Inteligencia Artificial, 24(68) (2021), 33-36
doi: 10.4114/intartif.vol24iss68pp33-36

INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

Procedural Content Generation for General Video Game Level Generation

Adeel Zafar[1*], Hasan Mujtaba[2**], and Omer Beg[3**]

[*]Department of Cyber Security and Data Science, Riphah International University, Islamabad, Pakistan
[**]Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, Pakistan

Abstract. With the passage of time, video games are becoming more complex, and their development incurs greater time and cost. The creation of video gaming content such as levels, maps, textures and so on represent a large part of the overall cost of game development. Procedural Content Generation (PCG) is a method of generating content via a pseudo-random process. Level generation has been the most significant and oldest problem in the PCG domain. The majority of the PCG level generators are specific to a particular game, content is generated only for a suited single type and these generators are evaluated mostly by computational metrics, user studies and fitness functions. Considering, the grand goal of general Artificial Intelligence, it would be beneficial to sculpt solutions that are applicable to a general set of problems. For the level generation problem, this can be achieved by constructing a level generator that generates levels for a set of games and not explicitly for a single game. In this research, we have created four different types of generators for the GVG-LG framework. The generators follow a distinct path and are able to solve multiple problems related to PCG including dynamic difficulty adjustment, creation of intelligent controllers, creating aesthetically appealing levels and using patterns as objectives for level generation. In addition, we evaluated all the generators using a variety of techniques. The experimental results show promising results and represent our attempt at general video game level generation.

Keywords: Procedural Content Generation in Video Games, General Video Game Level Generation, Computational Intelligence, Machine Learning, Deceptive Levels, FI2POP Genetic Algorithm, Evaluating Game Levels.

1 Introduction

Artificial Intelligence (AI) has been defined as "the study and design of intelligent agents" [1]. Games and AI have a long history, AI has been used to create different game playing agents (Deep Blue, Chinook, and Blondie24) [4]. The creation of controllers for board games also reached a milestone when controllers were developed for the game of *GO* in 2016. In addition to board games, video games are an important domain for AI research. Digital games is a large industry. In 2017, 118 million people downloaded games in the United States which generated a revenue of 3.88 billion dollars in that year [2]. However, the game industry is challenged by problems of resource management and cost. Typically, the cost is related to the game engine and content [3]. In video games, AI has also been used to develop different game playing

ISSN: 1137-3601 (print), 1988-3064 (on-line)

agents. An important milestone in this regard is the Google DeepMind agent that learned to play several Atari 2600 video games.

Another important usage of AI is Procedural Content Generation (PCG). PCG was started in early the 80s, where some game designer generated their game content algorithmically with less human intervention. Since then PCG has been used in both commercial and academic settings. In [4], the authors have also listed other examples of AI and games including player modeling and analyzing game data. This work particularly focuses on PCG in video games. In the next section, we would briefly explain the background and context of the thesis.

2 Background and Context

2.1 Procedural Content Generation

PCG is the technique of generating gaming content including levels [5][6], racing tracks [7], terrains [8] etc. automatically through some pseudo-random process. The early usage of PCG was seen in 1980s in *Rogue* [10]. During the early years of PCG, it was mostly used to overcome the storage limitations. There are different algorithms for generation of game content. In the literature, these techniques are categorized as constructive [10], search-based [9] and machine learning techniques [11]. PCG is a very scattered field. However, long term visions of PCG include creation of high quality content that fits together, making PCG a central part of game-play and generating complete games from scratch. These long term goals involve some immediate steps and one of these steps is the creation of general content generators that can generate content for any game. In the next section, we describe this vision in accordance with our research questions.

