A Heuristic Approach to Indoor Rock Climbing Route Generation

Frank Stapel
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
f.t.a.stapel@student.utwente.nl

ABSTRACT

The problem of setting a good climbing route is faced in many ways around the world. This research looks into the possibilities of generating climbing routes. We aim to achieve this by creating a greedy algorithm using heuristics based on the analysis of existing climbing routes. The algorithm generates multiple routes using trees and determines the quality of those routes. To make the research feasible the algorithm was implemented using Python and applied to the structure and constraints of a MoonBoard. The generated routes were then compared to existing MoonBoard routes by experienced climbers. Based on their comparisons the quality of the routes was assessed based on criteria found by analysis and evaluation of existing climbing routes. The principles of the algorithm can be used for generating climbing routes on regular climbing walls as well. The assessment of the grade of a route by the algorithm is on a comparable level to the assessment of human climbers. This is, therefore, an important finding for future work in climbing grade classifiers.

Keywords

Indoor rock climbing, Climbing route generation, Difficulty assessment, Greedy algorithm, Heuristic algorithm, Classification, Formalization, MoonBoard

1. INTRODUCTION

Bouldering has rapidly increased in popularity in the last years [12, 19]. The competitive scene of sport climbing received a boost in popularity as well with the first appearance of sport climbing on the Olympic Games in 2020 in Tokyo [11, 18].

This increase in popularity has caused a new demand for route setters. These are people tasked with setting new climbing routes. This means placing new holds on a three-dimensional space of wooden planes. Currently, the setting of a route relies on the creativity and climbing experience of a route setter. Generating routes for the setters to place on the wall would make this task easier.

The goal of this research is to help in the formalization of climbing routes and, ultimately, to generate climbing routes of a certain quality. The generation of climbing routes can be used in many applications, of which two important applications come to mind:

First of all, there is the setting of routes for competitions. A competition route needs to be increasingly difficult to distinguish climbers based on their climbing skills. Setting a route that is gradually increasing in difficulty has proven to be a hard task when looking back at the routes of the climbing World Cups of the last couple of years. Having a route generated that can gradually increase in difficulty would be a solution to this problem.

The second application lies in training. Training on a system board [1] can quickly get repetitive when a climber climbs a certain route multiple times as resistance training. Variation of a certain route is usually not possible, as a climber wants to train specific aspects of their climbing and has picked that route to satisfy this specific training. The ability to generate multiple routes with the same criteria would remove the repetitiveness of the training. This is the main focus of this research.

We hope this research gives us insights into what makes a climbing route a good climbing route and what properties of a route determine its grade. Hopefully, this can be used in future work to create routes even better than those created to this day.

2. PROBLEM STATEMENT

This paper answers the following research questions:

RQ1 What heuristics can be used to generate a climbing route?

- **RQ1.1** How can different climbing holds be classified?
- **RQ1.2** How can different climbing moves be classified?
- **RQ1.3** How do different heuristics affect the grade of a climbing route?

RQ2 How can a climbing route be generated from these heuristics?

- **RQ2.1** How can different routes be generated using the same arguments?

RQ3 What criteria can be used to evaluate the quality of a climbing route?

3. RELATED WORK

Indoor climbing is a fairly new sport. The first indoor climbing hall opened in 1974 in Bolzano, Italy [8]. Most climbing related research has been done in the last two decades and looks into the physiological aspect of climbing [20].

32^{nd} *Twente Student Conference on IT* January 31^{st}, 2020, Enschede, The Netherlands.

3.1 Climbing Route Generation

Phillips et al. presented a formalized system transcribing climbing routes into a list of moves and climbing hold types. Using chaotic variations, plans for new routes were created based on existing routes which could then be set on a climbing wall by a route setter [15].

In this research we looked at roughly the same problem as Phillips et al. However, we took the opposite approach, as Phillips et al. reduces the scope of the problem by transcribing routes to a language with a small set of words, leaving the actual holds and locations up to the route setter. The advantage of this is that the generated route is also applicable to walls different from a MoonBoard. The disadvantage is that it still requires a route setter to set the route and still a lot of the route is left to the interpretation of the plan by the route setter. In this research, we left the route setter out of the process, by generating a route directly for the MoonBoard.

3.2 Climbing Route Grade Assessment

In previous research machine learning has been used to classify the grade of climbing routes. L. Kempen used a variable-order Markov-model based classifier using generated plans from the work of Phillips et al. as input [6, 15]. Dobles et al. tested multiple classifiers using the data of MoonBoard problems [3].

The results of these two papers show that using machine learning on the available data could not determine the difficulty of routes on the same level as human climbers. We hope that by combining insights into the effect of heuristics on the grade of a route (RQ1.3) and machine learning techniques; in future work a classifier can be made that can accurately determine the grade of a route.

4. BACKGROUND

To clarify some of the decisions made in section 5, this section provides background information about greedy algorithms as well as a motivation why the MoonBoard was chosen for this research.

4.1 Greedy Algorithms

A greedy algorithm is defined as *"an algorithm that always takes the best immediate, or local, solution while finding an answer. Greedy algorithms find the overall or globally optimal solution for some optimization problem, but may find less-than-optimal solutions for some instances of other problems."* according to P.E. Black [2]. The optimal solution for this route generation problem is a route that matches the desired properties given by the user and has the most *flow*. A route flows if each move comes naturally from the previous move. Breaking this flow can, for example, occur when a climber who has to *match*, a climbing term for grabbing one hold with two hands, a hold too small to be matched to go to the next hold or a climber who needs to grab a hold in an awkward position to complete the route. In this research we will look into heuristics that can help a greedy algorithm to generate routes that approximate the optimal solution. A greedy algorithm is used instead of an algorithm using machine learning or neural networks because we expect a greedy algorithm to perform better.

4.2 MoonBoard

For this research, the scope of the problem is reduced by using a MoonBoard [10], which is a training board that is used by climbers all around the world. Images of the MoonBoard are shown in appendix A. A MoonBoard has a fixed set of holds that are placed on the same location with the same orientation on each MoonBoard in the world. For this research the MoonBoard in Neoliet Boulderbar Oberhausen is used containing the MoonBoard Master 2017 hold setup on 40° overhang. An LED is placed under each hold. The LEDs can be controlled from the MoonBoard app containing over 50.000 routes that have been created by the MoonBoard community.

By using a MoonBoard, the problem space is reduced to a flat surface on a fixed slope. All holds are placed on a grid of 11 by 18 hold places, each with a space of 20 centimeters between them[1]. This means that the problem space is reduced to only the locations that are on this grid. Because each hold has a specific place and rotation on the grid, choosing a hold and rotation is excluded from the route generation. The only thing that has to be chosen is whether a hold should be part of the route or not.

The MoonBoard routes in this research follow a set of rules that make clear how a route should be climbed. The climber starts climbing with both hands on the holds indicated with green LEDs. If there is only one hold the climber starts with two hands on this hold. The route ends when the climber holds the finish hold with both hands in control for two seconds. The finish hold is always on row 18 of the MoonBoard and is indicated with a red LED. To go from the start holds to the finish hold all holds indicated with a blue LED can be used, as well as the footholds on the kickboard underneath the MoonBoard. The climber can use all these holds with both hands and feet.

5. METHODOLOGY AND APPROACH

Expert opinions were used because route setting is not an exact science. For these expert opinions various experienced climbers have been selected, each of which has been climbing for at least four years and has accomplished at least the bouldering grade 7A on the Fontainebleau technical scale [5]. These expert opinions are from now on referred to as experienced climbers.

5.1 Finding Heuristics (RQ1)

5.1.1 Classification of Climbing Holds

To find out what heuristics can be used to generate a climbing route we first looked into the classification of climbing holds.

