A Virtual Reality Mission Planner for Mars Rovers

Fernando Ropero, Pablo Muñoz, María D. R-Moreno and David F. Barrero
Department of Computer Engineering, Universidad de Alcalá, Madrid, Spain
Corresponding author: `david@aut.uah.es`

***Abstract*—For the operation of Mars rovers the ground team has a set of utilities to generate safe mission planning. A relevant aspect of such planning are the paths that the rover has to to follow to reach the scientific objectives. For path planning, operators use a combination of orbital and surface imagery to analyze the terrain topography to generate a cost map. Applying path planning algorithms over the cost map it is possible to obtain (sub)optimal solutions to safely drive the rover through different waypoints. As well, the paths obtained can be assessed using simulators and handmade improved by means of Virtual Reality techniques.**

In this paper we present an integrated planner for the mission planning and its evaluation using the recent technological advances in Virtual Reality. We have developed a low cost application to provide a three-dimensional view of the path generated exploiting real Mars surfaces with the aim of helping operators in deciding the best paths to follow.

I. INTRODUCTION

The application of Virtual Reality (VR) or Augmented Reality (AR) technologies are becoming largely popular in different industrial processes. While the AR devices could help for in-situ operations (e.g., airplane maintenance []), VR applications could be used for situations where the user access is restricted. Moreover, VR could provide relevant improvements for training (e.g., fighter pilots, surgery [1]) reducing costs and risks.

In this direction, NASA exploits VR applications to help operators of Mars rovers since the Pathfinder mission [2]. In general, exploiting VR applications provide operators an immersive perspective of the robot in a representation of the environment obtained using orbital data and rover imagery. For the newer Mars rovers, JPL exploits VT technologies and 3D stereoscopic displays combined with interfaces and controls to enable easier and accurate operations for the mission planning. This enables not only to assess the plan in a VR simulation environment, but also to reconstruct the rover operations in the VR environment with the telemetry data provided by the rover.

Our research line is focused on the mission planning, taking advantage of current VR solutions to enable the deployment of low cost solutions for mission planning assessment. In this regard, we have developed a path planning algorithm called 3D Accurate Navigation Algorithm (3Dana) [3] that works either with cost maps and/or Digital Terrain Models (DTM) such as the ones provided by the High Resolution Imaging Science Experiment (HiRISE) [4] on-board the Mars Reconnaissance Orbiter. This path planning algorithm is integrated in the Unified Task Planning and Path Planning Architecture (UP2TA) [5], which generates a complete mission planning considering the paths to follow when facing a mission with multiple goals in different positions.

The mission planning provided with the UP2TA planner can be evaluated by means of simulation environments. In our case, we are interested in being able to evaluate our mission planning in a VR environment. However, common simulation environments do not allow us to exploit VR technologies, meanwhile the current VR devices are usually quite expensive. Moreover, we are focused on the path assessing, so for our research could be relevant to analyze the path in an immersive environment for allowing the evaluation of potential risks. Then, we have developed two supporting tools: one to provide assessments of the generated paths using a classical 3D application and a second one for paths visualization and analysis using Google VR technology [6] and a mobile phone with a headset.

This paper is structured as follows. Next section enumerates current technologies for AR and VR. Following, path planning algorithms are introduced. Next, our mission planner is briefly described. After, a 3D application for paths assessing is presented. Then, a VR application for mobile devices that enables path visualization is summarized. Finally, some conclusions and future research lines are depicted.

II. CURRENT TECHNOLOGICAL SOLUTIONS

VR aims to create virtual environments with the help of a computer, in which the user shares the world with others as being with them, with the possibility to interact among them and within the world [7]. VR focuses on the sensory experience, so they include head-mounted goggles with audio and sounds through headphones to make the experience more lifelike. They may also include haptic systems with tactile information.

The betterment in VR is creating great opportunities for people to immerse themselves in worlds that are dangerous, unique or simply out of human reach. As an example, *Mars 2030*, a join project between NASA and Fusion VR, aims to create a new interactive, virtual Mars experience to simulate how it would be to stay on the Red Planet. However, most of the VR applications are focused on video game applications such as the HTC Vive headset. Another company that is investing in this technology is Facebook with the acquisition of the leader in VR technology, Oculus VR by the Oculus Rift headset [8]. Facebook is developing JavaScript libraries for the own Social Network, i.e. *React VR* for creating components to compose scenes in 3D, combining 360 panoramas with 2D UI, text, and images on the Web. It is also developing the

Facebook Spaces application that allows one to see herself and her friends in VR. And the Facebook Scenarios that can use avatars as some gaming platforms for users interaction.

