Optimizing Pass Selection Algorithms in eFootball: A Graph-Theoretic Approach with Complexity Analysis

Hakam Avicena Mustain - 13524075
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: hakammustain@gmail.com , 13524075@std.stei.itb.ac.id

***Abstract*— This paper presents a graph approach to modeling pass selection in football simulation games in context of eFootball. The simulation calculates passing decisions based on five contextual factors: inter-player distance, opponent density, team tactics, player playing style, and AI behavior style. Each possible pass is represented as a directed edge in a weighted graph, and a scoring function is applied to determine the most optimal passing option. The simulation is implemented using TypeScript and React, with real-time visualizations of player positioning and pass evaluation. Computational complexity is also analyzed, with the algorithm shown to operate at an efficient O(n × m) per frame, where *n* is the number of teammates and *m* is the number of opponents. This study demonstrates the way discrete mathematics and algorithm analysis can be applied to real-time sports simulations to tactical decision-making.**

Keywords—graph theory; pass selection; football simulation; discrete mathematic; algorithm complexity

I. Introduction

In recent years, Artificial Intelligence (AI) has become as one of the most transformative technologies, enabling machines to replicate complex human capabilities such as learning, reasoning, decision-making, and autonomy [1]. Across numerous domains, AI is driving advances in performance and user experience. Especially, the gaming industry has seen rapid growth in the integration of AI to model human-like behavior, adapt to player skill, and deliver engaging, dynamic interactions.

Football (also known as association football or soccer) presents a compelling case study for AI implementation due to its continuous spatial and tactical complexity. In this sport, each the eleven players on a team must constantly decide how best to advance play, defend, or redistribute the ball without using their hands [2]. Among football's fundamental actions, passing (the act of deliberately moving the ball from one teammate to another) is one of the sport's most fundamental and nuanced actions. An accurate pass depends on a multiple of context-sensitive factors: the spatial distance between passer and receiver, the number and positioning of opponents, overarching team tactics and playstyle, the individual passing skill and technique of the player, and even environmental conditions such as pitch surface or weather [3], [4].

In Konami's simulation eFootball, AI contributes both opponent behavior and certain automated actions, striving to emulate the tactical and gameplay of real-world football as shown in figure 1. Among these, the AI's ability to select the most appropriate passing option in real time is critical to creating a challenging but also fair experience for human players. In this paper, we identify and model five primary factors that influence pass selection decisions, drawing from both real-world football dynamics and in-game mechanics observed in eFootball. The five factors are inter-player distance, local opponent density, configured team tactics/playstyle, player playing style, and AI playing styles. We will model the decision process as a directed-weighted graph. We then analyze the computational complexity of common graph-based selection algorithms, with the aim of accomplish feasible real-time implementation in eFootball's match engine.

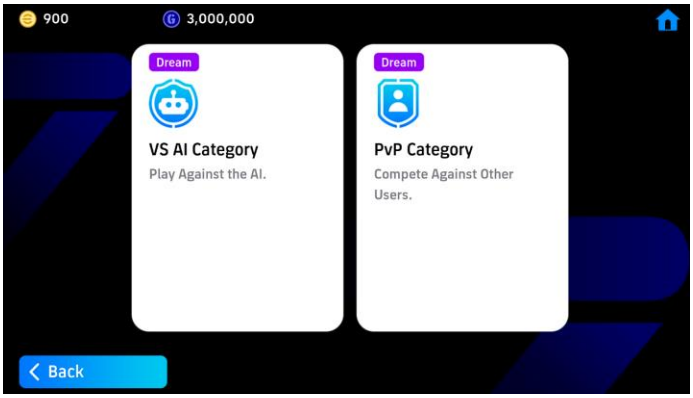

Fig.1 Play against AI feature in eFootball

II. THEORITICAL FRAMEWORK

The contents of this theoretical framework is primarily based on R. Munir's Lecture Notes [5].