For each hold the following aspects were taken into account:

- The difficulty of holding onto a hold if it were used in the best possible rotational angle.
- The rotational angle of the hold, relative to the ground.
- The primary type of the hold.
- Up to two secondary hold types.

The experienced climbers were asked to classify the difficulty of each of the 198 holds on the MoonBoard. The classification was made individually so it was not influenced by the opinion of the other experienced climbers. During the classification the experienced climbers tried out each hold and determined how difficult they thought it was to hold on to it. Each hold was then given a number between

[1]twice, the distance between rows is 22 cm instead of 20 cm. We will assume the small difference in row spacing won't have a significant impact on the results, to keep the algorithm as simple and clear as possible

1 and 5 with 1 being the easiest and 5 being the hardest based on the Likert scale [9]. To help the climbers chose between difficulties, some guidelines were given:

1. An easy hold; this can be steadily jumped to and caught with one hand.

2. Holding this hold, it is possible to cut loose feet if the other hand is placed on a hold with a similar score.

3. Holding this hold, it is possible to hang without feet in the wall if the other hand is placed on a hold with a similar score. Holding this hold when swinging, however, is not possible.

4. Holding this hold is possible, but only with both feet on the wall on good footholds if the other hand is placed on a hold with a similar score.

5. A difficult hold; this can only really be used as an intermediate. An intermediate hold is a hold that is only used to help the climber go to the next real hold. This means that the hand holding the intermediate will be the next hand moved when going to the next hold.

Because the experienced climbers all have roughly the same climbing level, these guidelines can be used as such. To make the difficulties more readable for later stages of the research, the given difficulties were converted to a 0 to 10 scale. Since each climber prefers other sorts of holds, the difficulty scores are subjective. The rotational angle and the type of a hold are objective. This means they could be classified by a single experienced climber. To improve the quality of this classification it was checked by another experienced climber.

For the classification of the rotation of a hold 8 directions were used. North, north-east, east, south-east, south, south-west, west and north-west were mapped to respective numbers 0 to 7. If a hold could be used in multiple directions, all the directions were noted down. For example, a hold that was held on the top was classified with a 0. A round hold that could be used from each direction was classified with all directions.

For the classification, each hold was assigned a hold type. On a MoonBoard a hold can be a jug, an edge (crimp), a pinch, a sloper or a pocket [14]. A distinction was made between a primary hold type and secondary hold types so holds could be described more accurately. For example, a sloping edge would be classified as an edge for its primary type and as a sloper for its secondary type. A sloper with an edge on the back would then be classified as a sloper for its primary type and an edge for its secondary type.

5.1.2 Classification of Climbing Moves

For the classification of climbing moves, we analyzed existing MoonBoard routes looking for different used moves. During this analysis, about 200 videos of existing MoonBoard Benchmarks were used [21]. These routes were all MoonBoard "Benchmarks", meaning they were selected and climbed by the staff of the MoonBoard company and associated climbers. These benchmarks are used as a baseline for all other created routes on the MoonBoard. Therefore it is a better representation of the moves in each grade than watching random public problems. Through watching these videos we made a summary of used moves in each climbing grade.

5.1.3 Effect of Heuristics on the Grade of a Route

In order to properly generate routes it is important to recognize how the heuristics found in sections 5.1.1 and 5.1.2 affect the grade of a route. Based on a group discussion with experienced climbers and the video analysis of section 5.1.2 a set of rules was made, giving insights into what affects the grade of a climbing route.

5.2 Generating Climbing Routes (RQ2)

Algorithm 1 A greedy algorithm for route generation

```
Data: Desired difficulty, hold types and length modifier
Result: A list of coordinates forming a climbing route
route = []
while last hold is not in row 18 do
    calculate the score of each hold
    add best scoring hold to route
```

In order to generate a climbing route we used the classification of holds, the classification of climbing moves, and the created set of rules used for determining the difficulty of a route. The form of the greedy algorithm is shown in algorithm 1. The main challenge was calculating the score of each hold. This was based on various factors. For each factor, a score between 0 and 1 was calculated, where 0 is the worst score and 1 the best. In order to get the final score of a hold, a weighted average of all factor scores was used:

- How well does the hold type match the desired hold type? This was based on the primary and secondary hold types given by the classification and the desired hold type.

- How well does the difficulty of the hold match the desired difficulty of the route? This was based on the difficulty given to the hold in the classification and the desired difficulty of the route.

- How well does the location of the hold match the desired location of the next hold? This was based on the current position of the climber. If this was the first hold in the route, a suitable starting place was awarded a good score. If this was the second hold in the route, the same applied as the first hold, but it was also necessary that the second hold was within reach of the first hold.

- How well does the rotation of the hold match the desired rotation? The desired rotation was based on the position of the climber and the rotations of the currently held holds.

5.2.1 Generating Different Routes

Algorithm 1 is deterministic. To make sure a climber gets different routes with the same parameters a tree was generated for each set of parameters. To get different routes all that has to be done is pick different paths in the tree.

Using a tree system greatly improves the execution time for the climber because the climber only needs to find a path in the tree of the used parameters. This means that generating the trees will take some time because each possible route for each subset of parameters is calculated. An execution time of under one second is acceptable for the climber finding a route in an already generated tree. The actual time taken to generate the trees is acceptable up to 6 hours because this only needs to be done once. The form of this tree implementation is shown in algorithm 2.

Algorithm 2 A greedy algorithm for route generation using trees

Data: Desired difficulty, hold types, length modifier, number of leafs
Result: A list of lists of coordinates forming climbing routes

```
calculate score of each hold
routes = [[best hold], [second best hold], ...]
final routes = []
while routes is not empty do
    for route in routes do
        calculate the score of each hold
        add best scoring reachable hold to route
        for 5 best holds do
            if new hold is not in row 18 then
                add the new route with the new hold to routes
            else
                add the new route with the new hold to final routes
```

The algorithm validates paths in a tree by assessing the quality of a path using criteria that were found. Using this quality assessment the algorithm sorts the paths, making sure the climber only gets the best generated routes.

5.3 Quality Analysis (RQ3)

In order to validate the generated routes we looked for criteria that can be used to assess the quality of a route. This was done by analysis and evaluation of existing routes. To test the quality of the generated routes, they were scored on the found criteria. The experienced climbers were given ten routes to climb; five of which were generated by the algorithm created in section 5.2. The other five routes were existing MoonBoard Benchmarks of which the grade has been verified by the MoonBoard company. The algorithm was compared to these routes. For both sets of five routes there was one route per grade, covering grades from 6A to 7B. For the algorithm each route was the route that was rated best by the algorithm's quality analysis. The MoonBoard routes were picked from the available top-rated MoonBoard Benchmarks.

The experienced climbers did not know the origin of the routes. They were asked to score each route based on the found criteria. Using these scores it can be determined how well the generated routes compare to the routes set by humans and more importantly, insights can be gained in the strong and weak points of the algorithm. After giving their scores for all routes, the origin and information of each route were revealed. After which, there was room for the experienced climbers to give some final remarks on the algorithm.

6. RESULTS

6.1 Finding Heuristics (RQ1)

6.1.1 Classification of Climbing holds

The final hold classification can be found in appendix D. In order to look at the reliability of the assigned difficulty scores the standard deviation of the difficulty score of each hold was calculated. The average standard deviations of each rounded difficulty score were compared and a trend line was drawn through the found averages as seen in figure 4 in appendix D. With an R^2 value of 0.85, the trend line is fairly accurate. It turns out the experienced climbers agreed more about the easiest and hardest holds than the holds in between. The climbers disagreed the most on holds that scored an average difficulty score of 4. This is logical because it's easy to say a hold belongs to the easiest or hardest holds on the MoonBoard. However, even with the given guidelines, deciding where on the scale an average hold belongs can be quite hard .