For mobile applications, companies such as Samsung (with the Samsung Gear VR) or Google (with the Cardboard and Daydream headsets [6]) aim to lets players use VR applications, including YouTube, Google Maps Street View, Google Play Movies & TV, and Google Photos in an immersive view.

III. PATH PLANNING TECHNIQUES

Path planning is a widely researched problem in mobile robotics. The objective is to find an optimal (or near to it) path, avoiding known obstacles. Here, we refer to path planning in its classical view, i.e., we have a non-dynamic terrain that is well known a priori. This terrain is commonly discretized as a regular 2D-grid with blocked and unblocked square cells [9]. Furthermore, for each square cell we can define the traversal cost, i.e., a value that defines the effort required to traverse through that area. When the costs are non-uniform, we refer to the map as a cost map. If we add to the grid the terrain elevation, we call it DTM. In any case, a valid path is that starting from the initial node reaches the goal node without crossing a blocked cell.

In the following subsections we briefly describe the features of the most relevant heuristic search path planning algorithms. We use heuristic search algorithms as they are suitable for real applications because they are deterministic instead of other solutions such as Ant Colony Optimization or probabilistic search. We assume the reader is familiar with the terms and functionality of the A* algorithm that is the base of heuristic path planning algorithms. If it is not the case, we refer to the work of Hart et al. [10].

A. Theta Algorithm*

Theta* [11], [12] is a variation of A* for any-angle path planning on grids. It has been adapted to allow any-angle paths, i.e., it is not restricted to artificial angles of $\pi/4$ as is the case of A* with 8 neighbors. Theta* works similarly to A*, but considering the line of sight between two non-adjacent nodes. During the search, Theta* evaluates for the current position if between the line that connects the previous position and the next neighbor there is any obstacle. If there are no obstacles (i.e., there is line of sight), the current position is removed from the path, connecting directly the previous position with the neighbor, which are non-adjacent. This process removes intermediate nodes, reducing both the path length and the heading changes, which can have an arbitrary amplitude. As a consequence, Theta* only performs heading changes at the vertex of the obstacles. The drawback of this algorithm is the cost associated to perform the line of sight computation, which is performed frequently degrading the algorithm performance.

*B. S-Theta**

The Smooth Theta* (S-Theta*) [13] algorithm is a variation of Theta* that aims to reduce the amount of turns that the robot should perform to reach the goal. To do that, the algorithm is based on a modified cost function adding an $\alpha(t)$ value that measures the deviation from the optimal trajectory to achieve the goal, traversing a node t as a function of the direction to follow. Considering an environment without obstacles, the optimal path between two points is the straight line. Therefore, any node t that does not belong to that line will involve both, a change in the direction and a longer distance. In this regard, S-Theta* considers the heading of the robot during the path search, aiming to reduce not only the distance traveled, but also the number of heading changes and their amplitude. As a consequence, S-Theta* can perform heading changes at any point of the map.

*C. Field D**

Field D* belongs to the family of algorithms (D* Lite [14]) that use interpolation to produce better value functions for discrete samples over a continuous state space. The innovation in Field D* is a method for computing the cheapest path of each grid node t to the goal when dealing with a cost map, given the path costs of its neighboring nodes (ns_i). This value is traditionally computed as the minimum cost of traversing the edge between s and any of its neighboring nodes ns_i plus the path cost of the chosen ns to the goal. But this computation only allows straight-lines from t to one of the ns_i. However, Field D* allows a straight-line trajectory from node t to any point on the boundary of its grid cell of the ns_i. Since computing all the boundary nodes is infinite, it provides and approximation for each boundary point by using linear interpolation. Then, the cost of an edge that resides on the boundary of two grid cells is defined as the minimum of the traversal costs of each of the two cells. Then, the path is extracted by starting at the initial position and iteratively computing the cell boundary point to move next. Although in some cases, as reported by Ferguson and Stentz [15], the linear interpolation returns a bad approximation, in general results show the benefits of the algorithm.

