Published online on the journal's webpage: **http://journal.ittelkom-pwt.ac.id/index.php/dinda**

Journal of Dinda

Data Science, Information Technology, and Data Analytics

Vol. 4 No. 2 (2024) 75 - 81 E-ISSN: 2809-8064

Application of Artificial Intelligence in the Design of 2D Escape From Pirates Game with A Star Algorithm Search Method

Rahmat Kurniawan[1], Armansyah[2], Muhammad Idris[3*]

[1,2,3*]Department of Computer Science, Faculty of Science and Technology, State Islamic University of North Sumatra

[1]rahmatkurniawan@uinsu.ac.id, [2]armansyah@uinsu.ac.id, [3*]idriz4031@gmail.com

Abstract

This research is designed to provide a challenging gaming experience by integrating strategy and problem-solving elements. The A* algorithm was chosen due to its efficient ability to find the shortest path in a complex search space. The implementation of this algorithm allows the main character to dynamically avoid obstacles and pirate threats and reach the destination in an optimal way. The test results show that the A* algorithm not only improves game performance but also provides a more realistic and challenging experience for the player. For testing this application, using obstacles and measured based on the value of nodes on the game map. Based on the test results, the A Star algorithm was successfully applied when comparing the computations in the game and manual calculations in the Escape from Pirates game in the test. Thus, this research contributes to the development of artificial intelligence-based games and opens opportunities for further innovation in interactive game design.

Keywords: *Game, A Star Algorithm, Enemy, 3D*

1. Introduction

The development of digital technology has brought about significant changes in various aspects of life, including the entertainment and gaming industry. The gaming industry, in particular, is undergoing rapid evolution with the introduction of various technological innovations. One of the most prominent innovations is the application of artificial intelligence (AI) in game design and development. AI gives games the ability to deliver a more realistic, dynamic and challenging gaming experience [1], [2].

Algorithm A Star where this algorithm is a search method in finding the shortest route where the value of steps and nodes is useful for achieving a goal, this algorithm the author applies in a game to the enemy, so that the enemy can find a player so that he can play players in the game [3], [4].

In Indonesian, game is also known as play. Games are complex activities in which there are rules, play and culture. A game is a system where the player is involved in an artificial conflict, while the player can interact with the system and also the conflict in the game which is engineered or artificial.

The general understanding of *games* as the meaning of the word "game", is a noun that has a function that can be fun, fun, entertainment and others. In fact, play is one of the cultural studies that is closely related to the entire human civilization [5]. Play is also the bud of civilization not only in *homo sapiens,* even animals can express their emotions by "playing". Animals also don't need human intervention to teach them how to play. We can see how a puppy jumps up and down specifically to show excitement [6]

Games are also one of the entertainment media that can reduce a person's level of boredom from their daily work activities. *Games are* also capable of increasing one's intelligence when the *game* requires a level of agility from a player [7].

Games are basically entertainment because if the user plays the game, it will feel happy and also entertaining. In the current era, *games* are presented with a fairly sophisticated visualization quality because they are supported by technology so that players can be more interactive according to their own wishes and players can also live in the *game*. So it can also be mentioned that *games* develop hand in hand with modern technology [8], [9].

Received: 14-07-2024 | Accepted: 26-07-2024 | Published: 01-08-2024

In recent years, in the industry or *game* companies, the emergence of *games* that are increasingly interesting to play and quality in terms of visualization and in terms of story or *storyline*. For example, the *flappy bird's game* has a simple concept but can attract attention and is widely played by players. The *game "Flappy Birds"* is foreign-made. But with the passage of time the development of *games* today is very rapid, it is unfortunate that gamers, especially in Indonesia, still often use several foreign-made *games*. *Games* that are specifically made by the nation's children are still not fully published so that what is often found is only some foreign or foreign-made *games*. It would be proud if Indonesian *gamers* love their own domestic products, especially *games* made by the nation's children [10].