6.1.2 Classification of Climbing Moves

During the video analysis of MoonBoard Benchmarks routes the following moves were used:

- **The "normal" climbing move**; moving your low hand up to the next hold that is placed within the general width of the hold. The new hold is also placed higher than the current high hold. This move will from now on be referred to as a move or a normal move.
- **Dead-pointing**; while dead-pointing the climber dynamically moves to the next hold. When the top of the movement, the so-called dead-point, has been reached, the climber grabs the next hold before he starts falling down again.
- **Dyno-ing**; a dyno is a big, dynamic move to the next hold. A key characteristic of this move is that both hands simultaneously let go of the current holds and for a moment the whole body of the climber is in the air, not holding onto any hold, before catching the next hold. Performing a dyno is usually a solution to cover a large distance with big holds.
- **Bumping from an intermediate**; this move means an intermediate hold is used to go to the next desired hold. A hold is only an intermediate if it is already released when the climber's other hand moves to another hold.
- **Crossing over**; with this move you cross your low hand over your high hand to reach a hold that is on the other side of the high hold.
- **Crossing through**; this move is the same as crossing over, except your low hand goes underneath your high hand.
- **The Rose move**; a Rose move is almost the same as crossing through, except with a Rose move your head comes underneath your high hand as well. This move is seen only once in the video analysis. [2]
- **Using a side-pull**; a side-pull is a hold that is used from the side. Usually a side-pull goes combined with good footholds to set up for a big move.
- **Using an under-cling**; an under-cling is a hold that is used from underneath. Good footholds allow the climber to make a big move.
- **Using a heel-hook**; when performing a heel-hook a climber places his heel on a hold that is usually higher than normal footholds. Because the foothold can be pulled from with the climber's leg, this move can offer the potential of a big move.

[2] The Rose move gets its name from the famous outdoor climbing route "La Rose et le Vampire" which is located in Buoux, France. Antoine Le Menestrel made the first ascend in 1985 making this the fourth climbing route that received the grade 8b [13].

6.1.3 Effect of Heuristics on the Grade of a Route

After looking at a number of changes between routes of different climbing grades, both in the video analysis of section 6.1.2 and in routes in climbing gyms, a number of effects of properties of a route on the grade of a route were found. The effects are not in order of importance, as the correlation between these effects is not clear.

- If the holds of a route get more difficult, the route gets more difficult.
- If the distance between holds gets larger, the route gets more difficult.
- If the availability of footholds goes down, the route gets more difficult.
- If the number of special moves used increases, the route gets more difficult. The special moves referred to are all moves listed in section 6.1.2, except for the normal climbing move.

There are other factors that can influence the grade of a route as well. A major example of this is the angle of the climbing wall. Since the algorithm used in this research was limited to generating routes for a MoonBoard of 40°, this was not taken into account. The four effects listed above were the main things taken into account when creating the route generating algorithm.

6.2 Generating Climbing Routes (RQ2)

To test the functionality of the algorithm, an implementation was made using Python. The source code of this algorithm can be found on GitHub [16]. The general structure of algorithm 1 described in section 5.2 was kept throughout the whole implementation. In this section the key functionalities of the algorithm are explained in-depth, but not the whole algorithm is covered. Please look at the source code of the Python program for the full and exact implementation.

In section 5.2 it was determined that the main challenge of generating a climbing route is calculating the score of each hold. The score of each hold was calculated by taking the weighted average of 4 scores.

In many of the score calculations, Gaussian functions [4] were used. These functions give a score to a difference between a property of a hold and the desired property of a hold. Gaussian functions were chosen because their behavior is in many ways the same as the function that is desired: If this difference is close to 0, a very high score is given. When the difference gets larger than a certain threshold, the function starts to look like a linear function. This linear part shows the relation between a larger difference and a worse score. Finally, as the difference becomes even larger, the score approaches 0. This is beneficial to make sure a score of 0 is never given, making all scores fit to be combined by multiplication. The steepness of this linear function can be changed by changing a single variable in the Gaussian function, as seen in appendix B. In the algorithm the most used variable is 3, but 1, 2 and 8 were used as well in the algorithm.

The parameters in functions 1 to 6 are based on estimated values from analyzing the functions behaviour. With trial and error the parameters were then tweaked until the functionality of the score assignment functions were satisfactory. This was determined based on a visual inspection of multiple generated routes.

The difficulty score is calculated by comparing the difficulty rating of the hold with the desired difficulty. This desired difficulty of a hold is assigned using a linear conversion from the desired grade of the route. The desired grade is mapped from its real grade to a number, starting from 0. For example, 6A is equal to 0, 6C is equal to 2 and 8A is equal to 6. The following formula is then used to get the difficulty score:

$$\text{difficulty score} = e^{-\frac{1}{3}*(\text{hold difficulty}-\frac{\text{desired route grade}}{0.6})^2} \tag{1}$$

The hold score is calculated by counting the primary hold type matches and the secondary hold type matches. The final score is calculated as follows:

$$\text{hold score} = 0.70*\text{primary matches}+0.15*\text{secondary matches} \tag{2}$$

The move score is calculated based on the current position. For this implementation, only the normal move and the cross-over move were taken into account. First, the average of the currently held holds coordinates is taken to find the center coordinates. From the medium of a line connecting these coordinates, a perpendicular line is drawn upwards. The distance of the goal position of the next hold is calculated as follows:

$$\text{distance} = 2.5 + 0.5 * \text{grade} + \text{length modifier} \tag{3}$$

The goal position is then found by going up the perpendicular line by the distance found in function 3. The move score can then be calculated with the following function based on the distance between the new hold and the goal position:

$$\text{move score} = e^{-\frac{1}{3}*\text{distance}^2} \tag{4}$$

Multiple goal positions can be added. The move score will then be the score of the distance between the new hold and the closest goal position.

The rotation score is based on the direction of the goal position, which was used for calculating the move score. The difference between the rotation of the hold and the direction of the goal position was used as follows:

$$\text{rotation score} = e^{-\frac{1}{3}*\text{difference}^2} \tag{5}$$

The scores are then combined into the final score of that hold by calculating the weighted average. The difficulty score has a weight of 2, the hold score has a weight of 1, the move score has a weight of 3 and the rotation score also has a weight of 3. These weights were given after testing the algorithm for different weight assignments.

When climbing a generated route on the MoonBoard, the climbers can place their feet on the screw-on footholds on the kickboard underneath the MoonBoard, on the holds of the route, and on the wall itself. When generating the routes no footholds are added because it is assumed the climbers have enough options to place their feet.

6.2.1 Generating Different Routes

To improve the quality of the routes, multiple routes are generated in the form of a tree as described in section 5.2.1. The algorithm starts with 5 leaves. Each leaf is the start of a branch in the tree representing a new route variation. With each iteration the number of leaves that is added to the tree is reduced using the following function:

$$\text{new leaves} = \lceil \frac{2}{3} * \text{old leaves} \rceil \tag{6}$$

This means that after the third iteration in each iteration two leaves are added to the tree. This system is used to create a better variation in the routes. In general, there are more options for choosing a new hold low down the route compared to moves close to the top. If for the final move of a climbing route a hold needs to be chosen usually there are only one or two viable options. When choosing a second start hold there are usually a lot more options. Having a tree with more early leaves compared to more leaves in later iterations also creates wider trees, with more overall route variations.