A. Graph

A graph represents a fundamental discrete mathematical structure composed of vertices interconnected by edges, formally defined as an ordered pair G = (V, E). The vertex set V = $\{v_1, v_2, ..., v_n\}$ constitutes a non-empty collection of distinct nodes that serve as the basic units of the structure, while the edge set E contains pairs of vertices representing their mutual connections, which may potentially be empty. This versatile framework models relationships between discrete objects across diverse domains.

Classification by Edge Presence

1. Simple Graph

These are fundamental graph structures characterized by the absence of both multiple edges between vertices and self-loops. Each pair of vertices is connected by at most one edge, and no vertex connects to itself as shown in figure 2.

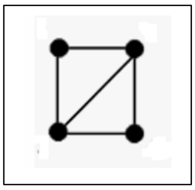

Fig. 2. Simple Graph.

2. Unsimple Graph

These graphs contain either multiple edges or self-loops, and can be further subdivided into:

- Multi-graphs: Graph structures that permit multiple edges between the same pair of vertices, allowing for more complex relationships between nodes as shown in figure 3.

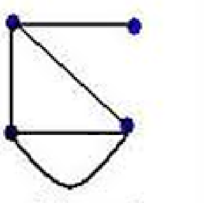

Fig. 3. Multi-graphs.

- Pseudo-graphs: Graphs that include self-loops, where vertices can have edges connecting back to themselves as shown in figure 4.

Fig. 4. Pseudo-graphs.

Classification by Directional Properties

1. Undirected Graphs

In these graph structures, edges represent bidirectional relationships with no inherent direction, meaning the connection between any two vertices can be traversed in either direction as shown in figure 5.

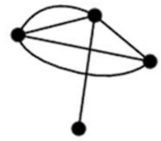

Fig. 5. Undirected Graphs.

2. Directed Graphs (Digraphs)

These graphs feature edges with specific directional orientation, creating asymmetric relationships where movement between vertices follows designated paths as shown in figure 6.

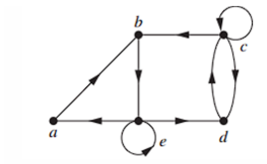

Fig. 6. Directed graphs.

Essential Graph Terminology

1. Adjacency

Two vertices are considered adjacent when they share a direct edge connection within the graph structure.

2. Incidence

An edge is said to be incident to vertices when it directly connects those vertices. In directed graphs, the edge's orientation determines the specific nature of this incidence relationship.

3. Isolated Vertices

These are vertices that lack any incident edges, existing independently within the graph structure without connections to other vertices.

4. Null Graphs

A graph configuration containing vertices but no edges, representing a collection of isolated points.

5. Vertex Degree

The total count of edges incident to a particular vertex. In directed graphs, this concept expands to include:

- In-degree: The number of edges terminating at a vertex
- Out-degree: The number of edges originating from a vertex

6. Paths

A sequence of connected vertices where each consecutive pair is linked by an edge, with no edge repetition allowed. Vertices may appear multiple times, but edges cannot be reused.

7. Cycles and Circuits

Closed paths that begin and end at the same vertex without repeating edges. Undirected graphs use the term "cycle," while directed graphs employ "circuit."

8. Graph Connectivity

A graph achieves connectivity when every pair of vertices can be linked through some path. Conversely, a graph is disconnected if at least one vertex remains unreachable from others.

9. Subgraphs and Complement Subgraphs

Graph structures formed by selecting subsets of vertices and edges from an original graph, maintaining the original connectivity relationships. Complement subgraphs containing all vertices from the original graph but featuring edges that were absent in a specified subgraph.

10. Spanning Subgraphs

Subgraph structures that include every vertex from the original graph while potentially containing only a portion of the original edges.

11. Cut-Sets

Sets of edges whose removal results in graph disconnection, representing critical connectivity elements.

12. Weighted Graphs

Graph structures where edges carry numerical values representing various parameters such as distance, cost, or other quantitative measures relevant to the specific application context as shown in figure 7.

