Optimizing Mobile Robot Navigation Based on A-Star Algorithm for Obstacle Avoidance in Smart Agriculture

Antonios Chatzisavvas *, Michael Dossis and Minas Dasygenis

Abstract: The A-star algorithm (A*) is a traditional and widely used approach for route planning in various domains, including robotics and automobiles in smart agriculture. However, a notable limitation of the A-star algorithm is its tendency to generate paths that lack the desired smoothness. In response to this challenge, particularly in agricultural operations, this research endeavours to enhance the evaluation of individual nodes within the search procedure and improve the overall smoothness of the resultant path. So, to mitigate the inherent choppiness of A-star-generated paths in agriculture, this work adopts a novel approach. It introduces utilizing Bezier curves as a postprocessing step, thus refining the generated paths and imparting their smoothness. This smoothness is instrumental for real-world applications where continuous and safe motion is imperative. The outcomes of simulations conducted as part of this study affirm the efficiency of the proposed methodology. These results underscore the capability of the enhanced technique to construct smooth pathways. Furthermore, they demonstrate that the generated paths enhance the overall planning performance. However, they are also well suited for deployment in rural conditions, where navigating complex terrains with precision is a critical necessity.

Keywords: path planning; optimizing; UGV; automobiles; unmanned ground vehicles; smart agriculture; A-star algorithm

1. Introduction

In an era marked by ever-advancing technology and the increasing integration of automation into diverse industries, the need for efficient and precise route planning has never been more critical. Whether it is guiding autonomous robots through complex environments, optimizing the routes of delivery vehicles in logistics, or even assisting in planning systems for smart agriculture, the A-star algorithm stands as a time-tested and versatile approach. Its widespread adoption spans various fields, from robotics to the automotive industry and beyond [1–4]. The A-star algorithm’s exceptional ability to navigate intricate spaces and calculate optimal paths has streamlined operations and yielded significant cost savings. Reducing path times has become an invaluable tool in facilitating automation across industries. Moreover, its role in route planning has played a pivotal role in enhancing overall productivity, thereby ensuring that tasks are carried out efficiently and in their actuality, thus driving progress in the modern age of technology and automation [5–9].

As the global demand for food production escalates to meet the needs of a burgeoning population, the agriculture sector faces an imperative to innovate and optimize its practices. In this pivotal moment, the A-star algorithm emerges as a critical ally that is poised to revolutionize farming. Across the vast expanses of modern farms, which are characterized by diverse terrains and varying degrees of complexity, the ability to navigate efficiently is a significant challenge. From the tractors that till the fields to the autonomous drones that monitor crops from above, agricultural machinery must traverse these landscapes with...
precision and efficacy. The A-star algorithm, renowned for route planning and pathfinding, is pivotal in enabling these machines to perform their tasks efficiently [10–14]. By leveraging the A-star algorithm, agricultural equipment can map out optimal paths, thus avoiding obstacles and minimizing soil compaction. This enhances the productivity of farming operations and contributes to sustainable practices by reducing environmental impact. In essence, the A-star algorithm is reshaping the agriculture landscape, thereby ensuring that the industry can rise to the challenge of feeding the world’s growing population while preserving the land and resources it relies on [15–18].

Nevertheless, the A-star method, although it is adaptable and often used, does have its constraints. One specific difficulty it deals with is the creation of routes that often lack the necessary smoothness. This inadequacy is particularly noticeable when used in agricultural settings, where the landscape might be rough and constantly changing, and the need for uninterrupted and secure movement is of utmost importance. The intrinsic unevenness of pathways provided by the A-star algorithm might be a substantial obstacle in the practical implementation of autonomous vehicles and robotic systems in challenging agricultural settings. The sudden changes in both trajectory and velocity may result in inefficiencies, heightened mechanical deterioration, and potential safety hazards, especially when operating with substantial and costly machinery [19–21]. Addressing this limitation is of utmost importance as we seek to harness the full potential of the A-star algorithm in agriculture. Finding innovative solutions to enhance path smoothness and adaptability to challenging terrains is an ongoing endeavor. Researchers and engineers are working diligently to refine the algorithm, thereby introducing modifications and supplementary techniques to ensure that it remains a valuable asset in the modernization and optimization of agricultural practices. Overcoming this obstacle will further solidify the A-star algorithm’s position as a transformative force in the agricultural industry, thus facilitating precision farming and sustainable food production [22–25].

