

The University of York

Department of Computer Science

Submitted in part fulfilment for the degree of BEng.

Assessing the Difficulty of a Generated Path-Planning Problem

Stephen Lewis Webb

April 5th 2016

Supervisor: Dr. Rob Alexander

Number of words = 0, as counted by wc -w.
This includes the body of the report, and Appendices ??, ??, but
not ??.

Abstract

Here is where the abstract goes.

Dedication.

Acknowledgements

Ack. Ack. Ack.

Contents

| | |
|---|-----------|
| List of Figures | 9 |
| List of Tables | 10 |
| I Preliminaries | 11 |
| 1 Introduction | 13 |
| 1.1 Motivation | 13 |
| 1.2 Background | 14 |
| 1.3 Project Aims | 14 |
| 1.4 Document Structure | 14 |
| 1.5 Statement of Ethics | 15 |
| 2 Literature Review | 16 |
| 2.1 Testing Autonomous Robots | 16 |
| 2.2 Path Planning Problems | 16 |
| 2.3 Factors of Difficulty | 16 |
| 2.4 Exploration | 16 |
| II Research | 17 |
| 3 Problem Analysis | 18 |
| 3.1 Introduction | 18 |
| 3.2 Task Definition | 19 |
| 3.3 Difficulty | 19 |
| 3.3.1 Occupancy Grid | 19 |
| 3.3.2 Difficult PPPs | 20 |
| 3.3.3 Measurement of Difficulty | 20 |
| 3.3.4 Application and Diversity | 20 |
| 3.4 Generator Requirements | 21 |
| 3.4.1 Rationale | 21 |

Contents

| | | |
|------------|----------------------------------|-----------|
| 3.5 | Simulator Requirements | 21 |
| 3.5.1 | Rationale | 21 |
| 4 | Design and Implementation | 22 |
| 5 | Evaluation | 23 |
| | | |
| III | Closing Remarks | 24 |
| | | |
| 6 | Conclusion | 25 |
| | | |
| 7 | Future Work | 26 |
| | | |
| 8 | Bibliography | 27 |

List of Figures

List of Tables

Part I

Preliminaries

1 Introduction

1.1 Motivation

Autonomous robotics are increasingly finding application in unpredictable, dynamic environments such as offices, road networks and shopping centres. Human oversight in these situations may be minimal or unlikely. Furthermore, the decisions taken by robots in these applications may well be safety critical - consider, for example, the actions taken by an autonomous car when an unseen pedestrian runs into the road ahead. Such robots must be carefully tested so that we are confident that they may be trusted to operate alone in these environments.

The problem of exploring and navigating the environment has received much work, producing well known methods such as SLAM. Emphasis is now moving towards testing such navigation algorithms so that we may be reasonably confident of the safety of both the robot platform and those in the environment around it. We must be confident that the robot will act safely in the situations it may find itself in during the course of its duties.

There is increasingly argument that traditional testing methods alone are not sufficient to prove the safety of an autonomous robot [ref Rob]. Consider, for example systems coverage - testing each system component at least once. This may not reveal faults in situations which are not addressed by any of the system components. Likewise, while we might test using requirements coverage, we cannot be certain that the stated requirements do not neglect to identify some unforeseen circumstance. Likewise, scenario coverage may have also neglected to include a scenario which such circumstances.

We must, in addition to traditional testing, aim to test the robot in as many diverse situations as possible. However, given any high level situation description - for example, navigating around a person in a room, we can generate a practically infinite set of test cases - every possible position of the person, every possible configuration of the furniture, every possible combination of every such variable. Clearly it is intractable to test every possible situation - instead we must aim to test a diverse

enough range of different situations to ensure we are confident in the algorithms and platform of the robot.

1.2 Background

1.3 Project Aims

While the generator is capable of evolving diverse maps, no attempt is made to assess the difficulty of navigating such maps. Doing so would allow the system to prioritise more difficult maps, covering more trivial situations implicitly.