Escape from Pirates Game is a *2D Adventure Game* which is a simple game *of* finding a way out and avoiding four pirates as *enemies* who chase us in this *game*, and after successfully escaping we don't just complete one *stage,* there are still 5 more *stages* that we have to pass to complete this *game*[6]. The *software* used by the author is *Unity 3D*, *Unity 3D* is an integrated *tool* for creating *games*, building architecture and simulations. Unity *3D* can be used for *PC (Personal Computer) games* and also *online games* [11].

*The A** (*A-Star)* algorithm is a *heuristic* search method that discards unnecessary steps with the consideration that the discarded steps are definitely steps that will not reach the desired solution. So, this algorithm is very suitable to be applied in the search for the enemy to attack us as a player in the *escape from pirates game* [12], [13], [10].

The A* algorithm and artificial intelligence (AI) can be used to make the game more engaging and difficult. AI enables the game to dynamically change the degree of difficulty and provide a more responsive and realistic gaming experience. One of the best and most successful pathfinding algorithms is the A* algorithm. Characters may identify the quickest routes and get around barriers more rapidly when it's used in game design, which enhances gameplay and performance. Opportunities for innovation in game design, especially in AI integration, are presented by this research. Other game creators may find inspiration in this method to investigate the application of AI in producing more interactive and captivating gaming experiences [14].

This work has the potential to make a major impact on the field of artificial intelligence research, especially in the area of pathfinding algorithm application. Academics and researchers interested in creating novel AI approaches and methodologies for simulation and gaming applications can also use it as a reference. Gamers may have a more realistic and immersive gaming experience with the use of artificial intelligence (AI) and the A* algorithm. As a result, playing might be more enjoyable and engaging for the player. AI-powered games can aid in the development of players' strategy and problem-solving abilities, which are two advantages of gaming beyond just amusement.

The issue stems from the urge to make modern games more complicated and realistic. Artificial intelligence (AI) that can provide a dynamic and challenging gaming experience is required as games get more complicated. Efficient pathfinding is essential in many strategy and navigation games, and the A* algorithm is widely regarded as one of the best techniques. However, implementing this method presents technical and design hurdles that must be overcome in order to assure optimal game performance and responsiveness. Players also demand a more realistic and engaged gaming experience, which can be achieved by implementing artificial intelligence. Furthermore, there is a need for creativity in game design, where the use of AI and pathfinding algorithms can open the way for new, more sophisticated features. This study is also critical for the advancement of knowledge and expertise in the fields of artificial intelligence and game production, as well as meeting the growing market need for more engaging and demanding games. As such, the goal of this project is to investigate and address numerous issues in the application of AI and A* algorithms in 2D game design, in order to contribute significantly to the production of more advanced and unique games.

Based on the description above, the author is interested in making the *game "Escape from Pirates"*. It is hoped that this *game* can be a reference for future research. In the *game "Escape from Pirates"* is presented with *2D* visualization in this *game*.

2. Research Methods.

In conducting this research, the author uses a way of working with several stages such as:

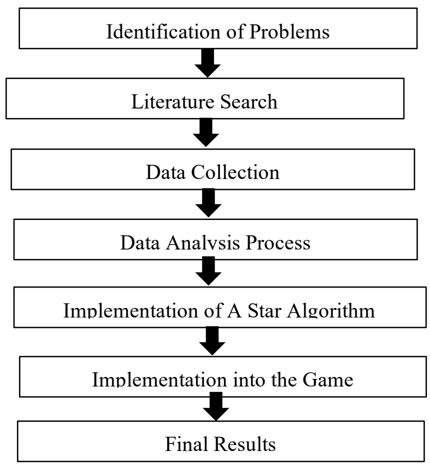

Figure 1. Research design

2.1 Observation

By directly observing an object to find information and knowledge related to research on the application of a *star* algorithm to *enemies* in *2d-based escape from pirates games* using unity *3d*.

2.2 Literature Study

In this study, several previously existing journals and books were searched to be used as a reference to develop a system made in research on the application of a *star* algorithm to *enemies* in *2d-based escape from pirate's games* using unity *3d*.

Gameplay system is in the game being played starting with the *output of the pirate* (*enemy*) in the initial position then the algorithm system directly searches for the route to the *player* then *the pirate* moves to chase the *player*, then there are two conditions obtained by the *pirate* while chasing the player, namely escaping and being caught, if the *player* escapes then the system continues to search for routes when the *player* avoids the *pirate*'s pursuit until successfully captured, if the *player* is captured automatically the game will be repeated from the beginning then the algorithm researches the original position, and if not the re-search is stopped which means the game is over.

