations and will eventually converge on winning variations as more data is collected.

To our knowledge, this is the first application of multi-arm bandit algorithms to dynamically modify game design. There has been previous work in applying multi-arm bandits for selecting moves in a game [8] and searching game trees [9]. These approaches are from the player perspective and are not targeted to the design and modification of the game itself. There is also considerable literature in adaptive and dynamic games including interactive story telling [10] and dynamically changing game difficulty based on user performance characteristics and adaptive AI techniques [11].

III. Experimental Setup

This study applies the multi-armed bandit approach to select the best game design for an open-source Java game called Diamond Hunter [12]. The gameplay involves traversing a maze and collecting all the diamond pieces. Initially, there are obstacles in the map such as shrubs and lakes which will prevent the player from getting to hard to reach areas containing diamonds. As the player progresses through the game, they are able to find tools such as an axe or a boat to overcome these obstacles. The game is short and simple, and players can complete it in approximately 4 to 7 minutes.

The experimental study asks the participant to play the game followed by an optional online survey. Volunteers were notified about the game via email and social media. The participant is first given a URL to an online consent form which describes the study procedures and disclaimers. Once the participant agrees to participate, they are redirected to the game page, which contains simple instructions on how to play the game. The player has the option to abort the game anytime.

When the player starts the game, the application connects to an online database to retrieve configuration information and then executes the multi-arm bandit algorithm to select a variant (game theme) which the user will play. The player will continue playing until completing the game or closing/exiting the application. Game data, including time played, is then sent to the database. A player will only experience one game theme in a session. After enjoying the game, the player then may complete a short survey to provide feedback.

In this experiment, the "arms" (or choices) of the multi-arm bandit are three different game background themes (or "skins"). The difference between the variants is mainly aesthetic, gameplay and difficulty remains unaltered. The original game has a nature theme (referred to as the "Forest" theme), with trees as permanent obstacles and shrubs as temporary obstacles. Two other themes were created: a "Winter" theme and a fiery "Volcanic" theme. The metric to be optimized is game play time.

The multi-arm bandit algorithm used is the epsilon-greedy algorithm. Since there is no prior data, the first few players will receive a random theme in order to obtain some knowledge about each theme choice. After the initial random, explore phase, the algorithm performs ε-greedy (with $\varepsilon = 0.15$) to pick a background variant to use. The game selects a background at random (explore) with probability 0.15 and selects the variant with the maximum average game play time (exploit) with probability of 0.85. Note that when selecting the multi-armed bandit algorithm to use, the UCB algorithm was not selected as it requires a good understanding of the underlying distribution.

Fig. 1 Diamond Hunter Themes

IV. Experimental Results

Over a period of 20 days, 42 participants volunteered to play the game and take part in the study. Figure 2 displays the number of times each game variant was selected by the multi-arm bandit as the number of players increased. Since no prior data was given to the algorithm on each design, the algorithm selected each design once early in the process to get initial data. It then proceeded to provide game variants to each new user in the study based on its expectation of the best variant.

As shown in Figure 2, the multi-armed bandit determined the Forest theme as the optimal choice, which is selected 30 times out of 42 trials. Even though the other variants were explored, convergence to the optimal choice is very fast. Per game data shown in Table I justifies this choice as the average play time is highest for the Forest theme. Since the variants only differ aesthetically without any difference in difficulty, the average time taken to complete the game should be fairly similar. The average time played is approximately 5 to 6 minutes, which falls within the expected length of 4 to 7 minutes.

The algorithm successfully handles data with extremely large variance amongst playtime, which occurs due to players dropping from the game immediately and some players taking quite long to complete the game. Even though the maximum recorded play time is 13 minutes for the Volcanic theme, its poor performance on subsequent plays demonstrates that it is not the best choice. The abandonment rate was 7% for the Forest theme, 0% for the Winter theme, and 50% for the Volcanic theme. The completion rate was 80% for the Forest theme, 71% for the Winter theme, and 50% for the Volcanic theme. The Volcanic theme times may have been longer as the interface elements have less contrast than the other themes so a player trying to complete the mazes may take longer. The higher abandonment rate for the Volcanic theme also demonstrates that it is not a suitable choice. The multi-arm bandit correctly detected this very quickly and did not provide that theme many times to players.

Table I. Players Metrics by Theme

Theme	Min. Time	Max. Time	Avg. Time
Forest	0.20	11.81	5.68
Winter	3.20	8.02	5.46
Volcanic	0.16	13.39	4.29

Fifteen out of the 42 volunteers completed the survey. 21% of the participants who provided feedback on the game length found it too short and the remaining 79% found the game length just right. Since no players complained about the game being too long, there is a possibility to increase the game length to better test abandonment rate. Several comments were received regarding the game audio being too loud and would recommend an option to reduce it.