D. 3Dana algorithm

The 3D Advanced Navigation Algorithm (3Dana) [16], [3] is a path planning algorithm designed to deal with realistic surface scenarios. In that sense, it can integrate during the search the DTM information in combination with a cost map, in order to try to generate safer routes, avoiding potentially dangerous areas and excessive slopes, while keeping the path length and heading changes as lower as possible. The search algorithm is based on A* with vertex re-expansion and integrates the line of sight check during search such as Theta* does. However, the line of sight check implemented in 3Dana provides the exact length for a segment that crosses a cell, while Theta* and other derived works only provide estimations at that point [17]. This allows us to obtain the real path cost for a given region in a cost map, which is the result of multiplying the segment length that crosses a cell by the cell cost. Another advantage is that, with this line of sight, we can support DTM information to perform any-angle paths using an interpolation method to obtain elevation for non-vertex points, allowing to

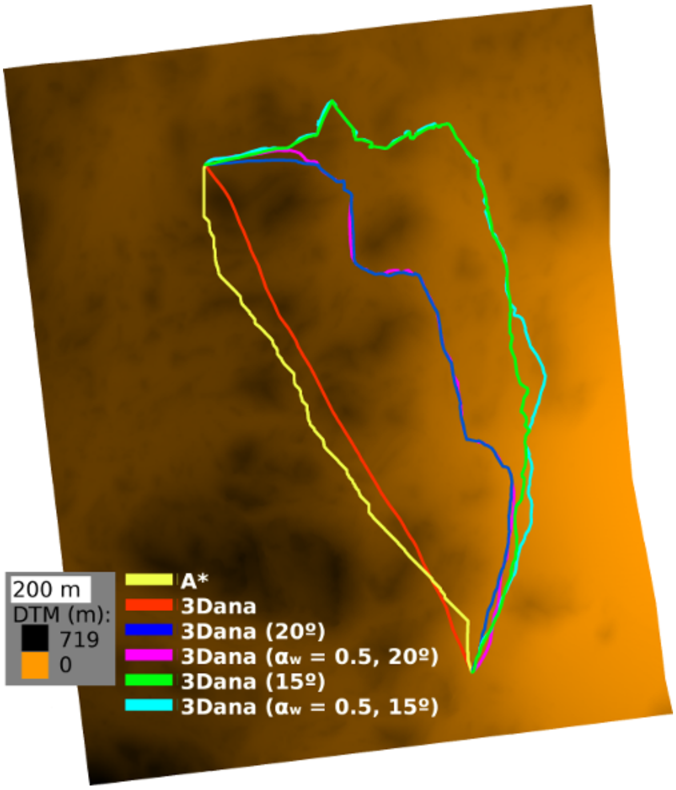

Fig. 1. Paths obtained for a region in the central uplift of a 30-Kilometer diameter crater in Noachis Terra using A* and different configurations of 3Dana. DTM ID: ESP_030808_1535

obtain the path length in non-flat surfaces. Also, using the DTM and the interpolation method we are able to compute terrain slopes at a given point. In this regard, 3Dana accepts a maximum slope constraint, discarding those paths that require to overcome the maximum defined.

For the heuristic, 3Dana takes into consideration both, the distance to the goal (which includes the altitude using a variation of the Euclidean distance) and the heading change required. Last one is inherited from the S-Theta* algorithm, and allows us to guide the search considering not only the path length, but also the current heading of the robot. This helps minimizing the total turns required in the generated path. In fact, it is possible to modify the weight of this heuristic, giving more or less relevance to the heading changes accordingly to the operational requirements. Fig. 1 provides a visualization of paths generated by 3Dana with different parameters on a real Mars DTM.

IV. THE UP2TA MISSION PLANNER

The UP2TA planner [5] integrates capabilities of path planning algorithms such as the presented in the previous section and task planning using a Planning Domain Definition Language (PDDL) [18], [19] planner. The main idea is to take advantage of path planning heuristics and merge them with domain independent heuristics to generate better solutions in robotic domains. During the mission planning, UP2TA considers the shortest path while performing scientific tasks in an efficient ordered way. A PDDL planner is responsible of ordering the tasks while the path planning algorithm searches for the route between tasks. These planners are highly coupled, allowing to merge the heuristics and cost functions of both planners to improve the plan quality.