The overarching objective of this work is to optimize the efficacy of the A-star algorithm and, concurrently, to refine the quality of the resultant path it produces:

- In order to enhance the search performance of the A-star algorithm and optimize its path planning capabilities, a dynamic weighting mechanism is added. This approach enables the modification of the specific gravity of each component throughout the algorithm’s assessment phase. The use of dynamic weighting allows the algorithm to adjust and give priority to certain components of route planning based on the environment and needs of the application. Furthermore, the heuristic function, a crucial component in directing the search process, is carefully adjusted to guarantee its correct representation of the environment. This process of fine-tuning allows for more knowledgeable and situation-aware choices to be made at the phase of determining the route.

- The A-star technique incorporates the use of Bezier curves, a well-known mathematical tool, during the postprocessing stage. This phase is crucial in improving the limitations of the created route. Bezier curves provide the inherent capacity to create a smooth route by gradually and continuously curving via the manipulation of control points. The control points, which are strategically modified according to the waypoints of the A-star route, serve as a transformational influence, thereby altering the originally uneven trajectory into a far smoother and more appropriate path for practical use.

- To ascertain the efficacy of our method, we subject our enhanced technique to a battery of tests within the context of rural conditions. The simulation results stand as a testament to the viability and potential of our approach, thus demonstrating not only its ability to address the perennial issue of path smoothness but also its capacity to augment planning performance in the demanding and ever-changing landscape of smart agriculture.

This study’s remaining units are structured as follows: In Section 2, we delve into the analysis papers that form the foundation of our study. In Section 3, we clarify our optimized A-star algorithm. Section 4 demonstrates the investigation outcomes derived
from applying our optimized A-star algorithm. Finally, in Section 5, we summarize our work’s effects.

2. Related Work

Continuous efforts are being made to find novel solutions that may improve the A-star algorithm’s smoothness and flexibility when applied to difficult terrain. Researchers and experts are dedicated to refining this algorithm, exploring new avenues for improvement, and introducing modifications and complementary techniques that cater to the agricultural industry. The end objective is to support the implementation of precision agricultural technologies and promote sustainable food production. These efforts aim to minimize the choppiness in generated paths, thereby ensuring that autonomous vehicles and robotic systems can navigate agricultural environments more efficiently [26–29].

The study’s authors [30] modified the heuristic procedure of the typical A-star algorithm in order to decrease the amount of time spent searching and the number of routes that are required to be explored. The experimental findings revealed that the enhanced A-star algorithm route planning approach was 5.34% shorter than the typical A-star algorithm in terms of the average distance period along the path and that the duration of the period of time needed was reduced by 22.56% on average. This has the potential to enhance the efficiency of robot route planning, as well as to level out the trajectory of driving action while simultaneously ensuring that the best path is taken.

The investigation’s authors [31] enhanced the A-Star algorithm because the performance of the typical A-Star algorithm is suboptimal for the space, time, and number of search routes, thus relying on the robot motion block. However, this algorithm’s effectiveness is often hampered by factors such as the total number of search routes, the overall cost of the path, and the algorithm’s complexity of time, especially when tailored to a specific robot motion block. This work has delved into optimizing the robot motion block to mitigate the exploration of redundant routes, thereby streamlining the search process. According to the investigations, the suggested procedure gained 93.98% in the number of search routes and 98.94% in time complexity in comparison to the typical A-Star algorithm.

The bidirectional alternating search A-star was presented by the authors in the article [32]. This particular search algorithm was developed for mobile robots. The goal of this method is to generate pathways that are not only efficient but also optimum. It does this by combining the qualities of the traditional A-star algorithm with those of bidirectional search approaches. The core of this strategy is to perform searches in both forward and backward directions in an alternating fashion until these routes cross, which will result in an increase in the efficiency of the exploration process. This technique addresses problems that arise in large task areas, excessive turning angles, longer calculation durations, and nonsmooth pathways.

The authors of the study [33] optimized the heuristic approach of the conventional A-star algorithm in order to reduce the amount of time that was spent searching, as well as the number of various pathways that were required to be evaluated. Following the completion of the trials, it was determined that the improved algorithm was able to decrease the amount of time required for route planning by an average of 9.11% while concurrently lowering the distance travelled by the route by an average of 9.29%. There was an improvement in the performance of the A-star algorithm, notably with regard to the functional efficiency of its paths and the length of its routes.