After a tree is generated the routes in the tree are sorted based on overall quality analysis of each route. This qualitative analysis is based on the following points:

- The average score of each hold that was calculated in the hold selection process.
- A score based on how well the distance between holds fits the desired grade.
- A score based on how well the difficulty scores of the holds in the route match the desired grade.
- A score based on how well the hold types of the holds in the route match the desired hold types.
- A score based on the length of the route.

The implemented algorithm takes about 10 seconds to generate a tree containing 2693 routes and sort these routes based on the quality analysis described in the previous paragraph. This test was done for a route with grade 7A, Jug as hold type and a standard length modifier of 0. With 7 different grades, 31 combinations of hold types and 4 possible length modifiers, the number of combinations that can be used to generate the tree for each combination of arguments is 868. Assuming that each tree takes 10 seconds generating all trees will cost less than 15 minutes. Finding a route in the sorted results is done almost instantly. These times are well within the acceptable execution times described in section 5.2.1.

6.3 Quality Analysis (RQ3)

For the quality analysis the experienced climbers were asked to give their opinions of the shown routes on the following subjects:

- The grade of the route.
- A brief characterization of the route.
- The flow of the route.
- Possible awkward moves in the route.
- Their enjoyment of the route.
- If the route is a MoonBoard Benchmark or a route generated by the algorithm.

The full results of the test can be found in appendix C. The first thing that became clear from the test was that the routes from the algorithm were not as good and enjoyable as the MoonBoard Benchmarks. On average, the experienced climbers thought 80% of the Benchmarks had a good flow. For the routes from the algorithm this was only 45%. Also, the experienced climbers found 70% of the Benchmarks enjoyable, for the routes from the algorithm this was 40%.

From the test climbs, a few flaws in the algorithm became clear. Firstly, when routes move to the side, they end up without footholds. Because this causes the rotation of the handholds to be wrong the moves become awkward. Secondly, the flow from the algorithm was worse than the flow of Benchmark routes. This happens because the algorithm only tracks the two holds the climber is currently holding. It does not keep track of which hand is holding which hold. For example, this results in a route that requires the climber to move up with their right hand four times. Thirdly, the routes were said to be straight-forward. This is caused by only the normal move and the cross-over move being implemented, meaning it can be improved by implementing more move types.

Next, the grades given by the experienced climbers were compared to the actual grades. For the routes of the algorithm the actual grades are the grades that were used to get the routes and for the MoonBoard Benchmarks the grades were used that were assigned by the MoonBoard company. For the routes from the algorithm the average difference was 0.40 grade points. For the MoonBoard routes this was 0.65 grade points. A grade point is a step of one letter on the Fontainebleau technical scale [5]. This implies that the grades from the algorithm are closer to the opinion of the experienced climbers than the grades from the MoonBoard Benchmarks.

7. CONCLUSION

In this section, the main research questions are answered.

7.1 Finding Heuristics (RQ1)

What heuristics can be used to generate a climbing route?

For each available hold the difficulty, rotation, and hold types were classified. From a video analysis a list was made describing climbing moves used in MoonBoard Benchmark routes. From this analysis, we also retrieved rules that can be used to determine the difficulty of a route.

7.2 Generating Climbing Routes (RQ2)

How can a climbing route be generated from these heuristics?

Combining algorithm 2 and the heuristics found in section 6.1, we were able to use the heuristics to generate climbing routes. Multiple routes can be generated using the same parameters with the tree system described in section 6.2.1. The routes are ordered by the algorithm based on a quality assessment, allowing the algorithm to show the climber the best generated routes.

7.3 Quality Analysis (RQ3)

What criteria can be used to evaluate the quality of a climbing route?

Routes were tested based on the route's grade, flow, and the climber's enjoyment as well as some questions to determine flaws in the algorithm. From the test it turned out the experienced climbers found the routes from the algorithm to be less enjoyable and to have a worse flow than the MoonBoard Benchmarks. However, the climbers indicated still 40% of the generated routes were enjoyable. Keeping in mind the flaws found in section 6.3, an improved algorithm shows potential for generating routes that are consistently enjoyable.

When comparing the climber's opinions of the grades to the grades given to the routes by the algorithm the climbers deviated 0.40 grade points on average. The average deviation from the MoonBoard Benchmark grades was 0.65 grade points. This implies the rules described in section 6.1.3 function well in assessing the grade of a route.

8. DISCUSSION

8.1 MoonBoard Problem Space

In this research an algorithm for climbing route generation was applied in the problem space of a MoonBoard. While this made the project feasible by providing a relatively small structure in which the algorithm can generate routes, it constrains the properties of the generated routes. Because the MoonBoard has a standard overhang of $40°$ the climbing style used in these routes is limited. MoonBoard routes focus mainly on upper-body strength and contact strength in straight-forward moves. A climber's skills in technique, balance, and flexibility are not tested as well on the MoonBoard because of its constraints, meaning they are not implemented in the algorithm's scoring system. Adding more technical moves can be done by implementing more move types for the move score. The algorithm will probably not do well on balance and flexibility, because moves focusing on these aspects are not expressible with a focus on only handholds; in order to generate routes that focus on balance and flexibility the fundamentals of the algorithm need to be reconsidered.

Another key difference between the MoonBoard routes and routes in a climbing gym is the problem space: On a MoonBoard the algorithm can only choose between 198 holds that are fixed in place. On a real climbing wall, the possibilities are endless. Each hold can be placed everywhere on the climbing wall, in every rotation, due to the introduction of screw-on climbing holds, that do not rely on bolts. This means that for each next hold, the perfect scoring hold can be placed. For finding good variations on these routes chaotic variations from C. Phillips research can be used [15]. The main problem with applying an algorithm to generate routes on an indoor wall is that indoor climbing walls are three-dimensional. Because the MoonBoard is flat, it can be modeled as a two-dimensional plane. Normal climbing walls with volumes attached form a three-dimensional problem space, introducing many new problems. A good example of this three-dimensional problem space is the set of routes for the IFSC Hachioji World Championships Boulder Men Finals 2019, where the complicated wall structures can be seen.

This set of routes clearly shows they are on another level compared to the routes of the algorithm. Because the algorithm will not come close to generating this kind of route, routes from the algorithm are best fit for training purposes as described in the introduction. For competition style routes there are too many factors that need to be exactly right for the route to be good enough; this can currently only be achieved by human route setters.

8.2 Flaws in the Algorithm

The quality analysis of the routes showed the flow of the generated routes was worse than the flow of the MoonBoard Benchmarks. This can be attributed to the algorithm not tracking which hand is holding what hold and the absence of added footholds. The first issue is easy to solve by keeping track of a hold being used by the climber's left or right hand.

Adding footholds is a bit harder: The first problem is that it should be clear when an extra foothold is required. This will probably require a calculation based on the full route that takes place each iteration. The second problem is that the footholds need to be separate from the route in place, meaning there is no easier solution to climb the route that uses the added foothold as a handhold.

Solving these two issues will probably increase the flow of the routes as well as how much climbers enjoy climbing them.

8.3 Testing Generated Routes

Four experienced climbers gave their opinions on the generated routes during the quality analysis. Some of these experienced climbers also helped with classifying the difficulty of the holds on the MoonBoard. The grades that were given by the climber that did not classify holds were in line with those that did participate in the classification. This indicates that if there is a bias, it is probably not significant.

Before the test it was confirmed all participating climbers had minimal knowledge of MoonBoard Benchmarks on the MoonBoard Masters 2017 hold setup. However, there still was the problem that some MoonBoard Benchmarks were recognized as Benchmarks by the properties of the route. There were two properties that gave Benchmarks away: Firstly, all algorithm boulders have two starting holds. Some of the MoonBoard Benchmarks had only one starting hold. Secondly, the algorithm is colorblind: it uses all hold colors. Because some versions of the MoonBoard don't have all hold sets, there are also Benchmarks that, for example, only use black holds. Even though the climbers were not informed about the details of the implementation of the algorithm, some of the climbers had their suspicions during the test. Because the recognition of MoonBoard Benchmarks was not confirmed during the test, the result remains unbiased.

