

Red Flower Hell: a Minecraft MALMÖ Challenge to Support Introductory Programming Courses*

Norbert Bátfai, Tünde Tutor, Zoltán Bartha,
András Czanik, Máté Szabó

Faculty of Informatics, University of Debrecen, Hungary
batfai.norbert@inf.unideb.hu
tunde.tutor@gmail.com
barthazoli00@gmail.com
czandris98@gmail.com
szabo.mate@inf.unideb.hu

*Proceedings of the 1st Conference on Information Technology and Data Science
Debrecen, Hungary, November 6–8, 2020
published at <http://ceur-ws.org>*

Abstract

The Red Flower Hell challenge is designed for undergraduate courses, such as introductory programming or artificial intelligence, in which students can learn in a competitive way. The name Red Flower Hell comes from the objective of the challenge because the agent programs to be developed must collect as many red flowers as possible in a battle royale-like gorge in Minecraft created directly for this challenge. The battle royale feature is given by that lava flows down the hillsides in the gorge. In the first round of the challenge, agents only have to deal with lava. In the second round, agents also have to fight various Minecraft monsters, such as zombies or spiders. The agent versus agent fight has not yet been implemented. We have used Project MALMÖ for implementing Red Flower Hell. It is a Minecraft mod created by Microsoft for researching artificial general intelligence. The Red Flower Hell challenge was tested at a High-level programming language course at

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

*This work was supported by the construction EFOP-3.6.3-VEKOP-16-2017-00002. The project was supported by the European Union, co-financed by the European Social Fund.

the University of Debrecen in the spring semester of the 2019/2020 academic year. We present these experiences in this paper.

Keywords: Minecraft MALMÖ, programming challenge, education of programming, agent programming, artificial intelligence

1. Introduction

Today, it is typical that leading tech companies have their own artificial general intelligence (AGI) research platform, which is organized around a given particular well-known game. Such a platform of Microsoft is based on the Minecraft game, this is the Minecraft MALMÖ project [11]. The goal with integrating this to the High-level programming course was to modernize teaching programming languages and bring them closer to undergraduate students.

1.1. Introduction to High-level Programming Course

We would like to share our experiences about the High-level programming course at the University of Debrecen. The main thread of this course was the writing of the yearbook of programmers, where students had to complete tasks and write about it in the book. At the end of each chapter there was a Minecraft MALMÖ related task, so it was an important part in the course's requirements. Beside the tasks in the book, there were optional ones, which meant extra experience points to the students, and these points rewarded benefits to completing the course, so it motivated the students to get known with Minecraft MALMÖ. This led to a continuous learning and then to competitions. Some optional tasks were easy because their goal was to teach something new about MALMÖ agents. For example, the first week was about how to move with the character and after that how to query data about what our character sees. There were many competitive tasks like phases of the Red Flower Hell competition detailed in Section 4. We participated in the Human Intelligence competition, where we compared our effectiveness in special tasks, and the Red Flower Hell III. and IV. too, where students watched live streams featuring their agents. These competitions and streams differ from classic assignments, and that is why students enjoyed it.

Our experience is that MALMÖ related tasks helped first year university students to catch up with programming and Python language. It could be helpful for Introduction to programming courses too as we had to use algorithmic thinking, functions and methods and the results are visualized immediately.

1.2. Minecraft and AGI Research

The MALMÖ AGI research platform is also evolving through competitions with renowned universities and companies [9, 10, 16]. But our esports department called DEAC-Hackers also has a Minecraft-MALMÖ division [2].

MALMÖ is the first and only Minecraft mod Microsoft ever released, and its aim was to support artificial intelligence research, but there are many companies, where gaming and AI research are linked together. This tendency can be said to be due to the fact that the mechanics and world of games provide a very good basis for these researches and there is no need to create a separate environment for them, because in many cases everything is given with the world of a game and its characters. The world of Minecraft simulates real world elements. For example, our character can be damaged, feel hunger, move, craft objects and it can create and solve tasks. We could compare the intelligent agents' and human intelligence's result with this platform, and this is why we were interested in Human Intelligence competition, where we examined a problem defined in the Red Flower Hell.