The researchers in the study [34] enhanced the A-Star algorithm to optimize the navigation of mobile robots. Their robot was outfitted with LiDAR and an Inertial Measurement Unit for accurate environmental mapping. A mapping algorithm created a two-dimensional grid map, which was then utilized by the advanced A-Star algorithm for the robot’s path planning. The algorithm introduced a strategy for making the path smoother, thus incorporating a safety mechanism and removing unnecessary points and sharp corners by evaluating potential obstacles between path segments. These improvements significantly increased the path’s smoothness, thereby making the robot’s movement more efficient.
in practical applications. The average time needed for path searching was reduced by 13%, and the number of nodes required for path extension was reduced by 11%, thereby addressing the traditional A-Star algorithm’s issues with excessive turning points and slow search speeds.

This work introduces enhancements to the A-star algorithm by incorporating the Manhattan distance, dynamic weight coefficients, and Bezier curves. The Manhattan distance heuristic is specifically tailored to improve pathfinding precision within the context of agricultural environments. Using dynamic weight coefficients allows the algorithm to adjust flexibly to the varied situations and environmental challenges typical in agricultural settings. By integrating Bezier curves, the paths generated are shorter and smoother. This is particularly beneficial in agriculture, where equipment often must navigate challenging terrains, thus requiring paths supporting continuous and safe movement.

3. Implementation

This section examines the foundational components of the typical A-star algorithm, where we introduce the Manhattan distance and dynamic weight coefficients. These enhancements not only address the issue of path smoothness but also empower the A-star algorithm to adapt dynamically to various scenarios and environmental constraints. Furthermore, we analyze the Bezier curves that bring smoothness to generated paths. By leveraging the flexibility of Bezier curves, the optimized A-star algorithm achieves paths that are not only shorter but also smoother—making them ideal for real-world agricultural operations that demand continuous and safe motion.

3.1. Typical A-Star Algorithm

The A-star algorithm is a widely employed and powerful pathfinding algorithm in computer and artificial intelligence science. It is particularly popular in applications where finding the fastest path between two points in a grid or graph is crucial, such as robotics and navigation systems. It finds the shortest path by assessing the cost of a particular route and estimating the remaining cost to reach the goal route. A heuristic function provides this estimate, usually denoted as $H(z)$, and it guides the search towards the most promising paths, thus ultimately leading to a more efficient exploration of the search space. However, choosing an appropriate heuristic function is critical to the algorithm’s performance. The quality of the heuristic greatly influences the algorithm’s speed and accuracy. A well-designed heuristic can significantly reduce the search space, thus leading to faster pathfinding [35–37]. We introduce the Manhattan distance as a specific implementation of the heuristic function $H(z)$ to enhance the A-star algorithm and incorporate dynamic weight coefficients. These modifications improve path smoothness and adaptability to different scenarios, thereby enhancing the algorithm’s overall performance.

The implementation of the typical A-star algorithm involves several crucial stages, with each contributing to its performance and effectiveness in route planning:

- Define open and closed lists: The open list maintains the routes ready for inspection and potential inclusion in the path, while the closed list stores routes assessed or checked during the search procedure.
- Define parent nodes: The A-star algorithm designates a parent route for each node in the search space. Routes neighbouring the parent route are identified as child routes and are added to the open list for further investigation.
- Path selection: Path selection is the crux of the A-star algorithm’s decision-making process. This selection hinges on the evaluation function, which is typically represented in Equation (1):

$$F(z) = G(z) + H(z)$$

Here, $F(z)$ stands for the evaluation function of route $z$. $G(z)$ signifies the real cost of traveling from the initial route to route $z$ within the state space. In contrast, $H(z)$ represents the calculated cost of navigating from route $z$ to the target route while
disregarding obstacles. In this work, we use the Manhattan distance as the specific heuristic function $H(z)$, thus enhancing the heuristic’s simplicity and effectiveness in grid-based environments.

- **Heuristic function $H(z)$**: The A-star algorithm’s efficacy heavily relies on the heuristic function, which is denoted as $H(z)$ in the evaluation function. This function estimates the cost required to earn the target route from the evaluated current route. It is important to note that $H(z)$ should be admissible, meaning that it never overestimates the actual cost. The choice and design of the heuristic function can significantly affect the algorithm’s evaluation, as a well-chosen heuristic can guide the search efficiently towards the goal. Here, we introduce dynamic weight coefficients within the heuristic to adapt to varying environmental constraints, thereby ensuring more flexible and effective pathfinding.

- **Open list priority queue**: To efficiently select nodes from the open list, many implementations use a priority queue data structure. This data structure ensures that nodes with the lowest evaluation function values $F(z)$ are chosen first for further exploration. This prioritization aids in reaching the goal route more quickly by evaluating the most promising paths earlier in the search.