8.3.1 Empirical Test

Because the goal of the quality analysis was to get information on the strong and weak points a qualitative test was done. This was successful because the strong and weak points of the algorithm are now known. The quality test implied the difficulty assessment of the algorithm outperformed the difficulty assessment of the MoonBoard company. To confirm this hypothesis, a quantitative test is required. For this test, the sample size should be larger than the qualitative test. It is also important the group of test climbers is large and varies in gender, age, climbing skill level, and climbing style to ensure a result without bias.

9. FUTURE WORK

9.1 Improving the Algorithm

In order to continue research in the generation of climbing routes, the algorithm in this paper can be improved. Next to the flaws described in the results and the discussion, it would also be interesting to see the implementation of more different climbing moves.

Another option would be a new algorithm that can handle the three-dimensional problem space of regular walls found in climbing gyms, as described in section 8.1. The algorithm formed in this research might be a good point to start and the found heuristics might provide useful insights into the complexities of climbing route generation.

9.2 Difficulty Assessment

Existing literature describes assessing the grade of climbing routes using machine learning, but the results were not as good as the assessment of human climbers [6, 3]. The combination of machine learning models, a classification of the holds of the MoonBoard, and the relations between a route's properties and difficulty found in section 6.1.1 looks promising for a MoonBoard grading assessor that can outperform grades given by individual climbers making new routes for the MoonBoard. Because this assessor will apply the domain-specific knowledge of climbing to existing machine learning techniques, we expect it to more accurate than the grade assessors in the existing literature.

A problem with this assessor is that it needs to know the order of the holds in the route to determine the distances between the holds. The easiest order of holds must be found because, otherwise, the grade would be too hard and therefore incorrect. The assessor should also only use handholds when checking the difficulty of the holds. Adding a small foothold to an easy boulder should not increase its grade. When there is a method to find the easiest order of holds and detect foothold in a route, we expect this assessor to accurately determine the grade of a MoonBoard route.

It might also be interesting to look at the actual physical measurements of a boulder such as the distance between holds in meters or the slope or a sloping hold in degrees relative to the wall. These measurements will be crucial if climbing route generation is ever applied to generating routes for competitions to achieve the property of a competition-style route where almost every move is harder than the previous move.

Another interesting point would be a hold difficulty assessor that can create a better classification of the holds on the MoonBoard. By using existing MoonBoard problems and the heuristics found in this research a classification might be found that is more accurate than the classification used by the algorithm found in section 6.1.1. Despite the large sample size of over 50.000 routes, we doubt this assessor will find a classification that outperforms the classification used by the algorithm.

9.3 Application in Training

As described in the introduction, one of the applications of a climbing route generation algorithm is getting multiple routes that satisfy the same set of criteria. For example, generating routes of grade 6C that mainly use edges. It could be interesting to design a system that can automatically form a training scheme using said generated routes based on a climber's training needs. This could, for example, be multiple routes generated to be climbed consecutively to train resistance, or a few generated routes close to climber's maximum difficulty level that are climbed with some rest in between the routes. This can be used as a power endurance training.

9.4 Other System Boards

The algorithm is usable on other training boards as well. Examples of such boards are the Tension Board [17] and the Kilter Board [7]. To make the algorithm work on these boards, a classification of the holds needs to be made and added to the current Python implementation. Since the angle of these boards can be changed, it should be checked if the scoring system of the algorithm should change with the overhang of the wall.

10. ACKNOWLEDGEMENTS

I thank my supervisor Rom Langerak and my track chair Marieke Huisman for the useful feedback during the research process and for allowing me to research a topic that aligns with my interests.

I am grateful for the experienced climbers who twice spent a day to classify the holds on the MoonBoard and test the generated routes on the MoonBoard in Neoliet Boulderbar Oberhausen.

Finally, I would like to thank my family and friends for all their help with the research process and their critical views on the research.

11. REFERENCES

[1] S. Bechtel. Introduction to system board training. *trainingbeta.com website*. Cited January 2020.

[2] P. E. Black. Greedy algorithm. *Dictionary of Algorithms and Data Structures, P.E. Black, ed. 2*, February 2005. Cited November 2019.

[3] A. Dobles, J. C. Sarmiento, and P. Satterthwaite. Machine learning methods for climbing route classification.

[4] C. F. Gauss. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*, volume 7. Perthes et Besser, 1809.

[5] J. Kasse, M. Ryan, T. Mager, T. Briggs, C. Boscoe and N. Margetts. Grade conversions. *Rockfax website*. Cited November 2019.

[6] L. Kempen. A fair grade: assessing difficulty of climbing routes through machine learning. *Formal Methods and Tools, University of Twente*, February 2018.

[7] Kilter Grips. *Settercloset website*. Cited January 2020.

[8] H. Kröss. Salewa cube: Highlight for the bolzano climbing scene. *Salewa Cube website*, May 2011.

[9] R. Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 140:1–55, 1932.

[10] Moon Climbing. *MoonBoard website*. Cited November 2019.

[11] Geen rots te vinden, maar toch naar de spelen (no rock to be found, still to the games). *Nederlandse Omroep Stichting*, September 2015. Cited November 2019.

[12] Klimwanden in trek: bouwers profiteren van professionalisering klimsport (climbing walls popular: builders benefit from professionalization of climbing). *Nederlandse Omroep Stichting*, November 2018. Cited November 2019.

[13] P. Badenoch and A. Le Menestrel. *The Crag website*. Cited January 2020.

[14] J. Parks. How to use rock climbing holds. *REI website*. Cited November 2019.

[15] C. Phillips, L. Becker, and E. Bradley. strange beta: An assistance system for indoor rock climbing route setting. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 22(1):013130, 2012.

[16] F. Stapel. Climbing route generation, a python implementation. *https://github.com/Artemixx/crg-public*, 2020.

[17] Tension Climbing. *Tension Climbing website*. Cited January 2020.

[18] tokyo2020.org/en/games/sport/olympic/sport climbing/. Sport climbing. *Tokyo2020.org*, December 2018. Cited November 2019.

[19] S. Vrijmoed and A. Pruis. Waarom het einde van de boulder-hype nog niet in zicht is (why the end of the boulder-hype is not yet near). *Nederlandse Omroep Stichting*, March 2018. Cited November 2019.

[20] P. B. Watts. Physiology of difficult rock climbing. *European journal of applied physiology*, 91(4):361–372, 2004.

[21] YouTube user: andyliu84. Moonboard benchmarks 40 degree masters 2017 beta videos. *YouTube website*. Cited December 2020.

APPENDIX

A. MOONBOARD

In figure 1 a climber can be seen climbing one of the routes during the quality analysis in Boulderbar Neoliet Oberhausen. The frontal view of the MoonBoard from the MoonBoard mobile application [10] in figure 2 clearly shows the grid structure of the MoonBoard and can be used as a reference when going through the algorithm.

Figure 1. A climber using the MoonBoard to climb a generated route

Figure 2. Front view of the MoonBoard, currently showing MoonBoard Benchmark "Shin Ramen" (6C)

B. GAUSSIAN FUNCTIONS

The scoring system of the algorithm uses multiple Gaussian functions. These functions are plotted in figure 3.