There are other companies like Google's Deepmind Technologies, who used Artificial Intelligence in gaming. They developed the AI named AlphaStar [19], which played Blizzard Entertainment's Starcraft II real time strategy game. The challenge for this AI was given as this game is competitive and uses ranking system to compare players based on their skills. AlphaStar reached the top rank, Grand Master, so its skill was better than 99.8% of players. This result shows that a properly trained AI can perform better on specific tasks than humans. However this was not the first case, as earlier DeepMind's AlphaGo [17] defeated the Go world champion and other artificial intelligences can play for example chess well enough. If we think about complexity, both Starcraft II and Minecraft are much more complex than chess, as in a real time strategy video game, the variation of steps we can make is nearly limitless, and in a sandbox game, we can do almost anything.

2. Related Work

This paper can be placed in many topics, like teaching with competition, AGI research, gamification, machine learning. Teaching any kind of subject with competition is common in education, because it improves the competitive spirit, innovative thinking and teamwork skills of students [13, 15]. In Diansheng Chen and Zhen Li and Tianmiao Wang's paper [4], the students' feedback showed that they preferred competition based learning instead of project based learning. Their competition's task was to navigate a robot through a maze. Artificial general intelligence is a constantly evolving area with many opinions about how it will work [1, 6, 7] and what aspects has [20]. As these papers describe, AGI is only a conception today, but it is sure, that it will be trained on specific tasks. One of these could be navigation and collecting objects, just like in our paper. Gamification appears in nearly every discipline, as it bring closer gaming to science, or in our case, gaming to teaching. As our students are learning computer science, they are mostly familiar with video games. In the recent years, many universities experimented about involving gamification to teaching programming [3, 8], nursing [5], or any other profession [12, 18]. They propose to introduce this method in a progressive way, as teachers has to analyze the users and the environment, then define the learning

objectives, design the milestones, and identify resources. After that, gamification elements can be applied. In Ibrahim Yildirim’s paper [21], we can read about the positive impact of gamification on students’ achievement and their attitudes toward the lessons.

3. Red Flower Hell

The Red Flower Hell (RFH) is a MALMÖ project-based, battle royale-like gorge in Minecraft, shown in Figure 1, designed for undergraduate courses, such as introductory programming or artificial intelligence to learn in a competitive way. The XML file defining our gorge-like game world can be found on the project repository at <https://github.com/nbatfai/RedFlowerHell>.



Figure 1. The base task is to collect as many red flowers as possible before the lava flowing down the hillside has reached the agent.

3.1. Human Intelligence Agents

Before we start programming AI or simple heuristic agents, it is interesting to know how many flowers a human player can collect. This question is answered by Human Intelligence (HI) agents. Human Intelligence agents mean that there is a human player, who tries to collect as many red flowers as possible in the Red Flower Hell. The current record is 46 flowers.

3.2. Software Agents for Introductory Programming Courses

These agents are based only on simple heuristic algorithms and do not contain standard MI solutions such as graph searching or Q-learning. As students of in-

introductory programming courses learn a programming language with MALMÖ, they can use simple instructions and methods, and with them, they will assemble heuristic algorithms. Learning algorithmization in a game's environment is a popular way as students can link studying to gaming. Learning programming with agents is another aspect as the agent can solve harder problems as the student knows more about programming, so we can think that the student's and the agent's skills improve parallelly. The current record in the Red Flower Challenge with a simple heuristic agent is 53 flowers.

3.2.1. Some Sample Agents

The following simple heuristic example agents have been created for RFH: The GreenPill, Test Subject #40 and MrPoppy. These agents mostly used some variant of the spiral collecting strategy. In the Red Flower Hell phase 3, these agents achieved 15th and 16th places with 28 and 27 flowers collected, in phase 4, they performed better and reached the 3rd place with the Python agent, which collected 9 flowers, the 4th place with the C++ agent, which collected 8 flowers, and the 6th place with a Python agent, which collected 6 flowers.