- **Termination condition**: The A-star algorithm continues its search until it reaches the goal route or exhausts all possible paths without finding a solution. Properly handling termination conditions is essential to avoid unnecessary computation and ensure the algorithm’s efficiency.

- **Reconstruction of the optimal path**: Once the A-star algorithm reaches the goal route, it reconstructs the optimal path by backtracking the parent routes from the goal route to the initial route. Considering actual and estimated costs, this path represents the most efficient and shortest route from the start to the terminus.

3.2. Manhattan Distance

The Manhattan distance is widely used for measuring the distance between two points in a grid-based system. In computer science, it is commonly used in algorithms like the A-star search algorithm for pathfinding, where it serves as a specific heuristic function $H(z)$ to estimate the cost of reaching a goal state. Its simplicity makes it a valuable tool for solving problems related to route planning, logistics, and optimization in grid-like settings, thereby contributing to its widespread adoption in practical applications [38–40].

The Manhattan distance as a specific implementation of the heuristic function $H(z)$ is calculated as the sum of the absolute differences in the horizontal and vertical coordinates between the current route ‘$z$’ and the goal route. Mathematically, $H(z)$ can be expressed in Equation (2):

$$H(z) = |CurrentRouteX - GoalRouteX| + |CurrentRouteY - GoalRouteY|$$  \hspace{1cm} (2)

In this context, ‘CurrentRouteX’ and ‘CurrentRouteY’ are the coordinates of the current node being evaluated, and ‘GoalRouteX’ and ‘GoalRouteY’ are the target or goal route coordinates. By incorporating the Manhattan distance as the heuristic function $H(z)$, the A-star algorithm will prioritize paths that minimize the total horizontal and vertical movement, thus aligning with the characteristics of grid-based systems where diagonal movement is not allowed.

3.3. Dynamic Weight Coefficients

One of the key and versatile features of the A-star algorithm is its capacity to adjust the weight coefficients within its evaluation function dynamically. This adaptability empowers the algorithm to fine-tune its behavior and decision-making process according to various applications’ specific requirements and constraints. The performance function is represented in Equation (3):

$$F(z) = G(z) + k \times H(z)$$  \hspace{1cm} (3)
In this formula, ‘k’ serves as the weight coefficient for the heuristic function \( H(z) \), while \( G(z) \) symbolizes the actual cost of reaching the current route from the start. By varying the value of ‘k’, the algorithm can modulate the influence of the heuristic component relative to the actual cost in determining paths. Increasing the weight coefficient ‘k’ emphasizes the evaluation function’s heuristic function \( H(z) \). Conversely, reducing the weight coefficient ‘k’ reduces the significance of the heuristic function, thereby making the algorithm more exploratory in nature.

It is possible to utilize dynamic weighting in the weighting coefficient in order to guarantee the best route while also increasing the effectiveness of the search, and we defined it as follows:

• Estimated cost (EC): EC represents the estimated cost of disregarding impediments when moving from the present search route ‘z’ to the target route. It is a heuristic function similar to \( H \) in the typical A-star algorithm.
• Dynamic weighting coefficient ‘k’: This is introduced to adaptively modify the weight assigned to the A-star algorithm’s heuristic component based on the EC value.

The dynamic weighting process is as follows:
If the EC is greater than 18, set ‘k’ to 3. Otherwise, set ‘k’ to 0.85.

This dynamic adjustment of ‘k’ ensures that the A-star algorithm behaves differently depending on the estimated cost of ignoring obstacles. When the EC is high (greater than 18), ‘k’ is set to a higher value of 3. This means the algorithm will prioritize paths closer to the target route, thus leading to faster convergence when the heuristic function is reliable. Conversely, when the EC is low (18 or less), ‘k’ is set to a lower value of 0.85. This dynamic adjustment of ‘k’ depends on the specific requirements and characteristics of the agricultural robot’s task and environment:

• Reliability of the heuristic: When the estimated cost of ignoring obstacles (EC) is high, it suggests that the heuristic function is generally reliable. The terrain is predictable in agricultural environments, and obstacle information is accurate. In such cases, setting ‘k’ to 3 means the algorithm heavily relies on the heuristic function, thereby allowing it to converge quickly to a solution.
• Speed vs. accuracy: Prioritizing the heuristic function by setting ‘k’ to 3 is an advantage when speed is critical. For agricultural robots, navigation efficiency is essential, especially when the robot needs to reach multiple points quickly, such as for irrigation or harvesting tasks.
• Avoiding overexploration: Setting ‘k’ to a lower value, like 0.85, when the EC is low helps avoid overexploration in situations where the environment is highly unpredictable or the heuristic function is less reliable. Overexploration leads to unnecessary computation and slower decision making when the terrain and obstacles are well known.
• Adaptability: The ability to dynamically adjust ‘k’ based on the EC allows the agricultural robot to adapt to changing environmental conditions. For instance, if weather conditions change and the reliability of obstacle data decreases, the algorithm can automatically switch to a more cautious and exploratory mode by reducing ‘k’.