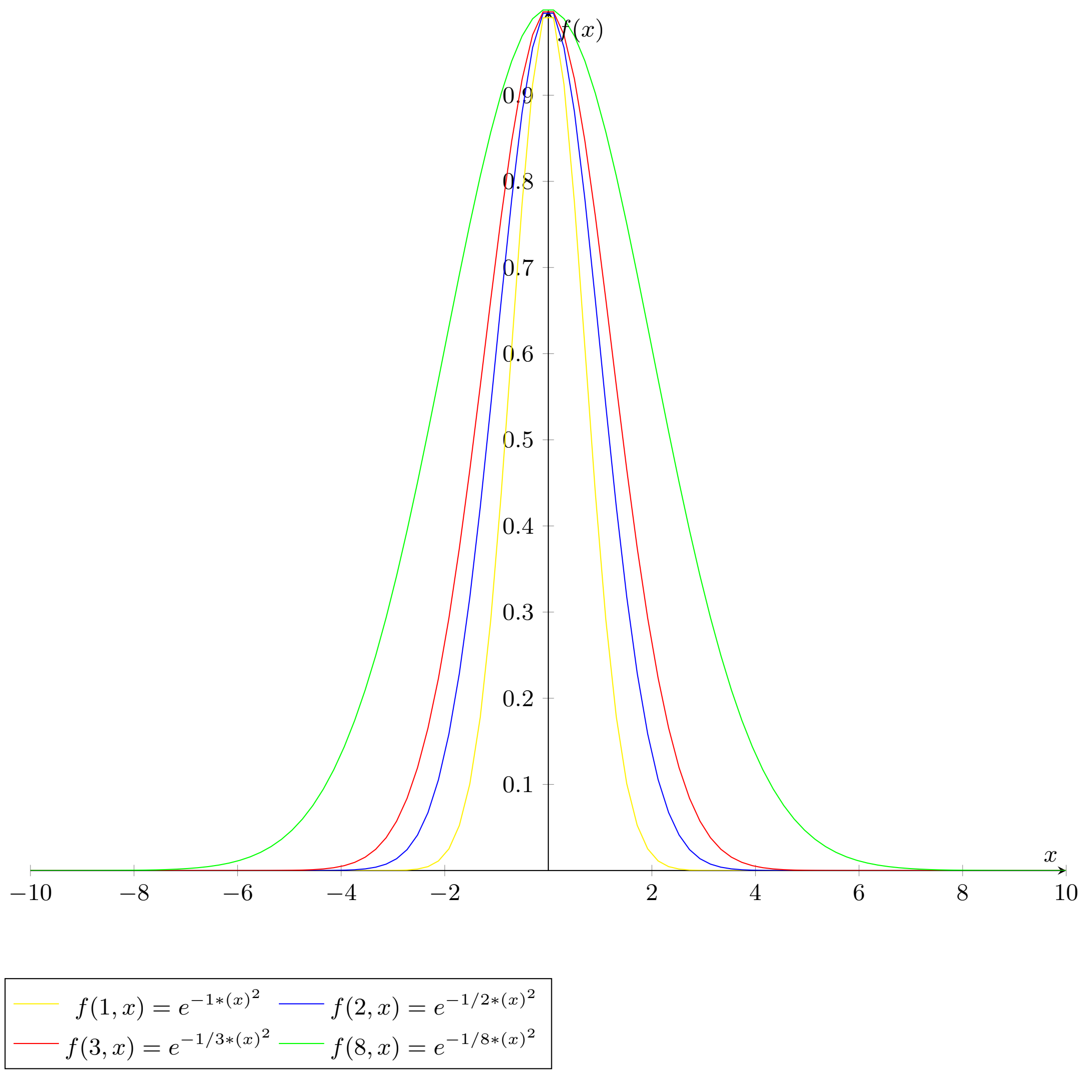

Figure 3. Different Gaussian functions

C. TEST DATA FROM QUALITY ANALYSIS

Table 1 shows the guess of the grade of each route per climber, comparing the routes generated by the algorithm to the routes from the MoonBoard Benchmarks.

Table 1. Guessed grade for each route, per climber

Route	Grade	Climber 1	Climber 2	Climber 3	Climber 4	Avg.	St. Dev.	Avg. Diff.	Abs. Diff.
6A+ Bnch.	0.5	1	2	0	1	1.0	0.82	0.5	0.5
6B Bnch.	1	2.5	2.5	2	2	2.3	0.29	1.3	1.25
6C Bnch.	2	2	2	1	1	1.5	0.58	-0.5	0.5
7A Bnch.	3	3	3	4	3.5	3.4	0.48	0.4	0.375
7B Bnch.	4	5	4	5	4.5	4.6	0.48	0.6	0.625
6A Alg.	0	0.5	1.5	-1	0	0.3	1.04	0.3	0.25
6B Alg.	1	0.5	1.5	1	0.5	0.9	0.48	-0.1	0.125
6C Alg.	2	2.5	3	3	2	2.6	0.48	0.6	0.625
7A Alg.	3	4	3	4	2.5	3.4	0.75	0.4	0.375
7B Alg.	4	4	3.5	3	3	3.4	0.48	-0.6	0.625

D. CLASSIFICATION OF CLIMBING HOLDS

Table 4 shows the full classification of the MoonBoard made by the experienced climbers. Table 2 shows a representation of the MoonBoard with the difficulty of each hold. Cell A18 represents the top left hold. Table 3 shows the same representation of the MoonBoard containing the standard deviation of the assigned difficulty to each hold, based on the difficulties assigned by the experienced climbers. Figure 4 shows the trend line for the standard deviations of the difficulty classifications in table 3. Here all difficulties are floored to determine their group.