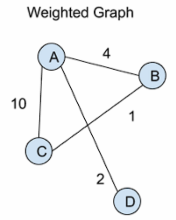

Fig. 7. Weighted graphs.

In this theoretical framework, directed-weighted graphs are utilized to model pass selection decisions. Vertices represent the players and edges will represent the pass path with each of the five factors (inter-player distance, local opponent density, configured team tactics/playstyle, player playing style, and AI playing styles) receiving weights that reflect their relative importance in the decision-making process.

B. Algorithm Complexity

In designing intelligent systems, it is not sufficient for an algorithm to simply be correct. It must also be efficient. The measure of this efficiency is what we refer to as algorithm complexity, which quantifies the computational resources (primarily time and memory) that an algorithm requires as a function of input size.

1. Time Complexity

Time complexity, denoted $T(n)$, refers to the number of computational steps an algorithm performs relative to the size of its input. In this paper, input size may represent the number of players considered during pass selection, or the number of potential paths in a passing graph. Different scenarios lead to different time complexities:

- Best Case $T_{\min}(n)$: Minimum time required (e.g., optimal pass found immediately).
- Worst Case $T_{\max}(n)$: Maximum time required (e.g., pass must be evaluated for all teammates).
- Average Case $T_{\text{avg}}(n)$: Expected time over all possible cases.

These complexities are evaluated using dominant operations for example, in pathfinding, the number of comparisons or edge traversals.

2. Asymptotic Notation

Since exact operation counts vary depending on platform or compiler, we adopt asymptotic notation to generalize how complexity behaves as n becomes large. The most common is Big-O notation, which provides an upper bound:

$$T(n) = O(f(n)) \text{ if } \exists\, C>0,\ n_0>0 \text{ such that } T(n) \leq C \cdot f(n)\ \forall\ n \geq n_0. \quad (1)$$

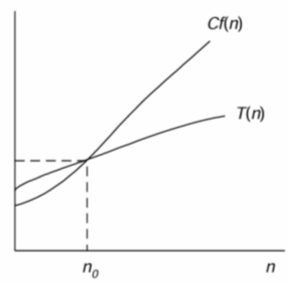

Fig. 8. Graphical illustration of the Big-O notation.

3. Time Complexity in Pass Selection Algorithms
In the context of pass selection modeled as a graph search problem, we often evaluate the complexity of known algorithms:
- Breadth-First Search (BFS):
$O\ (V{+}E)$ for unweighted graphs. Traverses level by level and finds the shortest path in unweighted graphs. Not ideal for pass selection, as passes differ in every condition.

- Dijkstra's Algorithm:
$O\ (V\ \log\ V\ +\ E)$ for graphs with non-negative edge weights. Finds the minimum total weight path from a source to all other nodes. Suitable for modeling passes with risk or distance.

- A Search:*
Enhances Dijkstra with a heuristic (e.g., proximity to goal), making it more efficient when a destination is known.

Where:
V: number of vertices (players)
E: number of edges (pass options)

Given that eFootball simulations must operate in real-time, choosing an algorithm with lower time complexity is critical to ensure smooth gameplay.

III. METHODS

This study uses a graph-based approach to model pass selection in eFootball simulations. Each potential pass is represented as a weighted edge in a directed graph, with weights determined by five factors: distance, opponent pressure, team tactics, player styles, and AI behavior. The system is implemented in TypeScript with React for visualization and user interaction. Complexity analysis evaluates the computational efficiency of the pass selection algorithm.

