Optimizing Minecraft Wheat Farming Stochastically

Author: Rodolfo Croes

Background:

Minecraft, as described by Mr. Clark, is a sandbox game where you set your own goals and find your own methods of achieving them [1]. Which is very thematic, seeing that this section of the project is about creating our own problems and solving them our own way. For my case, I wanted to integrate Minecraft in some sort of way into this report thus I started to think about what specific goal can be optimized in Minecraft. I looked at real life optimizations problems as inspiration and seeing that farming is a field that is full of optimization strategies, I look towards Minecraft and its farming mechanics. When I look at what criteria defined an optimal wheat farm in Minecraft, it is said by the fandom wiki page to be at a minimum a 9 by 9 area of farmland blocks with the center block being a water source block, and that each farmland block is getting at least 9 levels of light [2]. Wheat is a special type of crop because only when it is fully grown can it drop the item Wheat which is used in many ways such as making bread and breeding cows. Given this condition, you could let the crops in a farm be fully grown before harvesting. However, this comes with a flaw given that some or most of the crops would be fully grown before the last crop grows completely thus the slowest growing crop is considered a bottleneck for the farm. So, what can we do to mitigate this bottleneck while minimizing loss of production given that all the crops will be harvested at the same time? This is what I plan to achieve to find out in this project.

Aim:

The main objective of this project is to determine at what time, on average, should you harvest all the crops in a 9 by 9 Minecraft farm to maximize production while minimizing the time that the fully grown crops stay unharvested? Some assumptions that will be made for this project is that the farmland is fully hydrated by one water block in the center of the farm, and that each crop gets the proper amount of light such that the growth is not affected by the day and night cycle of the game. Some constraints will include an initial time cap which is the latest time the farm will be harvested, a fixed update rate that the game is running on and the number of stages of growth a crop has.

Model:

To start defining a model for this problem, we must consider how one wheat crop changes before it is fully grown. According to Miss Karlijk from Game Horizon, the wheat crop must go through 8 stages of growth before it is considered fully grown and ready to harvest [3]. She also states that the maximum amount of game loop runs, also know as ticks, it takes for a crop to be fully grown is approximately 72000 ticks which is 3 Minecraft days. There are 24000 ticks in one Minecraft day and one tick is equal to 0.05 seconds thus using this, and the approximation from before, we can assume that the latest point to harvest a farm for our problem is one hour. The type of tick that affects crop growth is called a random tick. This, according to the fandom wiki, has an average update rate of 68.27 seconds or 1365 ticks which is what we will use as the update rate in the model [4].

The way one wheat crop goes through the 8 stages of growth is dependent on a few variables. Firstly, it must have a light level of 9 or higher on the plant. For this model we will assume that all crops on our farm achieve this.

Secondly, if there is a wheat crop on the same diagonal as another wheat crop, then the growth rate of both crops is halved. This influences our farm because we can either accept the halved rates for more wheat crops growing at the same time or grow a different crop on floor(n/2) rows thus taking roughly half of the potential farmland to grow wheat on while allowing the leftover wheat to grow at an optimal rate. For this model, we will consider the farm with the most wheat growing at the same time because we are assuming that the farmer only cares about growing wheat.

Thirdly, if the farmland that the crop is planted on is hydrated, it increases the rate of growth of the crop drastically. For this reason, we will turn the center block of our farmland into water to hydrate all the crops in a 9 by 9 area.

Lastly, the growth rate is determined by probabilities and the amount of farmland surrounding a crop. The probability of a crop to grow from one stage to another is determined by the formula: 1/(floor(25/points) + 1) where points are determined by the state of the farmland that the crop is growing on and the amount of farmland surrounding the crop. The state of the farmland is when the farmland that the crop is on is either hydrated or dehydrated and these add 4 or 2 points respectively. The surrounding farmland of the crop consists of the 8 surrounding blocks and for each of them if they are hydrated farmland, add 0.75 to the points. If they are dehydrated farmland then add 0.25 to the points. Using these points, we can get a table of probabilities for each crop. These probabilities are gathered from the fandom wiki page and are represented by the table below:

Crop growth rate probabilities		
Surrounding farmland	Probability of crop per random tick (hydrated farmland)	Probability of crop per random tick (dehydrated farmland)
0	14.29%	7.69%
1	16.67%	8.33%
2	20.00%	9.09%
3	20.00%	10.00%
4	25.00%	11.11%
5	25.00%	12.50%
6	33.33%	12.50%
7	33.33%	14.29%
8	33.33%	14.29%

Figure 1: Table of Crop growth rate probabilities gathered from fandom wiki [3].

Seeing that the probabilities change depending on the location of the crop, we must categorize each crop by how many adjacent farmland blocks they have to get their probabilities. For a 9 by 9 farm, I have visualized the farm in figure 2 and have gotten a count for farm block in each category.