3.4. Bezier Curves

Bezier curves are mathematical curves that have extensive application in two-dimensional graphical domains. Various Bezier curves exist within mathematics, thus encompassing different orders and distinctive characteristics. These curves include the first-order Bezier, second-order Bezier, and an array of higher-order variants. Each Bezier curve is uniquely determined by a group of critical points, specifically the starting, end, and control points. The configuration and arrangement of these control points play a pivotal role in shaping the ultimate form of the curve. While the first-order Bezier curve simplifies to a straight line, higher-order Bezier curves offer diverse shapes and patterns, thus making them versatile tools for graphical representation [41–43].
This study focuses on harnessing the potential of multiorder Bezier curves to enhance the route generated by the typical A-star algorithm. Introducing Bezier curves into the postprocessing phase aims to impart a smoother and more refined character to the resultant path. This augmentation is made possible by leveraging the control points within the Bezier curve formulation, thus allowing for the fine-tuning and optimization of the generated path. Through the strategic placement and manipulation of these control points, the study endeavors to achieve a path that excels in length and exhibits a high degree of smoothness, thus aligning it more closely with the demands of real-world applications that necessitate fluid and safe motion.

The formula for a Bezier curve of degree m (where m is the order of the curve) is typically expressed in Equation (4):

$$B(r) = \sum_{i=0}^{m} (P_i \ast B_{i,m}(r))$$

(4)

In this formula:
- $B(r)$ represents the point on the Bezier curve at parameter value ‘r’.
- $P[i]$ symbolizes the control points of the Bezier curve. There are ‘m + 1’ control points indexed from 0 to n.
- $B[i,m]$ denotes the Bernstein basis polynomial of degree n for control point $P[i]$. It depends on ‘r’ and is used to interpolate between control points.

The Bernstein basis polynomial can be calculated in Equation (5):

$$B_{i,m}(r) = C(m, i) \ast (1 - r)^{m-i} \ast r^i$$

(5)

where $C(m, i)$ is the binomial coefficient, which is calculated in Equation (6):

$$C(m, i) = \frac{m!}{i! \ast (m-i)!}$$

(6)

This formula allows you to calculate the position of a point on a Bezier curve for a given parameter value ‘r’ by summing up the contributions from each control point, with each weighted by the corresponding Bernstein basis polynomial. The parameter ‘r’ typically ranges from 0 to 1, thus representing points along the curve. The span of the Bezier curve ‘m’ determines the whole number of control points and influences the shape of the curve. Higher-degree Bezier curves can represent more complex and intricate shapes.

3.5. Unmanned Ground Vehicle (UGV)

Given the often uneven terrain it must navigate, a robust and sturdy design is essential. Ensuring the vehicle’s stability and minimizing any oscillations impeding its movement becomes a critical focus, thus achieving this required meticulous attention to maintaining the center of gravity within optimal parameters. Moreover, the vehicle’s proportions were carefully modified to cater to the specific needs of its intended purpose, which primarily involves traversing narrow crop aisles. This adaptation allows for seamless movement through the fields, thus ensuring efficient and damage-free passage without compromising safety. One noteworthy feature enhancing the vehicle’s versatility is its detachable main body. This innovation facilitates easy transport between fields or locations, thus making it a versatile and practical choice for various agricultural applications. The detachable body simplifies the logistics of moving the vehicle and allows for efficient transportation to different work sites. Additionally, the frame of the vehicle, as depicted in Figure 1, was constructed from robust metal materials. This choice was made to enhance the vehicle’s durability and longevity, thus ensuring it can withstand the rigors of the often harsh operating environments encountered in agricultural settings. By selecting metal for the
frame, we have significantly increased the vehicle’s resistance to environmental stressors, corrosion, and other factors that could compromise its structural integrity.

Figure 1. The frame of the UGV.