3 Research Questions

AI methods can be used to design or generate parts of games. From the literature, it is evident that the game AI community is fast growing with large number of studies published each year that focus on game playing agents and content generation. However, the problem with most of these studies is that they are specific to a certain game. Most studies describe a common method or a comparison of two or more methods for generating content in a single game. Practically, these studies could not yield the maximum benefit. In order to take advantage of AI methods, it is better to generalize methods that can be applied to a number of scenarios. For game playing, most of the methods are generalized and a certain algorithm can play a bunch of games in a certain domain. However, the focus on game playing alone is too narrow. Therefore, designing methods that can generate content for variety of games can lead to more interesting and high value solutions. This perspective is mostly referred as general video game level generation: level generators that can generate levels for any game within a specific domain. To solve such a problem, different methods are needed that can capture the maximum level design space. To address these issues following are the central research questions of our study: **RQ1-** Can the existing framework for general video game level generation be improved? **RQ2-** If the answer to RQ-1 is yes. What type of methods or techniques can be used for improving the existing framework? **RQ3-** How can AI be used to generate content for improving the user engagement or player satisfaction? **RQ4-** How can generated content be better evaluated in perspective of general video game level generation? **RQ5-** Can we use these methods to evaluate our own techniques of general video game level generation?

A general video game level generation framework [12] has been built within the General Video Game AI (GVG-AI) framework. This track or framework allows researchers to develop their own generators that can generate levels for a particular set of unseen games. The generators are supplied with video game descriptions and the output is evaluated by human judges. Initially, the framework comes with three sample level generators: Random, Search-based and Constructive Level Generators. The application of this work is on the General Video Game Level Generation (GVG-LG) track.

We initially carried out a detailed literature review to highlight and identify techniques for content generation. Through out investigation, we identified that search-based techniques and constructive techniques are dominating methods used for content generation. Therefore, we have focused on both these techniques for content generation. Search-based techniques can cover a large design space, therefore, it is justified to use them as a solution to our problems. In addition, constructive methods are fast and are widely used in industry settings, therefore their application also poses interesting solutions. The research presents different constructive and search-based methods that generate content for unseen games. Each method proposed in this research solves different problems including dynamic difficulty adjustment, generating complex levels, generating aesthetically rational levels and using patterns as objectives for content generation. Some of the questions were focused on the evaluation of generated content.

Finally, there are three major outcomes of our research work: constructing one or more competent level generators, evaluating the levels generated by each generator, and creating a framework for general video game level generation.

4 Dissertation Contributions

The research presented in this work summarizes the thesis *Procedural Content Generation for General Video Game Level Generation*. The contributions of our work are as follows:

- We propose a novel constructive level generator for the general video game level generation which uses an effective approach for dynamic difficulty adjustment [13]. In this proposed technique, n-gram (a machine learning technique) is used for level aesthetics and constraints are used to make levels playable. For the evaluation, a user-based study is performed. The results of the study indicated that players preferred our generator in contrast to other sample level generators. In addition, a general video game evaluation mechanism was also performed on a set of 20 levels generated by the n-gram and constraint-based generator. The experiment indicated better performance of intelligent agents on our generated levels. The generator was also submitted to the 2017 GVG-LG competition, however, the competition did not take place due to a single entry.
- We developed a deceptive generator that generated three different types of traps including greedy, smoothness and generality trap [14]. The purpose of this generator was to challenge AI in robust conditions. The generator was successful in generating levels for a large set of games in the GVG-LG framework. Our experimental results show that all tested agents are vulnerable to several kinds of deceptions.
- We present a novel search-based general level generator (SBGen) that uses the Feasible In-feasible two population (FI2POP) Genetic Algorithm [16]. Where the fitness function of the algorithm focuses on more subjective factors. Results indicated that the generator demonstrated adequate performance for each type of game including density, balance, symmetry and reachability.
- A pattern-based level generator that uses design patterns as objectives for general video game level generation [15]. In this approach, the fitness function promoted multi-line or multiple patterns and penalized levels with single or isolated elements. The generator generates levels using two different types of fitness functions i.e. α and β. α levels are generated using the actual fitness function and β levels via the inverse of the pattern-based fitness function. Experimentation indicated better performance of good controllers on our α generated levels. The generator also finished up at the third spot in the 2018 GVG-LG competition.