3.3. RFH for AI Courses

These agents must already use standard MI techniques such as Q-learning. In AI courses, students know the basics of programming, so they could write better agents than the heuristic ones. As the goal and the environment is simple, it would be easy to teach an AI to collect flowers in this environment. Students could compare different models with different parameters to test, which one solves the problem better. As each phase is harder to solve than the previous, students could start with simple machine learning methods and the course could lead to deep neural networks and graph searching. The first example could be to train a decision tree, which will tell, what the next step will be based on the data we get from the environment. As MALMÖ supports Python, it is easy to integrate machine learning solutions to an agent, the difference is that the students see how their trained model decide in the game environment.

4. MALMÖ Competitions

In Microsoft MALMÖ, we have the opportunity to write AI agents and of course we can compete their problem solving skills with other agents. For this, we will introduce some challenges in this framework.

Our first question was that how AI agents could compete? In sports, there are simple competitions for running, for example, where the first competitor who reaches the goal wins. In esports and AI competitions the conditions are not so simple. The most straightforward winning condition is to reach the highest possible score in a specific task. However, these tasks are often varied just like in MALMÖ,

where one challenge is to catch the pig and ours was the Red Flower Hell, but there are many more.

Red Flower Hell was a series of competitions for High-level programming course in the first semester of 2020, where students could compare their programming skills to each other. We participated in these competitions so we could experience the competition based teaching and the positive effects of these events. In RFH, the goal is to collect as many red flowers as possible, which is not a hard task, because we only have to walk a spiral route. The hard part is that we have to do this before the lava flowing down the hillside has reached the agent. The competition has 4 phases, each of them relies on the previous one.

4.1. First Phase

In this phase, we have the opportunity to cheat, so for collecting flowers, we could use anything. The students had various solutions for this phase like querying all flower locations and teleporting to them or blocking the lava flow, so the agent has infinite time to finish the task. However, the trickiest solution was the most simple, the student replaced the lava with water, so the flowers flowed down to the agent.

4.2. Second Phase

The second phase was designed to be harder because there were limitations like agent cannot extract data or modify the map generator XML file, or agents must not use absolute movement, which imitates key pressing, they only could use discrete motion, which can move the character one block away. The goal was the same as in the first phase. Most of the students implemented nearly the same solution, that moved the character in a spiral path and tried to maximize the speed, however if the agent moves very fast, it becomes inaccurate, so it collects less red flowers.

4.3. Third Phase

The conditions of the third phase differ from the earlier phases because the rules did not change, but the participating agents must reach the qualification limit of 25 flowers. To participate, students must record a video of their agent successfully completing the task of collecting 25 flowers. The hard part of this phase is that recording video needs a lot of computing resources, so agents had to manage with less of it available.

4.4. Final Phase

The last phase of the competition took the agents to a different place than the gorge, where two slow zombies appeared to make flower collecting harder. In contrast to the first three phases, in this one, agents must fight the monsters to win, or the character dies, resulting fewer flowers. In this phase, our Python agent achieved the 3rd place, and our C++ agent 4th place.

5. Recommendation for Introductory Programming Courses

As we mentioned earlier, MALMÖ was an addition to High-level programming course, so the students could get a better mark if they practice with this software. As studying programming is based on practicing, MALMÖ makes it easier and interesting as students see the result of their code in their favourite game, Minecraft. We recommend MALMÖ to everyone, who is interested in programming, as it is an open-source project, and it supports beginners with many tutorials.

5.1. RFH for Undergraduate Courses

In our case, Red Flower Hell was optional, so not all students participated in the challenge, but teaching languages like Python and C++ could fully rely on Microsoft MALMÖ and its challenges. The main reason is that it has many interesting problems, which can be solved by undergraduate students and these solutions are visualized in a game environment. Our experience is that most first year students do not know how to solve a problem with programming or what will be their softwares' result. This is because they do not know the given programming language deeply enough, so they must experiment, but debugging a console application is not as interesting for them. To keep the level of interestingness high enough, they must see the results of their code in a living environment. The MALMÖ agent gives feedback immediately, so they can experience the success of their working code, which will result in improvements of the agents and further practicing.