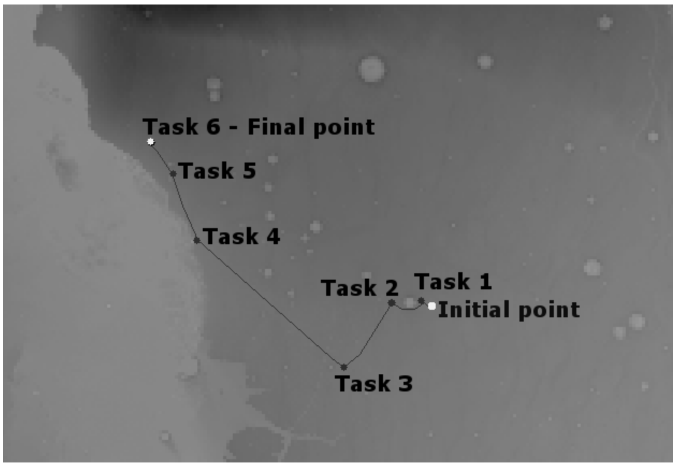

Fig. 2. Path obtained with UP2TA for a 6 goals mission in a Mars like terrain.

The current implementation of UP2TA combines a modified Fast Forward (FF) [20] planner and the 3Dana path planning algorithm. For the mission planning, the planner requires three inputs:

- Terrain: a DTM representation of the environment to be used by the path planning algorithm. For 3Dana we can use only the DTM (e.g., using the HiRISE data), the cost map or both.
- Domain: it is required to provide the PDDL domain that contains the operations that the rover can perform, which are up to the user to be defined. As well, an action called `Move_To` that allows the robot to move between waypoints has to be included. The waypoints are defined using the following format: `Ca_b` to represent a position in the terrain with coordinates x=`a` and y=`b`.
- Problem: the PDDL problem in which the initial state and the desired targets positions are defined.

Using these files, the UP2TA planner provides a sequence of actions with the (sub)optimal paths between the tasks, which achieves all the goals defined in the PDDL problem. For instance, fig. 2 provides a solution for a 6 goals mission on a Mars like environment.

V. 3D JAVA APPLICATION FOR PATHS ASSESSING

We have developed a light Java application for the assessing of the different path planning techniques presented above. This Java application allows operators to do more thorough comparisons among path planning algorithms thanks to its graphical environment, which sets up 3D Mars surfaces. For the 3D representations of the DTMs we have used jMonkeyEngine [21]. It is a free and open source game engine, designed for Java game developers who want to create 3D games using modern technology. The software is programmed entirely in

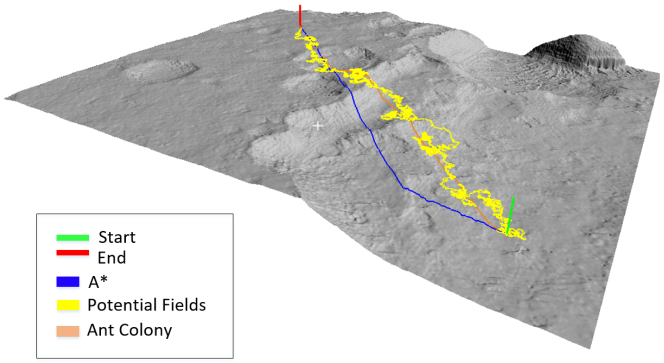

Fig. 3. Different paths generated with our Java application.

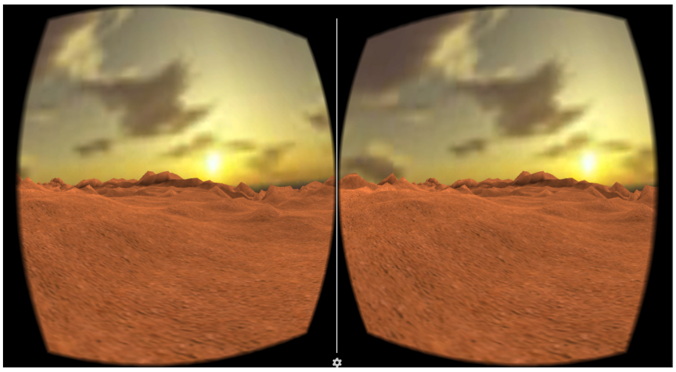

Fig. 4. Screenshot of our VR application.

Java, intended for wide accessibility and quick deployment. jMonkeyEngine gives to the Java application user-friendly interactions for camera movements over the 3D surface.