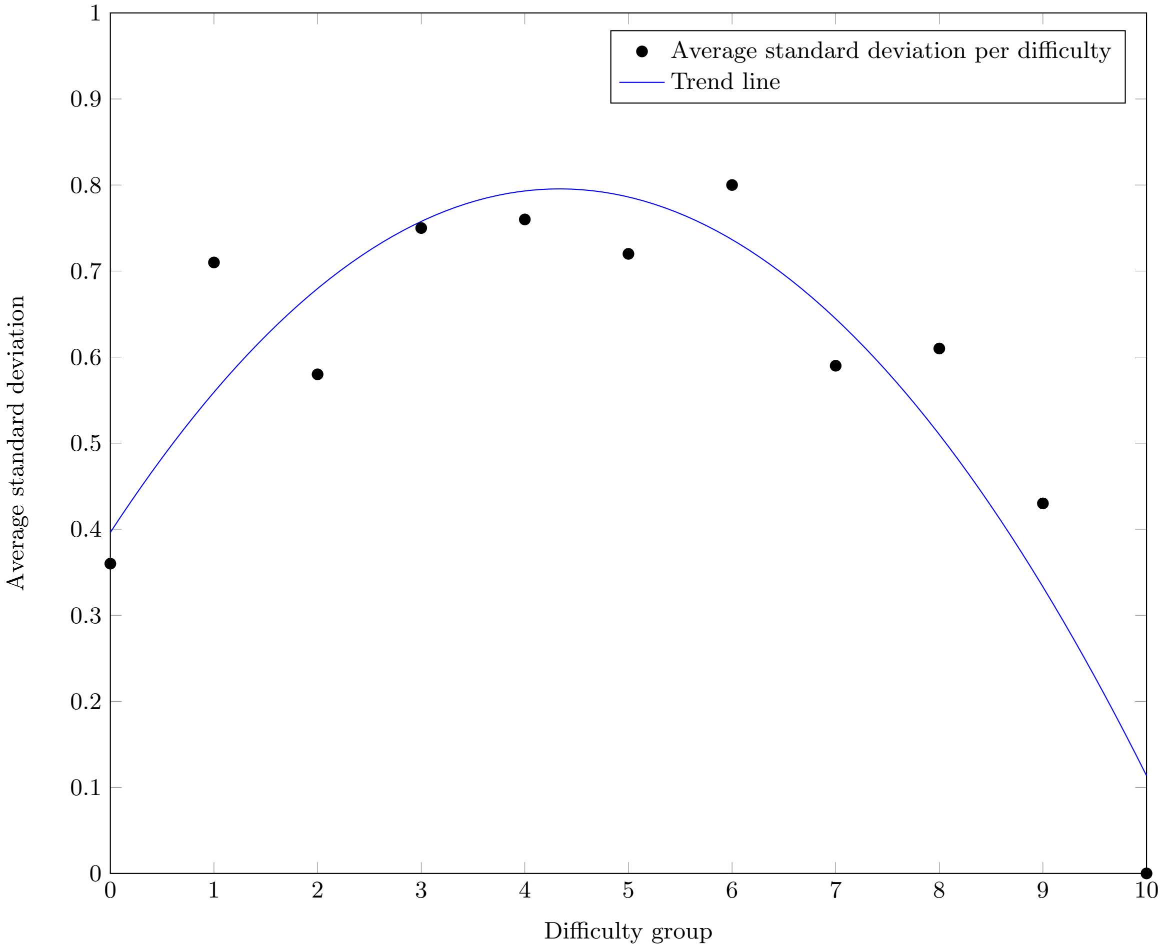

Figure 4. Trend line for standard deviations of difficulties found in the MoonBoard classification with difficulties from 0 to 10

Table 2. Difficulty classification, as a MoonBoard representation

	A	B	C	D	E	F	G	H	I	J	K	
18	2.5	0.6	7.2	4.4	4.7	3.8	8.1	1.9	7.2	1.9	7.5	18
17	7.8	9.4	8.1	5.3	8.1	5.0	4.7	6.3	10.0	9.4	4.7	17
16	3.4	10.0	0.3	6.6	4.7	8.8	6.3	7.5	5.3	1.3	7.5	16
15	7.2	6.6	8.8	4.1	7.8	7.5	7.8	6.9	9.4	7.2	1.3	15
14	2.8	9.1	2.2	9.4	4.4	7.2	5.6	5.9	6.9	8.8	7.5	14
13	8.1	3.4	3.8	7.8	1.3	5.6	2.5	7.8	4.4	4.1	2.2	13
12	5.3	5.9	6.6	5.9	6.6	3.4	6.6	8.4	6.3	7.5	8.8	12
11	3.1	9.4	6.6	6.6	4.4	8.8	3.1	5.6	6.9	1.9	4.4	11
10	5.9	6.3	3.4	1.9	8.8	7.2	3.8	3.1	6.3	8.8	5.6	10
9	8.1	5.9	6.9	2.8	6.9	5.0	7.5	6.3	6.9	2.5	6.3	9
8	4.1	4.4	9.4	7.2	4.4	2.5	4.1	5.0	1.3	6.9	8.1	8
7	7.2	6.9	5.0	7.5	6.3	4.7	1.9	6.9	1.3	6.3	7.5	7
6	2.5	7.5	6.6	6.6	5.0	5.0	4.1	8.1	6.3	6.9	6.9	6
5	0.6	8.1	9.4	3.8	7.5	3.1	7.2	2.5	6.9	8.1	4.4	5
4	2.2	4.4	3.4	2.5	5.0	5.6	4.4	5.0	6.3	4.4	5.6	4
3	4.1	3.8	9.4	7.5	3.1	8.1	6.9	7.2	6.6	6.6	7.8	3
2	5.6	3.8	9.4	2.8	7.8	3.8	7.2	5.3	5.9	3.1	5.6	2
1	3.4	4.4	8.1	5.0	7.8	4.1	6.9	6.3	6.6	6.6	2.5	1
	A	B	C	D	E	F	G	H	I	J	K	

Table 3. Standard deviations of the difficulty classification, as a MoonBoard representation

	A	B	C	D	E	F	G	H	I	J	K	
18	0.0	0.4	0.5	0.8	0.5	0.5	0.4	0.4	0.5	0.8	0.0	18
17	0.2	0.4	0.4	0.9	0.8	0.0	0.5	0.5	0.0	0.4	0.9	17
16	1.0	0.0	0.2	0.6	1.0	0.5	0.5	0.7	0.7	0.5	0.0	16
15	1.1	0.6	0.5	0.6	0.2	1.0	0.2	1.3	0.4	0.5	0.5	15
14	0.9	0.4	0.5	0.4	0.8	0.5	0.8	0.6	0.8	0.5	0.7	14
13	1.3	1.0	0.5	0.2	0.9	0.4	0.7	0.2	0.8	0.8	0.5	13
12	1.3	1.0	1.2	0.6	1.0	0.6	1.0	0.5	0.5	0.7	0.5	12
11	0.4	0.4	1.0	0.6	0.8	0.5	0.4	0.8	1.1	1.3	0.4	11
10	1.0	0.5	1.0	0.8	0.5	0.5	0.5	1.1	0.5	0.5	0.8	10
9	0.8	1.0	0.8	0.9	0.8	0.7	0.7	0.9	0.8	0.7	1.1	9
8	0.6	0.8	0.4	0.9	0.8	0.7	0.6	0.9	0.5	0.8	0.8	8
7	0.5	0.8	0.7	0.7	0.5	1.0	0.4	0.8	0.9	0.5	0.7	7
6	0.0	1.0	0.6	0.6	0.7	0.7	0.6	0.8	1.1	0.8	0.8	6
5	0.4	0.4	0.4	1.1	0.7	0.8	0.5	1.0	1.3	0.4	0.8	5
4	0.5	0.8	0.6	0.0	0.0	0.4	0.8	1.2	0.9	0.8	0.8	4
3	0.6	0.5	0.4	0.7	0.8	0.8	0.8	0.7	0.4	0.6	0.2	3
2	0.8	1.1	0.4	0.9	0.7	1.1	1.1	0.9	0.4	0.4	0.4	2
1	0.6	0.8	0.4	0.7	0.7	0.6	0.8	0.9	1.0	1.0	0.7	1
	A	B	C	D	E	F	G	H	I	J	K	

Table 4: **MoonBoard Hold Classification**

Column	Row	Difficulty	Rotations	Primary hold type	Secondary hold type	Secondary hold type
A	18	2.5	0	Jug		
A	17	7.8	0, 1, 7	Sloper	Pinch	
A	16	3.4	7	Edge		
A	15	7.2	0	Edge		
A	14	2.8	0	Jug	Sloper	
A	13	8.1	0	Edge	Pinch	
A	12	5.3	0	Edge		
A	11	3.1	0	Edge	Jug	
A	10	5.9	7	Edge	Sloper	
A	9	8.1	0	Edge		
A	8	4.1	7	Edge		
A	7	7.2	7	Pinch	Sloper	Edge
A	6	2.5	0, 7	Jug	Sloper	
A	5	0.6	7	Jug		
A	4	2.2	7	Jug	Edge	
A	3	4.1	0	Edge	Jug	
A	2	5.6	4	Edge	Sloper	
A	1	3.4	5	Edge	Jug	
B	18	0.6	0, 7	Jug		
B	17	9.4	0	Edge		
B	16	10.0	0	Edge	Pinch	
B	15	6.6	7	Edge	Sloper	
B	14	9.1	0	Pinch	Sloper	
B	13	3.4	0	Jug	Sloper	
B	12	5.9	7	Pinch	Sloper	Edge
B	11	9.4	7	Sloper	Pinch	
B	10	6.3	7	Sloper	Pinch	
B	9	5.9	0	Edge	Pinch	
B	8	4.4	7	Edge	Pinch	
B	7	6.9	0, 4	Edge	Pinch	
B	6	7.5	0	Edge	Pinch	
B	5	8.1	7	Edge	Sloper	
B	4	4.4	0	Edge		
B	3	3.8	5	Jug	Sloper	
B	2	3.8	3, 5	Edge		
B	1	4.4	4	Edge		
C	18	7.2	0	Edge	Sloper	
C	17	8.1	7	Edge		
C	16	0.3	0	Jug		
C	15	8.8	7	Edge		