A. Key Factors for pass selection

The pass selection model incorporates five key factors that influence decision-making in football simulations. The contents of this section are primarily based on three key sources [6], [7], and [8]. These factors are categorized into dynamic and static components:

Dynamic Factors (updated during match progression):
- Inter-player distance
- Local opponent density

Static Factors (predefined game parameters):
- Team tactics/playstyle
- Individual player playing styles
- AI behavioral styles

1. Team tactics/playstyle

For team tactics/playstyle will be divided into two sections are team playstyle and team formation:
- **Team Formation**
Different formations create distinct passing patterns and positional relationships:

a. 4-2-2 Formation
A balanced tactical setup providing equilibrium between attacking and defensive phases, offering moderate passing options in all areas of the pitch.
b. 4-3-3 Formation and the variation
An attack-oriented system emphasizing wide play through wingers, creating passing networks that prioritize flanks and crossing opportunities.
c. 3-4-3 Formation and the variation
A formation emphasizing both width and central control, providing multiple passing lanes through wing-backs and central midfielders.
d. 5-3-2 Formation and the variation
A defensively stable system that creates passing networks focused on security and counter-attacking opportunities.

Each formation is associated with a distinct positional bias that influences preferred passing lanes.

- **Team Playstyle**
In eFootball, there are 5 team playstyle that can be used:
a. Possession Game
Emphasizes ball retention through short, accurate passes and patient build-up play. Passing weights favor safe, maintaining-possession options with high success probability.

b. Quick Counter
Prioritizes rapid transitions from defense to attack, weighting direct forward passes and exploiting space behind opponent lines.

c. Long Ball Counter
Utilizes immediate long passes after possession recovery, bypassing midfield areas and targeting advanced positions.

d. Out Wide
Maximizes pitch width utilization, heavily weighting passes to wide areas and cross-delivery positions.

e. Long Ball

Direct style emphasizing immediate upfield progression through long passes.

2. Player playing style
Player-specific behavioral patterns influence passing decisions through role-based positioning and movement:
Attacking Roles

- Goal Poacher (CF): Positions near defensive lines, receives weighted passes for finishing opportunities
- Dummy Runner (CF/SS/AMF): Creates space through movement, influences passing through positional decoys
- Target Man (CF): Physical presence for aerial balls and hold-up play
- Deep Lying Forward (CF/SS): Drops deep to receive passes and initiate attacking phases

Creative Roles

- Creative Playmaker (SS/RWF/LWF/AMF/RMF/LMF): High passing ability to create chances
- Classic No. 10 (SS/AMF): Central attacking focal point for creative passing
- Hole Player (SS/AMF/RMF/LMF/CMF): Makes runs into space, receives weighted through-balls

Wide Roles

- Prolific Winger (RWF/LWF): Receives passes in wide areas with cutting-inside options
- Roaming Flank (RWF/LWF/RMF/LMF): Moves between wide and central areas
- Cross Specialist (RWF/LWF/RMF/LMF): Positioned for crossing opportunities

Midfield Roles

- Box-To-Box (RMF/LMF/CMF/DMF): Receives passes throughout midfield areas
- Anchor Man (DMF): Deep positioning for defensive passing options
- The Destroyer (CMF/DMF/CB): Defensive-oriented passing reception
- Orchestrator (CMF/DMF): Deep playmaking with forward passing emphasis

Defensive Roles
- Build Up (CB): Initiates attacks from defensive positions
- Extra Frontman (CB): Joins attacks, receives passes in advanced positions
- Attacking Full Back (RB/LB): High attacking area passing weights
- Defensive Full Back (RB/LB): Conservative passing reception patterns
- Full Back Finisher (RB/LB): Central attacking area positioning

Goalkeeper Roles
- Attacking Goalkeeper (GK): Extended area coverage for passing options
- Defensive Goalkeeper (GK): Goal-line focused positioning

3. AI playing styles
This defines how the AI controls the player currently in possession. These tendencies impact which passes options are even considered in the graph, based on risk preference and tactical intention.

- Trickster: Favors dribbling with skills; avoids early passing.
- Mazing Run: Penetrates by dribbling through defenders.
- Speeding Bullet: Prioritizes forward progression via speed.
- Incisive Run: Cuts inside from wings to look for shots.
- Long Ball Expert: Prefers sending long passes over the top.
- Early Crosser: Seeks to deliver early crosses before closing defenders.
- Long Ranger: Frequently takes long-range shots.