The vehicle’s propulsion system is designed to be powered by direct current (DC) motors, which is a choice made for their precision and versatility in controlling movement. These DC motors are equipped to gauge rotations accurately, thereby allowing them to calculate the distances traveled with a high degree of accuracy. This precision is crucial in an agricultural context where precise control over the vehicle’s movements is essential to ensure proper planting and cultivation. In addition to the choice of propulsion system, the manner of transmission underwent specific modifications to align with the unique characteristics of the robotic vehicle and the requirements of planting and cultivating the target crop. Agricultural fields often pose challenges such as narrow rows or uneven terrain, which demand a transmission system that can adapt to these conditions. The modified transmission system considers the vehicle’s size and dimensions, thus ensuring that it can navigate the tight spaces between rows and efficiently cover the designated area for planting and cultivation. This tailored approach to transmission optimizes the vehicle’s performance and enhances its overall efficiency and effectiveness in the agricultural tasks it is intended for. By customizing the transmission system to suit the specific needs of planting and cultivating crops, we aimed to maximize productivity while minimizing waste, thus ultimately contributing to more sustainable and efficient agricultural practices in Figure 2.

Facilitating seamless navigation through the dense vegetation of the crop, the vehicle boasts an articulated arm equipped with five degrees of freedom. This advanced arm design provides the necessary dexterity and flexibility to efficiently perform various tasks within the agricultural environment, from planting to cultivation and other precision operations. Many obstacle avoidance sensors and cameras were strategically installed around the vehicle’s perimeter to ensure safe and precise navigation through the fields while avoiding potential obstacles. These sensors and cameras work in tandem to continuously monitor the surroundings, thus enhancing the vehicle’s situational awareness and enabling it to make real-time adjustments to its path in order to avoid collisions and damage to the machine and the crops it operates within.

Global positioning system (GPS) receivers were integrated into the vehicle’s system for accurate positioning and route planning. This technology enables the vehicle to pinpoint its location, thus aiding in autonomous navigation and ensuring that it operates precisely
where needed within the agricultural landscape. The quality of GPS signals in outdoor, unstructured environments can vary depending on factors such as terrain, vegetation, and atmospheric conditions. However, for most agricultural applications where high precision is not critical, the GPS solution integrated into the vehicle’s system should be sufficient and satisfying.

Regarding safety, an emergency button has been thoughtfully incorporated into the vehicle’s design. This safety feature is a fail-safe mechanism, thereby allowing users to instantly shut off the electricity supply to the vehicle’s various components in the case of an unforeseen emergency or hazard. This immediate shutdown capability prevents accidents and protects the vehicle and its operators. Furthermore, the vehicle has strategically installed light-emitting diode (LED) lights for easy field identification and tracking. These lights make it more straightforward to locate the vehicle within the vast expanse of the field, even in low-light conditions. This feature enhances operational efficiency, as it aids in quickly finding and retrieving the vehicle when needed, thus contributing to a smoother and more productive agricultural workflow, which is presented in Figure 3.

Figure 2. UGV in agricultural environment.
3.6. Optimized A-Star Algorithm in Smart Agriculture

The optimized A-star algorithm, which incorporates the Manhattan distance, dynamic weight coefficients, and Bezier curves, is presented in Algorithm 1 and Figure 4. This advanced algorithm has been designed to address path planning challenges by dynamically adapting to different scenarios and environmental constraints. It leverages the power of the Manhattan distance as a heuristic, adjusts weight coefficients on the fly, and utilizes Bezier curves to generate smoother and more efficient routes. Algorithm 1 demonstrates how these enhancements optimize A-star route planning, thus making it suitable for real-world applications that demand precise and adaptable pathfinding.

In smart agriculture, the benefits of this enhanced A-star algorithm are particularly noteworthy. With the ability to adapt to changing field conditions, it becomes an invaluable tool for autonomous farming. The A-star algorithm can efficiently plan routes for agricultural machinery, such as tractors and harvesters, thus considering dynamic factors like crop growth and changing terrain. Optimizing routes using Bezier curves reduces unnecessary wear and tear on equipment and minimizes fuel consumption, thus contributing to cost savings and environmental sustainability. Furthermore, the A-star algorithm’s dynamic weight coefficients enable it to prioritize different objectives, such as minimizing travel time or reducing soil compaction, depending on the farmer’s goals and the specific needs of the field. This adaptability ensures that smart agriculture practices can be fine-tuned for optimal results.