Our application is able to build 3D environments based on the DTM extracted from the HiRISE data, enabling operators to select specific regions of the DTM to enable focusing on relevant terrain regions (HiRISE produced DTMs often cover several square kilometers) by means of its 3D visual representation. This is an important feature because operators can select specific and critical surfaces where they can be trained with the rovers capabilities. When the 3D surface is loaded in our application, operators have the capability to generate multiple paths using the path planning algorithms integrated in our application, but also it is possible to load paths generated by other algorithms computed using external applications. Fig. 3 shows an example of three path planning algorithms computed in a 3D surface with our application. The three algorithms shown in the example are: A*, Potential Fields [22] and Ant Colony [23].

This application allows operators to compare solutions of different path planning algorithms. Furthermore, thanks to the 3D visualization with complete camera movement, our application enables operators to perceive accurate details and potential terrain risks for the rovers that are hard to realize in 2D representations. Thus, it is a useful way to assess the behavior of rovers over 3D surfaces.

VI. VR APPLICATION FOR PATH VISUALIZATION

To assess the solutions generated by UP2TA in a VR environment, we have exploited the Google VR Cardboard technology. To implement a VR application using this development kit, we use the Unity engine [24]. Particularly, exploiting this schema, we can create a VR application to provide to the user immersion in a VR environment with a representation of the Mars surface using the HiRISE data. This application runs on smartphones that have to be attached to a headset to provide stereo vision and hands-free capabilities. Therefore, the hardware cost is barely significant in comparison with the cost of other devices that require an expensive headset and a high end graphics computer.

Our VR application currently enables to walk over a DTM terrain with absolute freedom. Notwithstanding, we are working towards including a rover model inside the terrain to enable path execution. In this regard, the idea is to enable the operator to view in the VR environment how the rover performs when executing different paths generated by various configurations of 3Dana to select the best one based on the human expertise. Fig. 4 shows a screenshot of the VR application on a Mars terrain that we have developed for easing the path visualization.

VII. CONCLUSIONS AND FUTURE RESEARCH LINES

Newer AR and VR technologies have a great potential to enhance robotics operations not only from the usability (easy and user friendly interfaces) perspective but also providing new solutions to improve the safety of the mission plans to be executed by remote robotics platforms. Particularly, the operation of Mars rovers seems to be a good candidate to apply these new techniques and technologies. Currently, mission planning is performed typically by combining Artificial Intelligence techniques with human expertise.

In this paper we briefly described our works towards providing and integrated environment for planning the operations of rover-like robotics. This integrated environment is based on the UP2TA planner, which provides a complete plan for a mission. A relevant aspect of this plan are the paths to follow between different scientific targets. These paths are computed by means of the 3Dana path planning algorithm. This algorithm enables the generation of safer paths based on the information provided by a DTM. Notwithstanding, the algorithm is highly parameterizable, so it is possible to extract multiple paths with different quality measures. In order to enable operators to choose the best one, we have implemented a Java application to enable comparison and accurate 3D visualization of the terrain and the path, so the operator can use its expertise to select the best one. Finally, to improve the environment, we are working on a low cost VR solution to provide an immersible environment in which the operator can easily observe how a simulated rover executes the paths.

Currently, the VR application only enables terrain inspection. In the future we want to enable a complete integration of the mission planner inside the VR environment, so the operator can walk through the terrain, selecting the scientific targets and observing the generated plan in real time in the VR environment. In this sense, this research will include relevant aspects that are currently not deeper accessed such as the Human Machine Interface in the VR environment.

VIII. ACKNOWLEDGMENT

Authors want to thank Paolo Bellutta at JPL for his valuable comments about path planning for rovers drivers. This work has been supported by Spanish Ministry of Economy and Competitivity project EphemeCH (TIN2014-56494-C4-4-P).

REFERENCES

[1] Sopra Steria and Oslo University Hospital. (2017) Microsoft 2017 Health Innovation Award. [Online]. Available: www.youtube.com/watch?v=aCdZTsszwFo

[2] L. A. Nguyen, L. J. E. M. Bualat, L. Flueckiger, C. Neveu, K. Schwehr, M. D. Wagner, and E. Zbinden, "Virtual reality interfaces for visualization and control of remote vehicles," *Autonomous Robots*, vol. 11, pp. 59–68, 2001.