5.2. RFH for Primary and Secondary School Pupils

We could extend RFH and other MALMÖ challenges to primary and secondary school students, who are interested in programming. As MALMÖ supports Python, C++, C# and Java, it can be the platform of teaching these languages before university. There are many advantages for both students and teachers, for example, many modern and popular programming languages can be taught with it and these can be compared too, coding is linked to gaming, so students remain interested, teachers can try modern technologies easier, students can write simple heuristic algorithms to solve simple problems. Our secondary school experience is that these students could write simple games in Godot [14] with Python, and they often practice programming in their free time if the result will be a better game. The technologies that these softwares use depend on both the teacher and the students, but it usually lacks machine learning features, however it could be used in simple games.

6. Results

This paper introduced a competitive teaching method using Microsoft's MALMÖ project and Minecraft. As creating agents to this platform needs minor programming skills, it is a good way to teach introductory programming courses, because most students are familiar with video games and linking gaming to studying could increase their motivation. To compete students' programming and problem solving skills, we created a challenge named Red Flower Hell, which consists of 4 phases. Each phase is a bit harder than the previous one, so students continuously encounter more complex problems, what needs more complex solution and code. With these phases, students first learned about the basics of programming and the MALMÖ environment, then they had to use simple instructions and methods, after that, they had to write heuristic algorithms, later they had to combine all of their knowledge to solve the problems. This challenge could continue with other phases that would support other courses with the topic of machine learning. The feedback from the students is positive, mostly because they learned faster as they could see the results of their programs in a game environment rather than a console window.

7. Conclusions

The way we used MALMÖ for the Red Flower Hell rewarded the problem solving with trial and error method, as trying out the changed agents took small amount of time and as the metric was the number of collected flowers, it was easier to improve its performance than in a more complex problem. However Minecraft needs a lot of resources that slowed down the development and testing. Supporting multiple programming languages is a huge advantage of MALMÖ, as we can code in the most popular languages like Java, C++, Python and C#.

As we progressed to Red Flower Hell's phase 4, the easily programmable tasks became complex ones, like beside the lava new threats were introduced in the form of hostile characters, which made the optimal solution harder to find and it was not trivial to decide what is the most intelligent behaviour. We can describe the dilemma with a simple example. The goal is to collect as many flowers as we can, but the hostile characters try to kill our agent. The first solution can be to ignore these characters as long as we can, but fight them if we must. This strategy was successful more times than the second solution. The second strategy can be to fight as much as we can, so the hostile characters will no longer be a threat to our goal. The problem with this one is that fighting takes a lot of time, and some of them was maybe unnecessary, so we will have less time to collect the flowers and avoid the lava. The first solution simplifies the problem by ignoring the hostile character factor, the second solution implements a complex behaviour even if it can be unnecessary. In this example which solution is the more intelligent?

8. Future Work

The Red Flower Hell challenge was easy to complete, because students knew where the characters and flowers would appear and what the map looks like. Developing agents in this environment was simple, because we knew what could threaten our character and what we can ignore from these, so we could optimize the behaviour of the agent. The result of this was that the agents performed well on a specific environment, but when it changed, they performed worse. In the future RFH challenges we could avoid this kind of overfitting with randomly generated elements and events in the environment. This could result effective agents, which can make decisions based on the circumstances. In RFH phase 4, the hostile characters were two slow zombies, so the agent could easily run away, but if one of them were a skeleton archer, running would not be an option.

In the future we could randomize the starting items of the agents or randomly place beneficial items that would help them solve the problem. We could place elements with doubtful outcome like water, which can speed up or slow down the movement, depending on the direction of flow. Another possible improvement could be to place more agents in a challenge simultaneously, so beside the goal of collecting flowers, developers must think about holding back other agents or killing them, what needs the prioritization of tasks. The question here is that how can we collect more flowers than the other agent? What is the most intelligent way? This extension rewards the out of the box type of thinking.