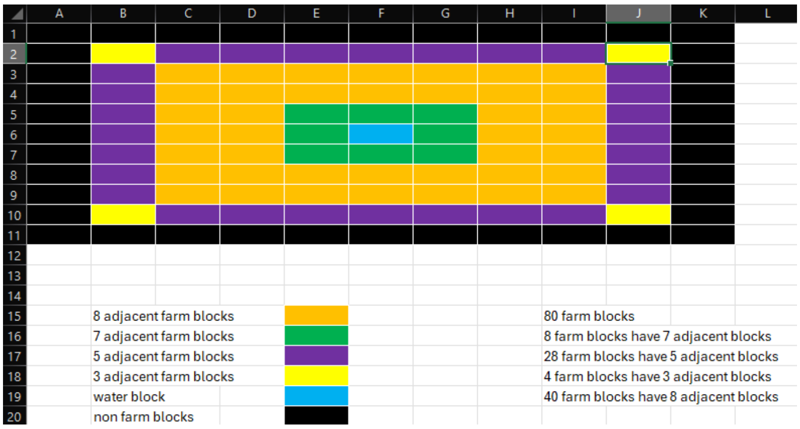

Figure 2: Visual representation of a 9 by 9 farm with the amount of farmland blocks and their adjacent farmland blocks

In the code, I have determined that for an n-by-n farm where n is greater than or equal to 5, the corners have 3 adjacent farm blocks, the sides excluding the corners have 5 adjacent farm blocks and the rest have between 7-8 adjacent farm blocks. Because the probabilities are the same for 7 and 8 adjacent farm blocks, I have counted them in the same group.

So, our model is a simulation of a 9 by 9 farm that returns the amount of wheat harvested each time. The objective function is to maximize the ratio of wheat harvested per time simulation. The form of this solution will be the optimal time of the largest ratio of the harvest to the nearest minute. To simulate the number of crops gathered at one point in time, we determine the number of updates that occur on average by dividing the harvest time by the update rate. From there we simulate each crop's growth in those number of updates and if they reach or the final growth stage, then we say that one wheat was harvested from that crop. We do this for all crops on the farm until we have a total amount of wheat harvested from the farm at that time point.

Optimization Method:

The way I have chosen to optimize this problem is by running many simulations of the farm at many different time points. The time points will be determined by random search and hill climb algorithms on the range of 0-72000 ticks or 0-3 Minecraft days. To determine when one result is better than another, we will take the ratio of wheat harvested between 0-80 divided by the point of the harvest in ticks. The random search starts by generating a new time point for the base case and from there gets the number of wheats harvested. Then we test the fitness of the base case by calculating the ratio. After this we generate a new random time point and determine the number of wheats harvested and the fitness value for this new point. We keep the new point if the ratio is better than the current best ratio and we run this for 1000 randomly generated points.

For the hill climb we generate new points by modifying the current best point and from there we analyze the fitness of the new point. We also introduce a change counter which stops the loop early if there has not been a change after a few iterations. Both algorithms are run for 100 iterations each to generate a list of 200 best ratios. From there we combine and sort the list by the ratio and determine a bound for the time point by taking the best 5% of the list and considering the smallest point as the lower bound. After this we ran the hill climb algorithm again but within the new bound for time to try and find a better ratio. After that we add the new values to the previous list, sort it and determine new bounds for points. We repeat the process of reducing the search space and running the hill climb algorithm for 4 generations until we get a bound and an estimated wheat production that satisfies maximizing the ratio.

Results:

From running the algorithms many times and collecting the data to the best 5 runs, we find that the best time to harvest the farm lies between 67205 and 71270 ticks or between 2.8 and 2.97 Minecraft days. The expected yield within this time lies between 60 and 63. The best ratio acquired from these simulations is 0.000893 at time 67205 ticks and with a yield of 60 wheats. There is not just one optimum in these results because while the simulations trend towards a range for time and crop yield, the randomness of the runs of the simulations changes where the optima can appear. For that reason, I prefer giving the answer in a range instead of a static number. I can say that we are not at a local optimum with the ratio because the upper bound for the time point does decrease when considering the ratio, thus harvesting the farm later can result in more yield but at a cost of time.

Conclusion:

When farming wheat in the maximum area that one water block can hydrate, we can say that within 2.8-2.97 Minecraft days, or between 56 and 59.3 minutes, after planting the wheat seeds is the optimal time to harvest the farm to maximize wheat output while minimizing the time crops stay fully grown. While the expected output may be less than when a player just waits for all the crops to grow completely, the yield is still sufficient for a farmer to be content with the production of the farm. The strength of the model developed is that it considers different growth rates of the crop depending on the position of the crop, the problem is scalable for farms between 5 by 5 and 9 by 9, and it uses a more accurate unit of measurement of time, ticks, compared to seconds or minutes. Weaknesses of the model include the scalability does not translate well to larger than 9 by 9 farms because there would need to be more water blocks added to the farm to ensure hydration, it does not consider other types of crops planted in a row configuration, and the time is capped to 3 Minecraft days.

References:

[2] J. H. Clark, Fantastic Minecraft Structural Designs, Farms, and Furnishings. Peachpit Press, 2014. Available at https://books.google.co.uk/books?hl=en&lr=&id=J4cbBQAAQBAJ&oi=fnd&pg=PA2&dq=minecraft+farming&ots=Xwl9iR5Abc&sig=jBjFbRlD3GZuZNHp4zlCljt2g_c&redir_esc=y#v=onepage&q=minecraft%20farming&f=false

[3] Fandom Wiki, "Tutorials/Crop farming," Nov. 13, 2023. https://minecraft.fandom.com/wiki/Tutorials/Crop_farming#Classic_farm (accessed Apr. 06, 2024).

[4] V. Kralijk, "Minecraft: Here's How Long It Takes for Wheat To Grow (& How To Speed It Up)," Oct. 16, 2023. https://fictionhorizon.com/how-long-does-it-take-for-wheat-to-grow-in-minecraft/ (accessed Apr. 06, 2024).

[5] Fandom Wiki, "Tick," Nov. 13, 2023. https://minecraft.fandom.com/wiki/Tick (accessed Apr. 09, 2024).