Each style modifies the pass evaluation formula, e.g., lowering the weight for long-range passes or increasing preference for dribble-first actions.

B. Calculation

Each of the five factors contributes to the computation of edge weights in the pass selection graph. For each teammate node, a total score is computed based on normalized and weighted contributions from the five factors. The directed edge with the lowest weighting represents the optimal pass based on current conditions.

A weighting function is defined as:

$$\text{Score}_{pass}(i,j) = w_1 \cdot D_{ij} + w_2 \cdot O_j + w_3 \cdot (100 - T_{ij}) + w_4 \cdot (100 - P_i) + w_5 \cdot (100 - A_i) \quad (2)$$

Where:
- D_{ij}: Normalized inter-player distance between player i and teammate j.
- O_j: Opponent density around j.
- T_{ij}: Tactical alignment between current team strategy and pass direction.
- P_i: Style match between teammate i's role and the current game context.
- A_i: Ball-holder's AI style preference.
- w_1, w_2, w_3, w_4, w_5: Weights for each factor equal (0.2)

To compute the edge weights in the pass selection graph, five factors are evaluated and combined using a weighted scoring function. Each variable is numerically enumerated based on its contextual significance within the football simulation. The first factor, inter-player distance (D_{ij}), is calculated using the standard Euclidean distance formula, representing the spatial separation between the ball-holder and a potential teammate, measured in meters.

The second factor, opponent density (O_j) quantifies the number of opposing players located within a 15-meter radius around the target teammate. Each nearby opponent contributes a fixed penalty of 20 points, resulting in discrete density scores such as 0, 20, 40, and so on.

The third factor, tactical alignment (T_{ij}), reflects the compatibility of a potential pass with the team's tactical approach. Because tactical strategies are qualitative, scoring rules are predefined for each style. For example, under the *Possession Game* strategy, short passes (less than 30 meters) are favored and assigned 50 points, while longer passes are given only 30 points. The *Quick Counter* strategy values forward momentum; thus, passes directed ahead receive a double multiplier, with longer passes above 40 meters earning 30 points and shorter ones only 10. The *Long Ball Counter* strategy awards 50 points to long passes over 50 meters and 10 otherwise. The *Out Wide* tactic prioritizes passes to wing players (WF or wide MF), assigning them 50 points and others 20. The *Long Ball* tactic gives 60 points to passes longer than 60 meters and 10 otherwise. Additionally, all tactics provide a bonus of 20 points if the pass is directed to a player positioned ahead of the ball-holder on the horizontal axis (forward progress).

The fourth factor, style match (P_i), begins with a base score of 70 and accounts for the positional role and playstyle of the target teammate. An additional 15 points are added if the player possesses a favorable passing-oriented style such as *Creative Playmaker*, *Classic No. 10*, *Cross Specialist*, *Orchestrator*, or *Build Up*. Moreover, if the ball-holder's AI playstyle is categorized as *Long Ball Expert*, *Early Crosser*, or *Long Ranger*, a further 10-point bonus is applied.

The fifth factor, AI preference (A_i), reflects how the ball-holder’s AI behavior influences pass decisions. For instance, if the ball-holder is a *Long Ball Expert* and the pass exceeds 40 meters, 80 points are assigned. Similarly, *Early Crosser* behavior results in 80 points if the ball-holder plays as a winger, and *Mazing Run* favors short-range passes under 20 meters with the same bonus. *Long Ranger* AI also receives 80 points for passes over 50 meters. In all other cases, the AI preference is calculated as a base value of 50 plus a random value in the range of 0 to 30.

The optimal pass selection corresponds to the edge with the lowest cumulative score, calculated by aggregating all five factors according to their individual characteristics and current game state relevance. To ensure that higher values represent better suitability while maintaining lower-is-better total score, the final scoring function inverts the positively scored components for tactical alignment, style match, and AI preference by subtracting them from 100. Consequently, the optimal pass is the one associated with the lowest cumulative score across all five weighted factors.