This project therefore aims to improve the classification carried out, building upon Wei's generator. To do so the factors which cause a map to be difficult to navigate will be explored and identified. Following this, the most promising will be selected and a method of measuring each defined. This measure will be implemented into the generator. The classification taxonomy will be extended to include these new measures, allowing selection of maps to take them into account. Finally, the extended taxonomy will be evaluated against the original taxonomy.

It is hoped that these extensions will improve the set of maps generated, resulting in both more diverse and more difficult maps. By increasing diversity, we cover more situations and potentially uncover more faults. By increasing difficulty, we avoid testing trivial situations and improve confidence in the results of our testing - if difficult situations are successfully covered, we may be confident that the more trivial maps would also be successfully covered. As a result, the time and cost of testing an autonomous robot may be reduced whilst improving our confidence in its ability to act safely without human oversight.

1.4 Document Structure

This project is divided into the following structure.

Chapter 2 reviews existing literature on the testing of autonomous robotics and description of Path Planning Problems, before focusing on the factors that make an environment difficult to navigate. **Chapter 3** describes the problem this project addresses, defining the task to solve and the set of requirements to be delivered. **Chapter 4** sets out the design and implementation of the simulator and extensions made to

the generator, including rationale for design decisions made during the project. **Chapter 5** evaluates the extensions made to the generator utilising a set of test agents and the simulator. **Chapter 6** presents conclusions drawn from the evaluation of the changes made to the generator. **Chapter 7** suggests future work and improvements to the generator.

1.5 Statement of Ethics

2 Literature Review

2.1 Testing Autonomous Robots

2.2 Path Planning Problems

2.3 Factors of Difficulty

2.4 Exploration

Part II

Research

3 Problem Analysis

3.1 Introduction

Testing of autonomous robots is a difficult task and, as they become more commonly applied to real world problems, it is increasingly important that we are confident in their safe behaviour. As demonstrated by the literature review, this requires that we cover as many situations that may be encountered in the robot's work as possible - both those we can predict, and those that may not be obvious beforehand. While hand-crafted test situations provide insight into such behaviour, and can be carefully tailored to challenge weak areas of the algorithms, robot platform, and so on, they are time consuming to create and have no hope of catching failures that we can't predict (and therefore can't provide a scenario for) at a testing stage.

Previous work by Ashlock and Wei provides us with the means to algorithmically generate large numbers of diverse path planning problems as test cases, taking us some way in addressing the issue of testing a wide range of diverse and potentially unexpected situations. However, the simple fitness function employed in generating these PPPs would maximally reward an environment where the test agent is forced along a canalised, winding path which maximises distance travelled and turns made, with no opportunity for deviation. Clearly, such an environment is not a challenging test case - the only choice is between proceeding or reversing.

Further work is required to improve the generated problems such that the difficulty of a generated PPP can be assessed more accurately and evolved for as part of the fitness function. This would allow for selection of diverse test cases of a difficulty suitable to the application of the robot under test. The literature review identifies a range of hazards and difficulties employed by current testing standards and robot competitions; the chief amongst these is exploring an unknown map and successfully identifying the optimal route.

This section begins with a high level definition of the goal and task of this project, defining the terms and measurements discussed above in

more detail. Subsequently, the high level requirements will be laid out and rationalised.

3.2 Task Definition

The high level goal of this project is to improve Wei's PPP generator to assess and evolve for difficulty in the output path planning problems. These path planning problems are then to be used to find faults in test robots; hence the improved generator should be more capable at identifying faults by causing more test robots to fail to reach the goal location.

3.3 Difficulty

As demonstrated in the literature review, there are numerous ways in which a robot test environment may be said to be difficult. These include but are not limited to locomotion hazards such as rubble, complex collaboration tasks and knowledge representation issues. Currently the PPPs produced by Wei's generator are simple 2D grid environments and do not include such complex tasks or rough but passable terrain. However, a common theme in all of the competitions and standard tests discussed is that of knowledge representation and exploration. Clearly, no matter how full the environment is with debris or sensory hazards, any robot with perfect prior knowledge of the PPP will be able to take the optimal route to the goal. The difficulty actually lies in exploration of the environment, identifying these hazards and planning how best to avoid or handle them such that the best route can be found. Therefore, the measure of difficulty will seek to assess how difficult it is to explore a given PPP.