With these changes, Red Flower Hell challenge would support behavioural pattern developments over specialized agents.

The current RFH challenge with the constant map is good for introductory programming courses, but for more advanced courses it could be extended with the earlier detailed improvements, or developing supplementary softwares like map generator, map editor, and automated tester.

References

- [1] S. ADAMS, I. AREL, J. BACH, R. COOP, R. FURLAN, B. GOERTZEL, J. S. HALL, A. SAMSONOVICH, M. SCHEUTZ, M. SCHLESINGER, S. C. SHAPIRO, J. SOWA: *Mapping the Landscape of Human-Level Artificial General Intelligence*, AI Magazine 33.1 (Mar. 2012), pp. 25–42, DOI: 10.1609/aimag.v33i1.2322, URL: <https://ojs.aaai.org/index.php/aimagazine/article/view/2322>.
- [2] N. BÁTFAI, C. CSUKONYI, D. PAPP, C. HERMANN, E. DEÁKNÉ OSVALD, K. GYÓRI: *A DEAC-Hackers esport szakosztály mesterséges intelligencia oktatási és kutatási elképzelése a Minecraftban*, Mesterséges intelligencia – interdiszciplináris folyóirat II (1 2020), pp. 95–109, DOI: 10.35406/MI.2020.1.95.
- [3] M. CARREÑO-LEÓN, A. SANDOVAL-BRINGAS, F. ÁLVAREZ-RODRÍGUEZ, Y. CAMACHO-GONZÁLEZ: *Gamification technique for teaching programming*, in: 2018 IEEE Global Engineering Education Conference (EDUCON), 2018, pp. 2009–2014, DOI: 10.1109/EDUCON.2018.8363482.

