

Computer Games Development

Software Functional Specification

Year IV

Donal Howe

C00249662

[Draft]

24/04/2023

**DECLARATION**

****

**Table Of Contents**

[1 Introduction 1](#_Toc133255733)

[2 Technical Design 2](#_Toc133255734)

[2.1 Header Files 2](#_Toc133255735)

[3 Class Diagram 8](#_Toc133255736)

[4 CRC Cards 8](#_Toc133255737)

[5 Sequence Diagram 12](#_Toc133255738)

[6 References 13](#_Toc133255739)

###### List Of Figures

[Figure 3‑1 Class Diagram 8](#_Toc133255707)

[Figure 4‑1 CRC Card "Grid" 8](#_Toc133255708)

[Figure 4‑2 CRC Card"Game" 8](#_Toc133255709)

[Figure 4‑3 CRC Card "Menu" 9](#_Toc133255710)

[Figure 4‑4 CRC Card "Astar" 9](#_Toc133255711)

[Figure 4‑5 CRC Card "Dstar Lite" 9](#_Toc133255712)

[Figure 4‑6 CRC Card "Dijkstras" 9](#_Toc133255713)

[Figure 4‑7 CRC Card "Depth First Search" 9](#_Toc133255714)

[Figure 4‑8 CRC Card "Lpa Star" 10](#_Toc133255715)

[Figure 4‑9 CRC Card "Cell" 10](#_Toc133255716)

[Figure 4‑10 CRC Card "Screen Size" 10](#_Toc133255717)

[Figure 4‑11 CRC Card "Mode" 10](#_Toc133255718)

[Figure 4‑12 CRC Card "Debug" 10](#_Toc133255719)

[Figure 4‑13 CRC Card "Race" 10](#_Toc133255720)

[Figure 4‑14 CRC Card "Grid Size" 11](#_Toc133255721)

[Figure 4‑15 CRC Card "Which Algorithm" 11](#_Toc133255722)

[Figure 5‑1 Sequence Diagram 12](#_Toc133255723)

###### List Of Tables

[Table 3‑1 Description Of Helper classes and structs code 3](#_Toc133255701)

[Table 3‑2 Description Of Utilised Class Images 4](#_Toc133255702)

[Table 3‑3 Description Of Code 7](#_Toc133255703)

# Introduction

The objective of this project is to compare the benefits and drawbacks of using commonly used heuristic based guided pathfinding algorithms to the incremental algorithm known as Dstar Lite. This project will discuss the direct benefits of each algorithm in depth, from Astar, Dijkstras search algorithm, Lifelong Planning Astar, and the non-guided algorithm known as Depth First Search when compared to D star Lite within a game’s context.

# Technical Design

The purpose of this document is to effectively communicate the technical details and design decisions of the system/algorithm to the readers.

It could include software architecture, algorithm design, class specifications, pseudo code, etc. with tools such as UML, Class Diagram, CRC Cards.

## Header Files

|  |  |
| --- | --- |
| Helper classes and structs code images | Description |
|  | The Enum class called “WhichAlgorithm” which controls which algorithm is being used a certain time. This Enum class contains the name for every pathfinding algorithm in the project. |
|  | The Enum class called “GridSize” controls the size of the grid which is being used in the program ranging from “small” to “very Large”.  “small” = “10x10” grid  “Large” “50x50” grid  “very Large” = “100x100” grid |
|  | This Enum class called “Race” depicts whether you want to race the algorithms in comparison to dstar lite on a chosen path |
|  | The Enum class called “debug” toggles whether the user wants to see the variable values for Dstar Lite on the screen. This is only available with the small grid size |
|  | The struct called “Screen Size” struct which controls the size of each window |
|  | The Enum class called “Mode” which controls which mode the application is in. behaving differently depending on which one it is in |

Table 3‑1 Description Of Helper classes and structs code

|  |  |
| --- | --- |
| Utilised class images | Description |
|  | Public variables of Cell Class  the header file for the Cell(node) which has all the current functions in use. |
|  | Private variables to Cell Class |
|  | private member variables of the grid class |
|  | public members of the Grid class |

Table 3‑2 Description Of Utilised Class Images

|  |  |
| --- | --- |
| Algorithm’s classes: UI and games class images | Description |
|  | Functor used in Dstar Lite |
|  | Private members of the Dstar Lite Class |
|  | Public members of the Dstar Lite Class |
|  | Functor used in “Astar”. |
|  | “Astar” class as declared in the header file |
|  | Functor used in “LpaStar” |
|  | Functor used in “Dijkstra’s” search algorithm. |
|  | “Dijkstra’s” Class as declared in the header file |
|  | “Depth First Search” Class as declared in the header file. |
|  | “Menu” class as declared in the header file. |
|  | Public members to class “Game”, as declared in the header file. |
|  | Private members to class “Game”, as declared in the header file. |

Table 3‑3 Description Of Code

# Class Diagram

Diagram

Description automatically generated

Figure 3‑1 Class Diagram

# CRC Cards

Graphical user interface, text, application, email

Description automatically generated

Figure 4‑1 CRC Card "Grid"