2.3 Implementation

In this study, the *enemy* tries to find the *player's* destination point randomly and will also repeat the process after arriving at the destination point. The author tries to design 4 *enemies* using *Unity 3D*, the enemy is *pirates*. The design of the destination point *(Player)* that is tested is the *enemy* that will go to the destination point that we can see the accuracy and all enemies to the destination point *(Player)* by applying the *A Star* algorithm that has been attached to the *Enemy*.

3. Results and Discussion

3.1 Discussion

Some of the stages that will be discussed in the following discussion are data analysis, data representation, data analysis results, and the design stage.

3.1.1. Data Analysis

At this stage of the research, the *Enemy is* looking for its destination point and will continue to search until the destination point. The author designed 4 different *enemy* characters, namely *pirates*. The steps taken are to create a grid so that it can perform calculations of each route or *node* traveled, the first is to determine the value of G (n) where each value has been determined every 1 step if the straight parallel step is given a value of √1 = 1 and 1 diagonal step is given a value of √2 = 1.4 which will later be used to calculate the coordinate points of the position of the subject (*enemy) and the* object (*player*) x and y. And will also use the pythagorean formula which is formulated in the formulation: G(n). And will also use the *pythagorean* formula which is formulated as follows:

$$c^2 = a^2 + b^2$$

$$c = \sqrt{a^2 + b^2} \quad \ldots\ldots (2)$$

The pythagorean formula is useful for finding the value of the destination step or object (*player*) which will get the results of the *heuristic* value. After calculating the distance with the grid has been found, the next step is to calculate the *heuristic* value H(n) and add it to G(n) in order to get the fastest path that will be traveled and will get the F(n) value of each route, the smallest F(n) value is the intended route. Thus, onwards it will be concluded about the fastest route that can be traveled to the *Player*'s whereabouts.

3.1.2. Data Representation

In the application of the A star algorithm, the following is an explanation of the data representation of the following discussion.

1. Knowing the position between *enemy* and *player*

In this discussion, it is carried out to find out the placement of the *enemy* and *player* positions, after being known, then calculate the number of *grids*, namely the number of rows and columns determined to prove the accuracy of the *enemy* finding the right route to chase the *player*.

The positions are determined based on the grid, namely:

- *Enemy* position: Initial X = 8, Initial Y = 2
- *Player* position: X Start =1, Y End = 12
- Grid = 12 x 12

2. Calculating the position Step value

The meaning of the position value here is to determine the value of G(n) first and then perform the calculation. The rules are as follows:

- *Horizontal* and *vertical* steps √1=1, then the resulting value is 1
- The diagonal step is √2=1.4, so the resulting value is 1.4.

It should be noted that there are provisions in giving the value of G(n) in each prediction step, namely:

- Steps cannot be calculated if there is a prediction of the steps that have been passed, i.e. the smallest value of F(n) has been determined.
- Steps also cannot be calculated if the step prediction contains parts of the *grid* containing *obstacle* components.

- Every shortest route that has been found, the next route step prediction will be summed up with parallel steps or diagonal steps.

3. Determining the *heuristic* value and its calculation

In this discussion, the determination of the *heuristic* value is carried out by calculating the number of columns or rows towards the object (*player*) and calculating the number of columns or rows of object predictions that the subject (*enemy*) will aim for from each prediction of parallel or horizontal steps, after being obtained, it will be squared and then the result will be the root value, according to the *pythagorean* formula.

The distance from the subject to the road to the object is 9 rows while the distance between the object and the subject is 7 columns, so after getting the distance obtained, the *pythagorean* calculation will be carried out. The calculation method is as follows:

$C^2 = a^2 + b^2$
$C^2 = 9^2 + 7^2$
$C = 81 + 49$
$= \sqrt{130}$
$C = 11.40$

So, the *heuristic* value or H(n) value obtained is 11.40

4. Determine the steps to be taken

After getting all the step values G(n) and the *heuristic* value H(n), then in the final stage it will look for 1 route to be traveled after summing up the results of G(n) and H(n) then in each prediction step it has summed up the values of G(n) and H(n) the results are in the F(n) section as explained in the formula for the *a star* algorithm, the smallest F(n) value is the shortest route.