5 Research Limitations

The implementation of our work is mostly on the general video game level generation framework. The track hosts different competitions each year. But we do not consider our work as a detailed usage of PCG in industry settings. It would be very interesting to adopt this research in a more commercial settings.

6 Acknowledgments

This paper is a summarized version of the thesis *Procedural Content Generation for General Video Game Level Generation.*

References

[1] Stuart J Russel and Peter Norvig. Artificial intelligence: a modern approach. *In Pearson Education Limited London*, 2016.

[2] The Statistics Portal, "Breakdown of U.S. computer and video game sales from 2009 to 2017, by delivery format". https://www.statista.com/statistics/190225/digital-and-physical-game-sales-in-the-us-since-2009/ (accessed November 26, 2018).

[3] Takatsuki, Yo. "Cost headache for game developers." http://news.bbc.co.uk/2/hi/business/7151961.stm (accessed November 26, 2018).

[4] Georgios N Yannakakis and Julian Togelius. Artificial intelligence and games. *In Springer*, 2018.

[5] Steve Dahlskog and Julian Togelius. A multi-level level generator. *In Proceedings of 2014 IEEE Conference on Computational Intelligence and Games*, pages 1-8, 2014.

[6] Diaz-Furlong Hector Adrian and Solis-Gonzalez Cosio Ana Luisa. An approach to level design using procedural content generation and difficulty curves. *In Proceedings of 2013 IEEE Conference on Computational Inteligence in Games (CIG)*, pages 1-8, 2013.

[7] Markus Kemmerling and Mike Preuss. Automatic adaptation to generated content via car setup optimization in torcs. *In Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*, pages 131-138, 2010.

[8] Miguel Frade, Francisco Fernandez de Vega and Carlos Cotta. Automatic evolution of programs for procedural generation of terrains for video games. *In Soft Computing volume 16*, pages 1893-1914, 2012.

[9] Julian Togelius, Georgios N. Yannakakis, Kenneth O. Stanley, and Cameron Browne. Search-based procedural content generation: A taxonomy and survey. *In IEEE Transactions on Computational Intelligence and AI in Games volume 3*, pages 172-186, 2011.

[10] Noor Shaker, Julian Togelius, and Mark J. Nelson. Procedural content generation in games. *In Springer*, 2016.

[11] Adam Summerville, Sam Snodgrass, Matthew Guzdial, Christoffer Holmgård, Amy K. Hoover, Aaron Isaksen, Andy Nealen, and Julian Togelius. Procedural content generation via machine learning (pcgml). *In IEEE Transactions on Games volume 10*, pages 257-270, 2018.

[12] Ahmed Khalifa, Diego Perez-Liebana, Simon M. Lucas, and Julian Togelius. General video game level generation. *In Proceedings of the Genetic and Evolutionary Computation Conference, ACM*, pages 253-259, 2016.

[13] Adeel Zafar, Hasan Mujtaba, Sohrab Ashiq, and Mirza Omer Beg. A Constructive Approach for General Video Game Level Generation. *In 2019 11th Computer Science and Electronic Engineering (CEEC), IEEE*, pages 102-107, 2019.

[14] Adeel Zafar, Hasan Mujtaba, Mirza Omer Beg, and Sajid Ali. Deceptive Level Generator. *In AIIDE Workshops*, 2018.

[15] Adeel Zafar, Hasan Mujtaba, Mirza Tauseef Baig, and Mirza Omer Beg. Using patterns as objectives for general video game level generation. *In ICGA Journal volume 41 issue 2*, pages 66-77, 2019.

[16] Adeel Zafar, Hasan Mujtaba, and Mirza Omer Beg. Search-based procedural content generation for gvg-lg. *In Applied Soft Computing volume 86*, pages 1-8, 2020.