[3] P. Muñoz, B. Castaño, and M. R-Moreno, "3Dana: A Path Planning Algorithm for Surface Robotics." *International Scientific Journal of Engineering Applications of Artificial Intelligence.*, vol. 60, pp. 175–192, 2017.

[4] R. L. Kirk, E. Howington-Kraus, M. R. Rosiek, J. A. Anderson, B. A. Archinal, K. J. Becker, D. A. Cook, D. Galuszka, P. E. Geissler, T. M. Hare, I. M. Holmberg, L. P. Keszthelyi, B. L. Redding, W. A. Delamere, D. Gallagher, J. Chapel, E. M. Eliason, R. King, and A. S. McEwen, "Ultrahigh resolution topographic mapping of Mars with MRO HiRISE stereo images: Meter-scale slopes of candidate Phoenix landing sites," *Journal of Geophysical Research: Planets*, vol. 113, no. E3, 2008.

[5] P. Muñoz, M. D. R-Moreno, and D. F. Barrero, "Unified framework for path-planning and task-planning for autonomous robots," *Robotics and Autonomous Systems*, vol. 82, pp. 1–14, 2016.

[6] Google. (2017) Google VR SDKs. [Online]. Available: www.developers.google.com/vr/cardboard/overview

[7] R. Schroeder, "Being There and the Future of Connected Presence," *Presence: Journal of Teleoperators and Virtual Environments*, vol. 15, no. 4, pp. 438–454, 2006.

[8] OpsLab JPL. (2013) Building immersive environments. [Online]. Available: www.youtube.com/watch?v=nrcj-90M-f8

[9] P. Yap, "Grid-based path-finding," in *Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, vol. 2338. Springer Berlin / Heidelberg, 2002, pp. 44–55.

[10] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths." *IEEE Transactions on Systems Science and Cybernetics.*, vol. 4, pp. 100–107, 1968.

[11] A. Nash, K. Daniel, S. Koenig, and A. Felner, "Theta*: Any-angle path planning on grids," in *In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2007, pp. 1177–1183.

[12] K. Daniel, A. Nash, S. Koenig, and A. Felner, "Theta*: Any-angle path planning on grids," *Journal of Artificial Intelligence Research*, vol. 39, pp. 533–579, 2010.

[13] P. Muñoz and M. R-Moreno, "S-Theta*: low steering path-planning algorithm," in *Procs. of the 32nd SGAI International Conference on Artificial Intelligence*. Cambridge, UK: Springer, 10-12 Dec. 2012, pp. 109–121.

[14] S. Koenig and M. Likhachev, "D* Lite," in *procs. of he National Conference on Artificial Intelligence (AAAI)*, 2002.

[15] D. Ferguson and A. Stentz, "Field D*: An interpolation-based path planner and replanner," in *Proceedings of the International Symposium on Robotics Research (ISRR)*, October 2005.

[16] P. Muñoz, M. R-Moreno, and B. Castano, "3Dana: Path Planning on 3D surfaces," in *Procs. of the 36nd SGAI International Conference on Artificial Intelligence*. Cambridge, UK: Springer, 13-15 Dec. 2016.

[17] S. Choi and W. Yu, "Any-angle path planning on non-uniform costmaps," in *IEEE International Conference on Robotics and Automation*, Shangai, China, May 2011, pp. 5615–5621.

[18] D. McDermott, "The PDDL planning domain definition language," *The AIPS-98 Planning Competition Comitee*, 1998.

[19] M. Fox and D. Long, "PDDL 2.1: An extension to PDDL for expressing temporal planning domains," *AI Research*, vol. 20, pp. 61–124, 2003.

[20] J. Hoffmann and B. Nebel, "The FF planning system: Fast plan generation through heuristic search," *Artificial Intelligence Research*, vol. 14, pp. 253–302, 2001.

[21] jME team. (2016) jMonkeyEngine. [Online]. Available: www.jmonkeyengine.org

[22] J. Barraquand, B. Langlois, and J.-C. Latombe, "Numerical potential field techniques for robot path planning," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 22, no. 2, pp. 224–241, 1992.

[23] A. C. M. D. V. Maniezzo, "Distributed optimization by ant colonies," in *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*. Mit Press, 1992, p. 134.

[24] Unity Technologies. (2016) Unity Engine. [Online]. Available: www.unity3d.com