5. Continues the calculation to get to the destination

The previous representation is only one example in proving a *star* algorithm that is being applied in the *game*. Next, further calculations will be carried out in detail and in detail where it will determine the value of G(n) for each step added, worth one if *horizontal* and *vertical*, worth 1.4 if *diagonal*, determine the value of the *heuristic* or H(n) using the *pythagorean* formula to get the heuristic result, after being obtained, all of them are summed up from the values of G(n) and H(n) to get the value of F(n), after each step the value of F(n) is obtained, the smallest value is a step towards the *enemy*.

Step 1: The *enemy* position (8,2)

Predicted coordinates (8,1)

$G(n) = 1$
$H(n) = 11^2 + 7^2$
$= \sqrt{121 + 49}$
$H(n) = 13.03$
$F(n) = G(n) + H(n)$
$= 1 + 13.03$
$F(n) = 14.03$

Then the result of coordinate (8,1) is 14.03

Predicted coordinates (7,1)

$G(n) = 1.4$
$H(n) = 11^2 + 6^2$
$= \sqrt{121 + 36}$
$H(n) = 12.53$
$F(n) = G(n) + H(n)$
$= 1.4 + 12.53$
$F(n) = 13.92$

Then the result of coordinate (7,1) is 13.92

Predicted coordinates (7,2)

$G(n) = 1$
$H(n) = 9^2 + 6^2$
$= \sqrt{81 + 36}$
$H(n) = 11.66$
$F(n) = G(n) + H(n)$
$= 1 + 11.66$
$F(n) = 12.66$

Then the result of the coordinate (7,2) is 12.66

Predicted coordinates (7,3)

$G(n) = 1.4$
$H(n) = 9^2 + 6^2$
$= \sqrt{81 + 36}$
$H(n) = 10.81$
$F(n) = G(n) + H(n)$
$= 1.4 + 10.81$
$F(n) = 12.21$

Then the result of the coordinate (7,3) is 12.21

Predicted coordinates (8,3)

$G(n) = 1$
$H(n) = 9^2 + 7^2$
$= \sqrt{81 + 49}$
$H(n) = 11.40$
$F(n) = G(n) + H(n)$
$= 1 + 11.40$
$F(n) = 12.40$

Then the result of the coordinate (8,3) is 12.40

Step 11: The *enemy* position (5,12)

Predicted coordinates (5,11)

G(n) = (Total *enemy* position coordinates + step value)
$= 11.2 + 1$
$= 12.2$
$H(n) = 1^2 + 16^2$
$= \sqrt{1+16}$
$H(n) = 4.12$
$F(n) = G(n) + H(n)$
$= 12.2 + 4.12$
$F(n) = 16.32$

Then the result of the coordinate (5,11) is 16.32

Predicted coordinates (6,12)

G(n) = (Total *enemy* position coordinates + step value)

$= 11.2 + 1$
$= 12.2$
$H(n) = 0^2 + 4^2$
$= \sqrt{16}$
$H(n) = 4$

F(n) = G(n) + H(n)
= 12.2 + 4
F(n) = 16.2
Then the result of the coordinates (6,12) is 16.2
Predicted coordinates (4,12)
G(n) = (Total *enemy* position coordinates + step value)
= 11.2 + 1
= 12.2
H(n) = $0^2 + 2^2$
= √4
H(n) = 2
F(n) = G(n) + H(n)
= 12.2 + 2
F(n) = 14.2
Then the result of coordinates (4,12) is 14.2
Predicted coordinates (4,11)
G(n) = (Total *enemy* position coordinates + step value)
= 11.2 + 1.4
= 15.6
H(n) = $1^2 + 3^2$
= √1+9
H(n) = 3.16
F(n) = G(n) + H(n)
= 15.6 + 3.16
F(n) = 18.76
Then the result of coordinate (4,11) is 18.76
Step 12: The *enemy* position (4,12)
Predicted coordinates (4,11)
G(n) = (Total *enemy* position coordinates + step value)
= 12.2 + 1
= 13.2
H(n) = $1^2 + 3^2$
= √1+9
H(n) = 3.16
F(n) = G(n) + H(n)
= 13.2 + 3.16
F(n) = 16.36
Then the result of the coordinate (4,11) is 16.36
Predicted coordinates (5,11)
G(n) = (Total *enemy* position coordinates + step value)