C. The Program

This simulation was implemented using the TypeScript programming language, with the React.js framework for user interface interactivity. Visualization of the football pitch and player movement is handled using native HTML canvas and state management libraries, while the logic behind pass selection relies on modular function-based utilities. Core algorithms are implemented within dedicated modules to support extensibility and dynamic evaluation. The simulation includes the use of libraries such as React, TailwindCSS, and TypeScript interfaces for managing domain-specific data such as players, AI styles, positions, and pass scoring.

The program's input is defined through user interaction on the interface. Prior to the simulation, the user selects two main parameters are team formation and team tactic. These choices determine the configuration of the player's team in terms of positional layout and passing strategy.

Once the simulation is initiated by pressing the Start button, the system randomly selects a ball-holding player from the team. The opponent players dynamically reposition themselves toward the ball holder to simulate real-time pressure. Based on the current tactical setup, player locations, and opponent proximity, the program evaluates all possible pass options by constructing a directed, weighted graph, where each edge from the ball-holder to a teammate represents a feasible pass. Each edge is weighted using a formula based on five factors: distance, opponent density, tactical alignment, style match, and AI preference. The computation follows the equation described in the previous subsection, and the score is shown beside each pass line on the field.

When the user clicks on any of the edges (pass options), a Pass Calculation is triggered on the right side of the interface. This panel displays the detailed breakdown of each factor, their respective weights, and the computed total score. The lowest total score corresponds to the most optimal pass under current conditions. The user may then click the Step (Pass Ball) button to advance the game state, which passes the ball to the optimal teammate and updates the field. The simulation terminates after do the optimal pass based on the calculation.

In terms of algorithmic complexity, the pass selection logic is designed within a graph-based decision framework. For each decision frame, a directed graph is constructed with the ball-holder as the source node and all teammates as target nodes. The system computes edge weights based on five contextual football factors, and selects the pass with the lowest cumulative score. As outlined in the theoretical framework, this approach represents a practical application of graph-based evaluation and scoring rather than classical pathfinding algorithms.

IV. Result and Implementation

This program demonstrates how pass selection operates using a graph-based approach in eFootball simulations. The implementation serves as a simplified prototype that illustrates the core logic and decision-making process behind intelligent pass selection systems. The simulation allows users to select a formation and team tactic, after which the system generates players and opponents dynamically, evaluates all possible passes using a scoring function, and visualizes the decision-making process on a football field. This provides readers with a clear and interactive representation of the methodology proposed in the title of this study.

A. System Interface and Setup

The initial interface of the program is presented through a clean layout where users can select their preferred team formation and team tactic before starting the simulation as shown in figure 9 and figure 10. These two dropdown menus allow combinations such as "4-3-3" with "Quick Counter" or "3-4-3" with "Possession Game" and many more combinations. These enabling users to experiment with different tactical setups and observe how they influence pass selection decisions. Once the desired inputs are chosen, users click the Start button to begin the simulation. The system then automatically generates a set of players based on the selected formation and assigns them realistic positions on a football field layout. At the same time, opponents are placed strategically on the other half of the field. This setup reflects the selected strategy, offering a clear and interactive visualization of how pass decisions could dynamically change under different tactical conditions.