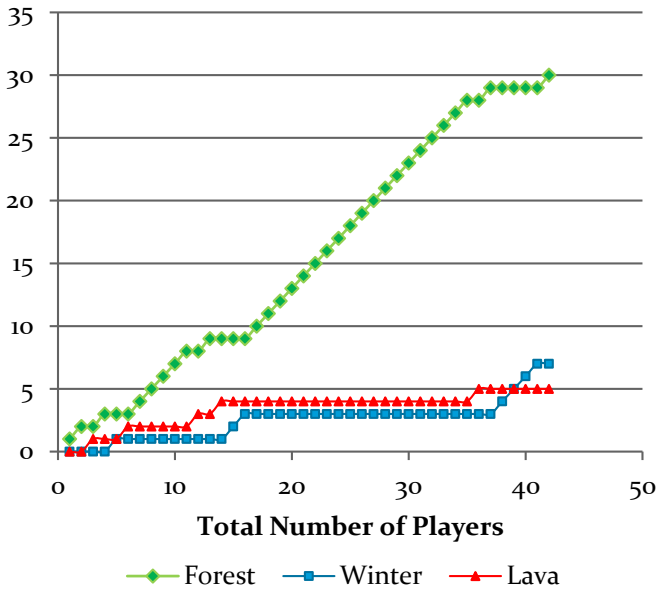

Fig. 2 Game themes selected over time as more players play the game

V. Future Work and Conclusions

This work is the first application of multi-arm bandit algorithms to game design with the goal of increasing user enjoyment and metrics. Multi-arm bandits allow a game designer to easily test various design variations in real-time and have the algorithm converge on selecting the best designs to maximize revenue and player metrics. The experimental results demonstrate that it is possible to dynamically modify a game using a multi-arm bandit algorithm selecting variants, and this modification can quickly converge on optimal designs while helping designers understand what works and what does not. This convergence requires very little data (players). The ability to modify a game as it is being played to improve player metrics allows for easier experimentation of different designs and more efficient and systematic approaches to selecting the best design. The multi-arm bandit algorithm helps a designer chose between many different design variants without having to run complicated user experiments and testing.

Since the application of multi-armed bandits on game design and development is a new topic, there are many avenues for further research. One of the most promising areas is for in-app purchasing and advertising where even minor design and aesthetic changes can have a dramatic effect on revenue and success. Using this multi-arm bandit technique and evaluating in this space would be extremely valuable. For example, an app could experiment with the location for advertisements to achieve the highest click rate. With many games having millions of players, this data can be collected using only a small fraction of players, and then the optimal design chosen.

It would also be interesting to investigate how different variants can be provided to gamers of different abilities and demographics. Contextual multi-arm bandit algorithms allow for identifying optimal arms based on user characteristics (age, skill level, gender). Deploying such a system would allow the game variant to adapt to the user as certain design choices may be more effective based on user characteristics. Recognizing and adapting to player differences, especially player gender, can dramatically improve player satisfaction as it is well-known that males and females have different motivations and engagement with games.

Finally, investigation of other multi-arm bandit algorithms including UCB and others in a variety of application domains would allow for building a general testing framework that can be easily used by game designers. A longer term vision is to construct this testing framework to allow game developers to use multi-arm bandits for their game design activities.

References

[1] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, no. 1, pp. 4-22, 1985.

[2] Entertainment Software Association, "Industry Facts", Retrieved August 2016 from http://www.theesa.com/about-esa/industry-facts

[3] C. J. Watkins, "Learning from delayed rewards", Ph.D. thesis, University of Cambridge, 1989.

[4] J. Vermorel and M. Mohri, "Multi-armed bandit algorithms an empirical evaluation,", *European Conference on Machine Learning*, Springer, pp. 437-448, 2005.

[5] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis for the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2-3, pp. 235-256, 2002.

[6] L. Zhou, "A survey on contextual multi-armed bandits", *arXiv preprint arXiv:1508.00326.*

[7] G. Burtini, J. Loeppky, and R. Lawrence, "Improving online marketing experiments with drifting multi-armed bandits,", *ICEIS 2015 – 17th International Conference on Enterprise Information Systems*, pp. 630-626, 2015.

[8] A. J. Ramirez and V. Bulitko, "Automated Planning and Player Modeling for Interactive Storytelling," *IEEE Transactions on Computer Intelligence and AI in Games*, vol. 7, no. 4, pp. 275-286, 2015.

[9] C. H. Tan, K. C. Tan, and A. Tay, "Dynamic Game Difficulty Scaling Using Adaptive Behavior-Based AI," *IEEE Transactions on Computer Intelligence and AI in Games*, vol. 3, no. 4, pp. 289-301, 2011.

[10] A. Garivier, E. Kaufmann, W. M. Koolen, "Maximin Action Identification: A new Bandit Framework for Games,", 29th Annual Conference on Learning Theory, pp. 1028–1050, 2016.

[11] S. Ontañón, "The Combinatorial Multi-Armed Bandit Problems and Its Application to Real-Time Strategy Games,", AIIDE 2013 - Ninth Artificial Intelligence and Interactive Digital Entertainment Conference, pp. 58-64, 2013.

[12] Diamond Hunter. Retrieved August 2016 from https://www.youtube.com/watch?v=AA1XpWHhxw0