3.3.1 Occupancy Grid

Wei's generated PPPs can be considered as a 2D grid with a defined start and goal position and certain cells occupied by obstacles as described by the PPP itself. A test agent with some sensory capability - for example, perfect "sight" into immediately surrounding cells - must then explore and path to the goal. A dynamic programming algorithm (DPA) is used to assess the optimal path across this grid.

As discussed in the literature review, the occupancy grid map representation closely mirrors this representation. Exploration progress of a bounded map may be understood by initially setting the entire grid to a background ‘‘noise’’ occupation certainty. Then, during exploration, visited areas of the map become updated and their occupation becomes more certain. Thus unexplored areas are close to the background uncertainty.

3.3.2 Difficult PPPs

A ‘‘difficult’’ PPP is one where the goal position and optimal route from the initial position is difficult to identify in full without exploration. As such, the actual path taken may be far from optimal and the optimal route may only be apparent once the goal position has already been found and reached. The test agent, as a result of this deviation from the optimal path, incurs a higher than optimal movement cost to reach the goal position.

3.3.3 Measurement of Difficulty

By applying the occupancy grid map representation to the current PPP representation, uncertainty may be included in the DPA assessment of the optimal path from the initial position to the goal. Adjustments to the DPA evaluation function can be made to take into account the reward of exploration - finding parts of the optimal route, or the goal position - alongside the cost of movement. Hence, the DPA may model exploration of the PPP from very little a priori knowledge by a simple agent. The resulting optimal route produced by the DPA may be different from the optimal route as assessed by movement cost with perfect a priori knowledge - the deviation may hence be computed and the cost incurred by the requirement to explore derived. A more difficult PPP will therefore require more exploration and make the route difficult to find, increasing this cost. As a result, larger PPPs and those with a greater number of obstacles will clearly be measured as more difficult than smaller, simpler PPPs.

3.3.4 Application and Diversity

The difficulty measurement can be included in the fitness function and as a taxonomic character. Inclusion in the fitness function allows the

difficulty of generated PPPs to be adjusted and tailored to a level suitable for the robot under test. In addition, by using this measure as a taxonomic character, the produced PPPs will be clustered into groups of similar difficulty. Hence the final selection of test problems should be improved in difficulty while still allowing for diversity. This should improve the capability of generated PPPs to identify faults in test robots.

3.4 Generator Requirements

3.4.1 Rationale

The overall aim of this project is to assess a measurement of the difficulty of a PPP. Ro1 guarantees this. As discussed previously, a good testing standard should be able to challenge the whole spectrum of capabilities. This is covered by Ro2. We must also test for diverse PPPs, hence Ro3. The generator currently assesses diversity via a clustering taxonomy. The new measure must be included in this taxonomy so that it might be considered during PPP selection, hence Ro4.

3.5 Simulator Requirements

The code for Wei's evaluation simulator is not available, hence a new simulator must be developed to evaluate the changes made to the generator.

3.5.1 Rationale

Wei's work laid out a .ppp file format for generated PPPs. So1 ensures the simulator tool can read these independent of changes to the generator. To gather meaningful results, tests must be carried out over a range of agent capabilities, as set out by So2. In particular, buggy agents with known faults must be available so that the ability of the improved PPPs to detect such faults may be understood in comparison to Wei's PPPs. To allow for further analysis, these results must be stored as set out in So3. As we are dealing with agents with faults, it may be the case that an agent may path in a loop and never reach the exit. In this case the simulator should record the test run as a failure for that agent, as never reaching the goal is a clear fault. This is set down by So4.

4 Design and Implementation

5 Evaluation

Part III

Closing Remarks

6 Conclusion

7 Future Work

8 Bibliography

- [1] Leslie Lamport, *TEX: a document preparation system*, Addison Wesley, Massachusetts, 2nd edition, 1994.
- [2] My Dog, A dog, Dog, Dog, 2nd Edition, 2016