Integrating these advanced techniques in the A-star algorithm showcases its capacity to address the unique challenges faced in modern agriculture. Whether navigating intricate routes or optimizing planting patterns, the enhanced A-star algorithm offers a versatile and efficient solution. Its adaptability and precision align perfectly with the demands of precision agriculture, thereby making it an indispensable asset for farmers striving to increase yields, reduce costs, and promote sustainability.
Algorithm 1 A-star algorithm with Manhattan distance, dynamic weight coefficients, and Bezier curves

Require: startNode, goalNode, map, controlPoints

1: Initialize open and closed lists
2: Initialize dynamic weight coefficient $k \leftarrow 0.85$
3: Create a map for backtracking optimal path: cameFrom
4: Initialize costs for startNode:
5: $gScore[startNode] \leftarrow 0$
6: $fScore[startNode] \leftarrow gScore[startNode] + k \cdot \text{calculateManhattanDistance}(startNode, goalNode)$
7: Add startNode to open list with $fScore$ value
8: while openList is not empty do
9:     current $\leftarrow$ node with lowest $fScore$ in open list
10:    if current is the goalNode then
11:        return reconstructPath(cameFrom, current)
12:    end if
13:    Move current node from open to closed list
14:    for each neighbor in getNeighbors(current) do
15:        if neighbor is in closed list then
16:            {Skip neighbors already evaluated}
17:        end if
18:        Calculate tentative $gScore$:
19:        tentativeG $\leftarrow gScore[current] + \text{distance}(current, neighbor)$
20:        if neighbor is not in open list or tentativeG $< gScore[neighbor]$ then
21:            Update the path to this neighbor:
22:            cameFrom[neighbor] $\leftarrow$ current
23:            gScore[neighbor] $\leftarrow$ tentativeG
24:        Calculate EC (Estimated Cost) using Bezier curves:
25:        EC $\leftarrow$ calculateBezierCurveEC(neighbor, goalNode, controlPoints)
26:        if EC $> 18$ then
27:            $k \leftarrow 3$
28:        else
29:            $k \leftarrow 0.85$
30:        end if
31:        Update $fScore$ for neighbor:
32:        $fScore[neighbor] \leftarrow gScore[neighbor] + k \cdot \text{calculateManhattanDistance}(neighbor, goalNode)$
33:        if neighbor is not in open list then
34:            Add neighbor to open list with updated $fScore$
35:        end if
36:    end for
37: end while
38: return failure // No path found
Figure 4. Diagram of A-star algorithm with Manhattan distance, dynamic weight coefficients, and Bezier curves.

4. Investigation Outcomes

All experiments with the A-star algorithm were executed on an Intel NUC i5 with 32 GB of RAM utilizing the Python language. Figure 5 depicts the A-star algorithm, which has been enhanced with the weight coefficients and the Manhattan distance as a heuristic. Figure 6 illustrates the algorithm’s prowess enhanced with the Manhattan distance and weight coefficients, as well as the integration of multiorder Bezier curves.

The outcomes depicted in Figures 5 and 6 were generated using simulations. These simulations were conducted to demonstrate the effectiveness of our proposed enhancements to the A-star algorithm, specifically the incorporation of the Manhattan distance heuristic and dynamic weight coefficients. The simulation environment was a grid-based system that was used for pathfinding algorithms. The grid cells represent possible positions the robot can occupy, and obstacles are represented by blocked cells. The green circle represents the start position, and the red circle represents the goal position. The green rectangles represent obstacles that the robot must navigate around. The placement and size of these obstacles were designed to create a pathfinding challenge that demonstrates the algorithm’s capability to handle complex environments. The Manhattan distance was used as the heuristic function $H(z)$ to estimate the cost to reach the goal from the current node. The weight coefficient $k$ was adjusted based on the estimated cost $EC$ described in Section 3.3. Specifically, $k$ was set to 3 when the $EC$ was greater than 18 and to 0.85 when
the EC was 18 or less. This dynamic adjustment was implemented to balance the trade-offs between exploration and exploitation in the pathfinding process. The A-star algorithm was run on the grid to find the optimal path from the start to the goal position. The algorithm considered the actual cost $G(z)$ and the heuristic function $H(z)$ to evaluate potential paths. The resulting path was visualized on the grid, thus showing the sequence of moves from the start to the goal position while avoiding obstacles. Figure 5 shows the path generated without dynamic weight coefficients, thus resulting in a more angular path. Figure 6 demonstrates the improved path smoothness achieved by incorporating dynamic weight coefficients and Bezier curves. The layout in Figures 5 and 6 is a simplified representation of an agricultural field. The start and goal positions represent the robot’s initial position and target point in the field, respectively. The obstacles simulate rows of crops or other impassable areas that the robot must navigate. The grid-based abstraction simplifies the complex agricultural environment into manageable units for algorithm testing.