= 12.2 + 1.4
= 13.6
H(n) = $1^2 + 16^2$
= √1+16
H(n) = 4.12
F(n) = G(n) + H(n)
= 13.6 + 4.12
F(n) = 17.72
Then the result of the coordinate (5,11) is 17.72
Predicted coordinates (3,12)
G(n) = (Total *enemy* position coordinates + step value)
= 12.2 + 1
= 13.2
H(n) = $0^2 + 1^2$
= √1
H(n) = 1
F(n) = G(n) + H(n)
= 13.2 + 1
F(n) = 14.2
Then the result of the coordinates (3,12) is 14.2
Predicted coordinates (3,11)
G(n) = (Total *enemy* position coordinates + step value)
= 12.2 + 1.4
= 13.6
H(n) = $1^2 + 2^2$
= √1+4
H(n) = 2.23
F(n) = G(n) + H(n)
= 13.6 + 2.23
F(n) = 15.83
Then the result of the coordinate (3,11) is 15.83
Step 13: The *enemy* position (4,12)
Predicted coordinates (3,11)
G(n) = (Total *enemy* position coordinates + step value)
= 13.2 + 1
= 14.2
H(n) = $1^2 + 2^2$
= √1+4
H(n) = 2.23
F(n) = G(n) + H(n)
= 14.2 + 2.23
F(n) = 16.43
Then the result of coordinates (3,11) is 16.43
Predicted coordinates (4,11)
G(n) = (Total *enemy* position coordinates + step value)
= 13.2 + 1.4
= 14.6
H(n) = $1^2 + 3^2$
= √1+9
H(n) = 3.16
F(n) = G(n) + H(n)
= 14.6 + 3.16
F(n) = 17.76
Then the result of coordinate (4,11) is 17.76
Predicted coordinates (2,12)
G(n) = (Total *enemy* position coordinates + step value)
= 13.2 + 1
= 14.2
H(n) = $0^2 + 0^2$
= 0
H(n) = 0
F(n) = G(n) + H(n)
= 14.2 + 0
F(n) = 14.2
Then the result of the coordinates (3,12) is 14.2
Predicted coordinates (2,11)
G(n) = (Total *enemy* position coordinates + step value)
= 13.2 + 1.4
= 14.6
H(n) = $1^2 + 1^2$
= √1+1

H(n) = 1.41
F(n) = G(n) + H(n)
= 14.6 + 1.41
F(n) = 16.01
Then the result of the coordinate (2,11) is 16.01

From the above calculations, it can be seen from each of the smallest F(n) values that can go to the route the *player* is on, very clear description of the route in the picture.

3.1.3. Data Analysis Results
After analyzing the data, the result is that the *enemy* has successfully traveled the intended route according to the rules of the *a star* algorithm. The author attaches the image and data as follows.

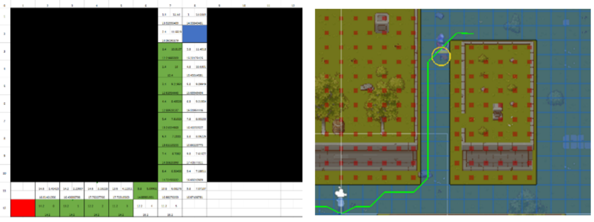

Figure 2. Comparison Results of *Enemy* Getting the Route

As for the picture above explains the display of data analysis results that will be obtained, namely the *enemy* successfully travelled the intended route according to the rules of a *star* algorithm. and for the results of the data analysis itself will be explained in table 1 below.