Graphical user interface, text

Description automatically generated

Figure 4‑2 CRC Card"Game"

Graphical user interface, text, application

Description automatically generated

Figure 4‑3 CRC Card "Menu"

Table

Description automatically generated with medium confidence

Figure 4‑4 CRC Card "Astar"

Graphical user interface, text, application

Description automatically generated with medium confidence

Figure 4‑5 CRC Card "Dstar Lite"

Graphical user interface, text, application

Description automatically generated

Figure 4‑6 CRC Card "Dijkstras"

Graphical user interface, text

Description automatically generated

Figure 4‑7 CRC Card "Depth First Search"

Table

Description automatically generated with low confidence

Figure 4‑8 CRC Card "Lpa Star"

Table

Description automatically generated with medium confidence

Figure 4‑9 CRC Card "Cell"

Text

Description automatically generated

Figure 4‑10 CRC Card "Screen Size"

Graphical user interface, text, application

Description automatically generated

Figure 4‑11 CRC Card "Mode"

Table

Description automatically generated with medium confidence

Figure 4‑12 CRC Card "Debug"

Table

Description automatically generated

Figure 4‑13 CRC Card "Race"

Text, table

Description automatically generated with medium confidence

Figure 4‑14 CRC Card "Grid Size"

A picture containing graphical user interface

Description automatically generated

Figure 4‑15 CRC Card "Which Algorithm"

# Sequence Diagram

Diagram, box and whisker chart

Description automatically generated

Figure 5‑1 Sequence Diagram

# References

Swift, N. (2020) *Easy A\* (star) pathfinding*, *Medium*. Medium. Available at: https://medium.com/@nicholas.w.swift/easy-a-star-pathfinding-7e6689c7f7b2 (Accessed: April 24, 2023).

Likhachev, M. *et al.* (2005) *Anytime D\* - CMU school of computer science*, *Carnegie Mellon University School of Computer Science*. DARPA’s MARS program, NSF Graduate Research Fellowship. Available at: https://www.cs.cmu.edu/~ggordon/likhachev-etal.anytime-dstar.pdf (Accessed: April 24, 2023).

Stentz , A. (1994) *The D\*Algorithm for real-time planning of optimal traverses*. Available at: https://www.ri.cmu.edu/pub\_files/pub3/stentz\_anthony\_\_tony\_\_1994\_2/stentz\_anthony\_\_tony\_\_1994\_2.pdf (Accessed: April 24, 2023).

Lu, J. *et al.* (2022) *Jump point search algorithm*, *Encyclopedia*. Jiakai Lu. Available at: https://encyclopedia.pub/entry/24246 (Accessed: April 24, 2023).

Raheem, F.A. and Hameed, U.I. (2018) *Path planning algorithm using D\* heuristic method based on PSO in ...*, *American Academic Scientific Research Journal for Engineering, Technology, and Sciences*. Available at: https://core.ac.uk/download/pdf/235050716.pdf (Accessed: April 24, 2023).

Harabor, D. and Grastien, A. (2011) *(PDF) improving jump point search - researchgate*. Available at: https://www.researchgate.net/publication/287338108\_Improving\_jump\_point\_search (Accessed: April 24, 2023).

Rachmawati, D. and Gustin, L. (2020) *Analysis of Dijkstra's algorithm and a\* algorithm in ... - iopscience*. IOP Publishing Ltd. Available at: https://iopscience.iop.org/article/10.1088/1742-6596/1566/1/012061 (Accessed: April 24, 2023).

kaur, N. and Garg, D. (2012) *Analysis of the depth first search algorithms - gdeepak.com*. Available at: https://www.gdeepak.com/pubs/Analysis%20of%20the%20Depth%20First%20Search%20Algorithms.pdf (Accessed: April 24, 2023).

Pathak, M.J., Rami, S.P. and Patel, R.L. (2018) *Comparative analysis of search algorithms - ijcaonline.org*. International Journal of Computer Applications. Available at: https://www.ijcaonline.org/archives/volume179/number50/pathak-2018-ijca-917358.pdf (Accessed: April 24, 2023).

Pandey, K.K. and Kumar, N. (2017) *A comparison and selection on basic type of searching algorithm in data ...*, *researchgate*. Available at: https://www.researchgate.net/publication/308119139\_A\_Comparison\_and\_Selection\_on\_Basic\_Type\_of\_Searching\_Algorithm\_in\_Data\_Structure (Accessed: April 24, 2023).

Foead, D. *et al.* (2021) *A systematic literature review of a\* pathfinding*, *Procedia Computer Science*. Elsevier. Available at: https://www.sciencedirect.com/science/article/pii/S1877050921000399 (Accessed: April 24, 2023).

Koenig, S., Likhachev, M. and Furcy, D. (2004) *Lifelong planning a∗*, *Artificial Intelligence*. Elsevier. Available at: https://www.sciencedirect.com/science/article/pii/S000437020300225X (Accessed: April 24, 2023).

Koenig, S. and Likhachev, M. (2002) *D\* lite - idm-lab.org*. Available at: http://idm-lab.org/bib/abstracts/papers/aaai02b.pdf (Accessed: April 24, 2023).