![Figure 5. Dynamic weight coefficients and the Manhattan distance (Green Circle: Starting Point, Red Circle: End Point, Black Line: Path, Green Parallelogram: Obstacles).](image)

The comparative analysis presented in Table 1 elucidates the significant enhancements realized through the optimization of the A-star algorithm. This analysis spans critical performance metrics, including path length, computation time, and the number of search routes, thereby highlighting the profound impact of the optimization process. Prior to optimization, the typical A-star algorithm yielded paths with an average length of 22.97 units. The implementation of optimization techniques markedly reduced this path length to 16.34 units. This notable decrease underscores the efficacy of the enhancements, as shorter paths inherently contribute to more efficient and resource-conserving route planning, which is a crucial factor in applications demanding optimal performance. In terms of computational efficiency, the optimization achieved remarkable improvements. The computation time required by the typical A-star algorithm was initially recorded at 1.653 milliseconds. Postoptimization, this time was significantly curtailed to 0.912 milliseconds. This reduction in computation time is critical, particularly in real-time applications, as it significantly enhances the algorithm’s responsiveness and agility.
Furthermore, the optimization had a substantial impact on the number of search routes generated during the planning process. The typical A-star algorithm generated 361 search routes, which are indicative of a comprehensive exploration of the search space. In contrast, the optimized A-star algorithm streamlined this process, thereby generating only 122 search routes. This reduction signifies a more focused and efficient exploration of the search space, thereby further validating the effectiveness of the optimization. The ability to condense the number of search routes without compromising the quality of the pathfinding solution underscores the effectiveness of the optimization. This improvement not only enhances computational efficiency but also ensures that the algorithm can operate more effectively in time-sensitive and resource-constrained environments, thereby validating the practical applicability and superiority of the optimized method.

Figure 6. Manhattan distance, dynamic weight coefficients, and the integration of multiorder Bezier curves (Green Circle: Starting Point, Red Circle: End Point, Black Line: Path, Green Parallelogram: Obstacles).

Table 1. Comparison of typical A-star and optimized A-star path length, computation time, and the number of search routes.

<table>
<thead>
<tr>
<th></th>
<th>Typical A-Star</th>
<th>Optimized A-Star</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length (units)</td>
<td>22.97</td>
<td>16.34</td>
</tr>
<tr>
<td>Time (ms)</td>
<td>1.653</td>
<td>0.912</td>
</tr>
<tr>
<td>Routes</td>
<td>361</td>
<td>122</td>
</tr>
</tbody>
</table>

5. Conclusions and Future Work

The A-star algorithm is a tried solution for route planning across many domains, thus encompassing fields as diverse as robotics, automotive navigation, and agriculture. However, its inherent limitation in producing sufficiently smooth paths has prompted a dedicated investigation into its enhancement within agriculture. This study has refined the A-star algorithm to overcome this challenge by focusing on fine-tuning individual node evaluations and, more importantly, the overall path smoothness. The introduction of Bezier curves as a postprocessing step in this research has emerged as a transformative
intervention, thereby yielding pathways that are not only shorter but also notably smoother. By seamlessly integrating Bezier curves into the path generation process, this study has effectively bridged the gap between the A-star algorithm’s traditional outputs and the practical requirements of modern agriculture. The simulation’s findings validate the suggested approach’s effectiveness and illustrate that the improved strategy can create smooth paths while simultaneously enhancing planning performance in rural situations. Our future study will explore optimizing Bezier curve parameters to strike an even finer balance between path smoothness and path length. Additionally, integrating real-time data and sensor feedback into the A-star algorithm could improve its performance in dynamic agricultural environments. Finally, the scalability of the enhanced A-star algorithm should be examined to ensure its applicability across different types and sizes of agricultural fields and machinery.

**Author Contributions:** Validation, A.C.; project administration, A.C.; conceptualization, A.C.; resources, A.C.; visualization, A.C.; formal analysis, A.C.; methodology, A.C.; writing—review and editing, A.C.; investigation, A.C.; software, A.C.; writing—original draft preparation, A.C., M.D. (Michael Dossis), and M.D. (Minas Dasygenis); supervision, M.D. (Minas Dasygenis). All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data are contained within the article

**Conflicts of Interest:** The authors declare no conflicts of interest.

**Abbreviations**

The following abbreviations are used in this manuscript:

- DC Direct Current
- EC Estimated Cost
- LED Light-Emitting Diode
- GPS Global Positioning System
- RAM Random Access Memory
- UGV Unmanned Ground Vehicle

**References**


**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.