Table 1. Data Analysis Results

Initial Steps	G(n) value	H(n) value	F(n) value	Coordinates x	y
1	G(n) = 0 + 1.4 = 1.4	H(n) = $9^2 + 6^2$ = √81 + 36 H(n) = 10.81	F(n)=G(n)+H(n) =1.4 + 10.81 F(n) = 12.21	7	3
2	G(n) = 1.4 + 1 = 2.4	H(n) = $9^2 + 6^2$ = √64 + 36 H(n) = 10	F(n)=G(n)+H(n) = 2.4 + 10 F(n) = 12.4	7	4
3	G(n) = 2.4 + 1 = 3.4	H(n) = $7^2 + 6^2$ = √49 + 36 H(n) = 9.21	F(n)=G(n)+ H(n) = 3.4 + 9.21 F(n) = 12.61	7	5
4	G(n) = 3.4 + 1 = 4.4	H(n) = $6^2 + 6^2$ = √36 + 36 H(n) = 8.48	F(n)=G(n)+ H(n) = 4.4 + 8.48 F(n) = 12.88	7	6
5	G(n) = 4.4 + 1 = 5.4	H(n) = $5^2 + 6^2$ = √25 + 36 H(n) = 7.81	F(n)=G(n)+H(n) = 5.4 + 7.81 F(n) = 13.21	7	7
6	G(n) = 5.4 + 1 = 6.4	H(n) = $4^2 + 6^2$ = √16 + 36 H(n) = 7.21	F(n)=G(n)+H(n) = 6.4 + 7.21 F(n) = 13.61	7	8
7	G(n) = 6.4 + 1 = 7.8	H(n) = $3^2 + 6^2$ = √9 + 36 H(n) = 6.70	F(n)=G(n)+H(n) = 7.8 + 6.70 F(n) = 14.10	7	9
8	G(n) = 7.4 + 1 = 8.4	H(n) = $2^2 + 6^2$ = √4 + 36 H(n) = 6.32	F(n)=G(n)+H(n) = 8.4 + 6.32 F(n) = 14.72	7	10
9	G(n) = 8.4 + 1 = 9.4	H(n) = $1^2 + 5^2$ = √1 + 25 H(n) = 5.09	F(n)=G(n)+H(n) = 9.4 + 5.09 F(n) = 14.89	6	11
10	G(n) = 9.8 + 1.4 = 3.4	H(n) = $0^2 + 3^2$ = √9 H(n) = 3	F(n)=G(n)+H(n) = 3.4 + 3 F(n) = 14.2	5	12
11	G(n) = 11.2 + 1 = 12.2	H(n) = $0^2 + 2^2$ = √4 H(n) = 2	F(n)=G(n)+H(n) = 12.2 + 2 F(n) = 14.2	4	12
12	G(n) = 12.2 + 1 = 13.2	H(n) = $0^2 + 1^2$ = √1 H(n) = 1	F(n)=G(n)+H(n) = 13.2 + 1 F(n) = 14.2	3	12
13	G(n) = 13.2 + 1 = 14.2	0	F(n)=G(n)+ H(n) = 14.2 + 0 F(n) = 14.2	2	12

Table above shows the results of data analysis from the application of the A* algorithm in the game 'Escape from Pirates.' In the first step, the value of G(n) is 1.4, H(n) is 10.81, and F(n) is 12.21 with coordinates (7, 3). These values represent the initial cost of the chosen path and the estimated cost to the destination. In the second step, the value of G(n) increases to 2.4, H(n) remains 10, and F(n) becomes 12.4 with coordinates (7, 4). This continues until the third step with G(n) 3.4, H(n) 9.21, and F(n) 12.61 at coordinates (7, 5).

At each subsequent step, the value of G(n) continues to increase as the cost of travelling from the initial node to the current node increases, while the value of H(n) tends to decrease as the distance to the destination gets closer.

For example, in the fourth step, G(n) is 4.4, H(n) is 8.48, and F(n) is 12.88 with coordinates (7, 6).

These values continue to change until the last step, the thirteenth step, where G(n) reaches 14.2, H(n) is 0, and F(n) is 14.2 at coordinates (2, 12). This shows that the A* algorithm successfully finds a path from the starting point (7, 3) to the destination point (2, 12) by calculating the values of G(n), H(n), and F(n) at each step to determine the most efficient path.