- [4] D. CHEN, Z. LI, T. WANG: *Exploration and practice: A competition based project practice teaching mode*, *Mechatronics* 24.2 (2014), pp. 128–138, ISSN: 0957-4158, DOI: <https://doi.org/10.1016/j.mechatronics.2013.12.009>, URL: <http://www.sciencedirect.com/science/article/pii/S0957415813002432>.
- [5] C. DAY-BLACK, E. B. MERRILL, L. KONZELMAN, T. T. WILLIAMS, N. HART: *Gamification: An Innovative Teaching-Learning Strategy for the Digital Nursing Students in a Community Health Nursing Course*. The ABNF journal : official journal of the Association of Black Nursing Faculty in Higher Education, Inc 26.4 (2015), pp. 90–4.
- [6] B. GOERTZEL: *Artificial General Intelligence: Concept, State of the Art, and Future Prospects*, *Journal of Artificial General Intelligence* 5.1 (Dec. 2014), pp. 1–48, DOI: 10.2478/jagi-2014-0001, URL: <https://content.sciendo.com/view/journals/jagi/5/1/article-p1.xml>.
- [7] B. GOERTZEL, C. PENNACHIN: *Artificial general intelligence*, vol. 2, Springer, 2007, DOI: 10.1007/978-3-540-68677-4.
- [8] C. S. G. GONZÁLEZ, A. M. CARREÑO: *Methodological proposal for gamification in the computer engineering teaching*, in: 2014 International Symposium on Computers in Education (SIIE), 2014, pp. 29–34, DOI: 10.1109/SIIE.2014.7017700.
- [9] W. H. GUSS, C. CODEL, K. HOFMANN, B. HOUGHTON, N. KUNO, S. MILANI, S. MOHANTY, D. P. LIEBANA, R. SALAKHUTDINOV, N. TOPIN, M. VELOSO, P. WANG: *The MineRL Competition on Sample Efficient Reinforcement Learning using Human Priors*, 2019, arXiv: 1904.10079 [cs.LG], URL: <https://arxiv.org/abs/1904.10079>.
- [10] J. HSU: *AI takes on popular Minecraft game in machine-learning contest*, *Nature* 575 (Nov. 2019), pp. 583–584, DOI: 10.1038/d41586-019-03630-0.
- [11] M. JOHNSON, K. HOFMANN, T. HUTTON, D. BIGNELL: *The Malmo Platform for Artificial Intelligence Experimentation*, in: 25th International Joint Conference on Artificial Intelligence (IJCAI-16), AAAI - Association for the Advancement of Artificial Intelligence, 2016, URL: <https://www.microsoft.com/en-us/research/publication/malmo-platform-artificial-intelligence-experimentation/>.
- [12] G. KIRYAKOVA, N. ANGELOVA, L. YORDANOVA: *Gamification in education*, in: Proceedings of 9th International Balkan Education and Science Conference, 2014.
- [13] H. F. LADD, E. B. FISKE: *Does Competition Improve Teaching and Learning? Evidence from New Zealand*, *Educational Evaluation and Policy Analysis* 25.1 (2003), pp. 97–112, DOI: 10.3102/01623737025001095, eprint: <https://doi.org/10.3102/01623737025001095>, URL: <https://doi.org/10.3102/01623737025001095>.
- [14] J. LINIETSKY, A. MANZUR, R. VERSHELDE, ET AL.: *Godot Engine - Free and open-source 2D and 3D game engine*, 2020, URL: <https://godotengine.org/>.
- [15] LONGJUN HUANG, LIPING DAI, MINGHE HUANG, HAO WANG, BIN GUO: *Promoting teaching by competition and combining teaching with competition*, in: 2011 International Conference on Computer Science and Service System (CSSS), 2011, pp. 3383–3385, DOI: 10.1109/CSSS.2011.5972062.
- [16] D. PÉREZ-LIÉBANA, K. HOFMANN, S. P. MOHANTY, N. KUNO, A. KRAMER, S. DEVLIN, R. D. GAINA, D. IONITA: *The Multi-Agent Reinforcement Learning in MalmÖ (MARLÖ) Competition*, CoRR abs/1901.08129 (2019), arXiv: 1901.08129, URL: <http://arxiv.org/abs/1901.08129>.
- [17] D. SILVER, J. SCHRITTWIESER, K. SIMONYAN, I. ANTONOGLU, A. HUANG, A. GUEZ, T. HUBERT, L. BAKER, M. LAI, A. BOLTON, ET AL.: *Mastering the game of go without human knowledge*, *Nature* 550.7676 (2017), pp. 354–359, DOI: 10.1038/nature24270.

- [18] S. VILLAGRASA, D. FONSECA, E. REDONDO, J. DURAN: *Teaching case of gamification and visual technologies for education*, Journal of Cases on Information Technology (JCIT) 16.4 (2014), pp. 38–57,
DOI: 10.4018/jcit.2014100104.
- [19] O. VINYALS, I. BABUSCHKIN, W. M. CZARNECKI, M. MATHIEU, A. DUDZIK, J. CHUNG, D. H. CHOI, R. POWELL, T. EWALDS, P. GEORGIEV, ET AL.: *Grandmaster level in StarCraft II using multi-agent reinforcement learning*, Nature 575.7782 (2019), pp. 350–354,
DOI: 10.1038/s41586-019-1724-z.
- [20] P. WANG, B. GOERTZEL: *Introduction: Aspects of Artificial General Intelligence*, in: Proceedings of the 2007 Conference on Advances in Artificial General Intelligence: Concepts, Architectures and Algorithms: Proceedings of the AGI Workshop 2006, NLD: IOS Press, 2007, pp. 1–16, ISBN: 9781586037581,
DOI: 10.5555/1565455.1565457.
- [21] I. YILDIRIM: *The effects of gamification-based teaching practices on student achievement and students' attitudes toward lessons*, The Internet and Higher Education 33 (2017), pp. 86–92, ISSN: 1096-7516,
DOI: 10.1016/j.iheduc.2017.02.002,
URL: <http://www.sciencedirect.com/science/article/pii/S1096751617300696>.