C	14	2.2	7	Jug		
C	13	3.8	7	Jug		
C	12	6.6	5	Edge	Sloper	
C	11	6.6	0	Edge	Sloper	
C	10	3.4	0, 5	Pinch	Edge	
C	9	6.9	7	Edge	Sloper	
C	8	9.4	0	Edge		
C	7	5.0	7	Jug		
C	6	6.6	7	Edge	Sloper	
C	5	9.4	0	Edge		
C	4	3.4	0, 4	Pinch	Edge	
C	3	9.4	0	Edge		
C	2	9.4	0, 4	Edge	Pinch	
C	1	8.1	4	Edge		
D	18	4.4	0	Jug		
D	17	5.3	0	Sloper	Edge	
D	16	6.6	7	Edge	Sloper	
D	15	4.1	0	Jug	Sloper	
D	14	9.4	1	Pinch	Edge	
D	13	7.8	0	Edge	Sloper	
D	12	5.9	7	Pinch	Sloper	
D	11	6.6	7	Edge		
D	10	1.9	7	Jug		
D	9	2.8	7	Edge		
D	8	7.2	5	Pinch	Sloper	
D	7	7.5	5	Edge	Sloper	
D	6	6.6	0	Edge		
D	5	3.8	6	Edge		
D	4	2.5	7	Jug		
D	3	7.5	5	Edge		
D	2	2.8	7	Jug		
D	1	5.0	4	Edge	Pinch	
E	18	4.7	0	Edge		
E	17	8.1	All	Sloper	Pinch	
E	16	4.7	0	Edge	Sloper	Jug
E	15	7.8	7	Edge	Sloper	
E	14	4.4	1	Edge		
E	13	1.3	0	Jug		
E	12	6.6	0	Edge	Sloper	
E	11	4.4	1, 6	Pinch	Sloper	
E	10	8.8	All	Sloper	Pinch	
E	9	6.9	3, 7	Edge	Pinch	Sloper
E	8	4.4	0, 1	Sloper	Pinch	
E	7	6.3	0	Sloper	Edge	
E	6	5.0	0	Edge		
E	5	7.5	0	Edge	Sloper	
E	4	5.0	5	Pocket		
E	3	3.1	4	Jug		
E	2	7.8	4	Edge		
E	1	7.8	4	Pocket	Pinch	Sloper
F	18	3.8	0	Jug		
F	17	5.0	0	Edge		
F	16	8.8	1, 7	Edge		
F	15	7.5	1, 5	Pinch	Sloper	
F	14	7.2	0, 6	Edge		
F	13	5.6	0	Edge		
F	12	3.4	0, 4	Pinch	Sloper	
F	11	8.8	1	Edge	Pinch	
F	10	7.2	0	Edge		
F	9	5.0	0	Edge	Sloper	
F	8	2.5	0, 7	Jug	Sloper	
F	7	4.7	0	Edge		
F	6	5.0	7	Edge	Sloper	
F	5	3.1	0, 2, 4	Pinch	Jug	

F	4	5.6	0	Edge		
F	3	8.1	3	Edge		
F	2	3.8	4	Edge	Sloper	
F	1	4.1	4	Edge	Jug	
G	18	8.1	0	Edge		
G	17	4.7	0	Edge	Sloper	
G	16	6.3	7	Edge		
G	15	7.8	1	Edge	Sloper	
G	14	5.6	0	Edge		
G	13	2.5	1, 5	Pinch		
G	12	6.6	0	Edge	Sloper	
G	11	3.1	7	Edge	Sloper	
G	10	3.8	1	Pinch	Sloper	
G	9	7.5	1, 5	Edge	Pinch	Sloper
G	8	4.1	1	Sloper	Pinch	Edge
G	7	1.9	3	Pocket		
G	6	4.1	3, 7	Pinch	Sloper	
G	5	7.2	0	Edge	Sloper	
G	4	4.4	0	Edge	Sloper	
G	3	6.9	0	Edge		
G	2	7.2	3	Edge		
G	1	6.9	3	Edge		
H	18	1.9	0	Jug		
H	17	6.3	0	Edge		
H	16	7.5	1	Edge	Sloper	
H	15	6.9	0	Edge		
H	14	5.9	0	Edge		
H	13	7.8	0	Edge	Sloper	
H	12	8.4	0	Edge		
H	11	5.6	7	Pinch	Sloper	
H	10	3.1	0	Jug		
H	9	6.3	0	Edge		
H	8	5.0	1	Jug	Edge	
H	7	6.9	3	Edge	Sloper	
H	6	8.1	0	Edge		
H	5	2.5	2	Jug		
H	4	5.0	3	Edge		
H	3	7.2	3	Pocket		
H	2	5.3	4	Edge		
H	1	6.3	3	Edge		
I	18	7.2	0	Edge	Sloper	
I	17	10.0	0	Edge	Pinch	
I	16	5.3	7	Edge		
I	15	9.4	0	Pocket		
I	14	6.9	1	Edge		
I	13	4.4	3	Jug		
I	12	6.3	0	Pinch	Edge	
I	11	6.9	0	Edge	Sloper	
I	10	6.3	1	Pinch	Edge	
I	9	6.9	1	Edge	Sloper	
I	8	1.3	1	Jug		
I	7	1.3	1	Jug		
I	6	6.3	1	Edge	Sloper	
I	5	6.9	3	Edge		
I	4	6.3	1, 5	Pinch	Edge	
I	3	6.6	3	Pocket		
I	2	5.9	3	Jug		
I	1	6.6	5	Jug	Sloper	
J	18	1.9	0	Jug		
J	17	9.4	0, 2	Sloper	Pinch	
J	16	1.3	0	Jug		
J	15	7.2	1	Edge	Sloper	Pinch
J	14	8.8	0, 2	Sloper		
J	13	4.1	1	Edge	Pinch	

J	12	7.5	1	Edge	Sloper	Pinch
J	11	1.9	1	Jug		
J	10	8.8	1	Pinch	Edge	
J	9	2.5	1	Jug	Edge	
J	8	6.9	0	Edge		
J	7	6.3	1, 5	Pinch	Edge	
J	6	6.9	0	Edge	Pinch	
J	5	8.1	1	Edge	Sloper	Pinch
J	4	4.4	0	Edge		
J	3	6.6	1	Edge	Pinch	
J	2	3.1	1	Jug		
J	1	6.6	0, 4	Pinch	Edge	
K	18	7.5	0	Edge		
K	17	4.7	1	Edge		
K	16	7.5	1	Edge		
K	15	1.3	2	Jug		
K	14	7.5	1	Jug	Edge	
K	13	2.2	1	Jug	Pinch	
K	12	8.8	0	Pinch	Edge	
K	11	4.4	0	Edge		
K	10	5.6	1	Edge	Sloper	Pinch
K	9	6.3	1, 5	Pinch	Edge	
K	8	8.1	0	Edge		
K	7	7.5	2	Pinch		
K	6	6.9	0	Edge		
K	5	4.4	1, 7	Pinch	Edge	
K	4	5.6	2	Sloper	Pinch	
K	3	7.8	3	Edge		
K	2	5.6	4	Jug		
K	1	2.5	1	Pinch		