4. Conclusion

After the testing and implementation process, the author can draw several conclusions about the game and also the application of the A Star algorithm as the development of this 2D game using the Unity 3D engine successfully implementing a star algorithm on the enemy or enemy. The enemy successfully chases the Player through the shortest route. In this game, it does not only focus on 1 enemy that applies a star algorithm, there are 4 enemy characters who can also chase the Player applying a star algorithm through the shortest route. In the calculation of a star algorithm trial table, the value of G(n) has increased compared to the value of the heuristic H(n) which has decreased and also the value of F(n) tends to experience the same value but an increase in value.

References

[1] S. M. Umran, S. Lu, Z. A. Abduljabbar, and X. Tang, “A Blockchain-Based Architecture for Securing Industrial IoTs Data in Electric Smart Grid,” *Comput. Mater. Contin.*, vol. 74, no. 3, pp. 5389 – 5416, 2023.

[2] J. Wang, G. Huang, R. Simon Sherratt, D. Huang, and J. Ni, “Data secure storagemechanism for IIoT based on blockchain,” *Comput. Mater. Contin.*, vol. 78, no. 3, pp. 4029 – 4048, 2024.

[3] Q. Zhou and G. Liu, “UAV Path Planning Based on the Combination of A-star Algorithm and RRT-star Algorithm,” in *Proceedings of 2022 IEEE International Conference on Unmanned Systems, ICUS 2022*, 2022.

[4] D. Mandloi, R. Arya, and A. K. Verma, “Unmanned aerial vehicle path planning based on A* algorithm and its variants in 3d environment,” *Int. J. Syst. Assur. Eng. Manag.*, vol. 12, no. 5, 2021

[5] K. Linvill, G. Kaki, and E. Wustrow, “Verifying Indistinguishability of Privacy-Preserving Protocols,” *Proc. ACM Program. Lang.*, vol. 7, no. OOPSLA2, 2023.

[6] Z. Xu, D. K. Jain, S. Neelakandan, and J. Abawajy, “Hunger games search optimization with deep learning model for sustainable supply chain management,” *Discov. Internet Things*, vol. 3, no. 1, 2023.

[7] W. Hohl, F. Kharvari, and G. Klinker, “Wayfinding in Museums: A Cross-sectional Comparison between 3D Serious Games and 2D Drawings as Tools for Participatory Design,” in *IEEE Conference on Computatonal Intelligence and Games, CIG*, 2020.

[8] F. Octavian and L. Hermawan, “Penerapan Algoritma Pathfinding A* dalam Game Dual Legacy berbasis Android,” *J. Buana Inform.*, vol. 14, no. 01, 2023.

[9] B. Aljabhan and M. A. Obaidat, “Privacy-Preserving Blockchain Framework for Supply Chain Management: Perceptive Craving Game Search Optimization (PCGSO),” *Sustain.*, vol. 15, no. 8, 2023.

[10] Z. Jiang, “CFD Design Optimization of a 2D Airfoil for Drag Minimization Using Parameterization Method of Mesh Motion,” in *Proceedings - 2023 7th International Symposium on Computer Science and Intelligent Control, ISCSIC 2023*, 2023.

[11] A. N. Putri, M. Hariadi, and R. F. Rachmadi, “Supply Chain Management Serious Game Using Blockchain Smart Contract,” *IEEE Access*, vol. 11, pp. 131089 – 131113, 2023.

[12] B. Cao, Z. Yang, L. Yu, and Y. Zhang, “Research on the star algorithm for safe path planning,” in *2023 IEEE International Conference on Control, Electronics and Computer Technology, ICCECT 2023*, 2023.

[13] X. Xu and M. Zhoya, “Path Planning in Multi-AGVs Using a Modified A-star Algorithm,” *Int. J. Innov. Educ. Res.*, vol. 8, no. 5, 2020.

[14] J. A. Parra-Espín, R. Marín-López, G. López-Millán, F. Pereñíguez-García, and O. Canovas, “SDN-based automated rekey of IPsec security associations: Design and practical validations,” *Comput. Networks*, vol. 233, 2023.