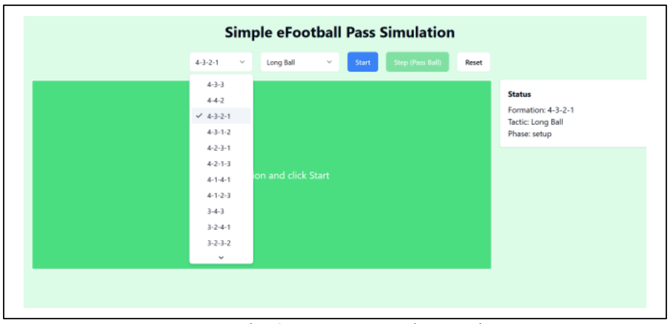

Fig. 9. Team Formation Option

Fig. 10. Team Tactics Option

B. Simulation and Output

After the user clicks the Start button, the simulation progresses to the next phase where the system selects a random player to act as the ball-holder. At this stage, the interface transitions to display the football field overlaid with a passing graph. Each teammate on the field is represented as a node, and all feasible pass options from the ball-holder are visualized as directed edges pointing toward those nodes as shown in figure 11. These edges are forming a weighted directed graph, where the ball-holder acts as the source node, and each directed edge carries a weight representing the pass score calculated using the five evaluation factors. The score is shown on or near each edge to reflect the pass quality. This graph gives users an immediate visual understanding of all available pass options and how each is evaluated in context, based on position, tactic, pressure, and player style.

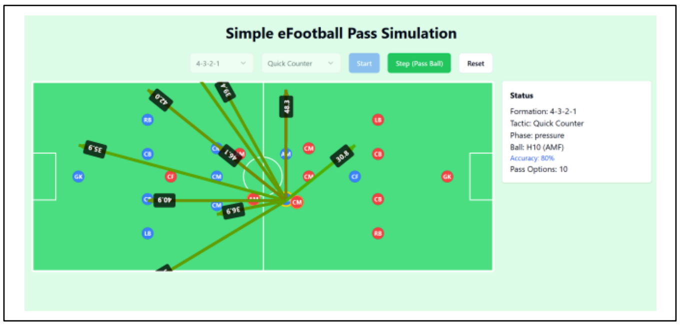

Fig. 11. Directed-Weighted Graph as the passes option

C. Score Evaluation and Calculation

Users can examine the calculation behind each pass option by clicking on any of the directed edges in the graph. Once selected, a detailed calculation panel appears on the right side of the interface, displaying the five contributing factors: distance, opponent density, tactical alignment, style match, and AI preference. Each factor is shown along with its raw value, normalized adjustment, and contribution to the total pass score. This allows users to transparently observe how each pass is evaluated and why certain passes are prioritized over others. As shown in the figure, this feature provides insight into the decision-making process of the system and demonstrates how the weighted scoring function operates in real time.

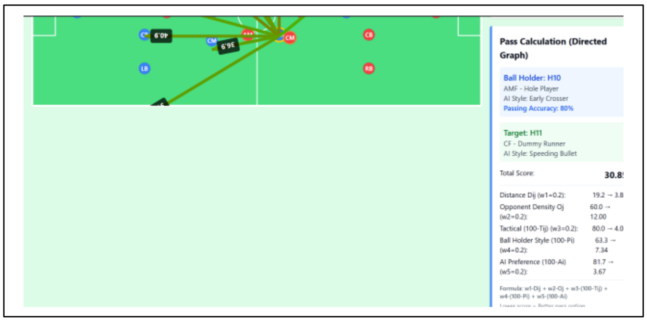

Fig. 12. Calculation Panel after clicking an edge

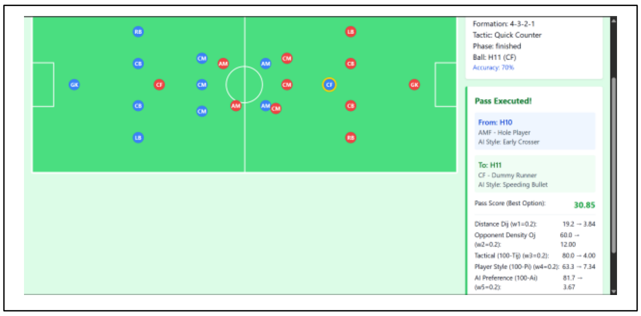

Fig. 13. Calculation Panel after the pass executed

D. *Computational Efficiency*

The main computational cost of the simulation arises in the calculatePassOptions function, where the system evaluates multiple scoring factors for each teammate. Among these, the opponent density factor introduces the most significant performance impact, as it requires scanning all opponent positions. This behavior is evident in the following operation:

As shown in figure 14, the algorithm iterates over all teammates (n) and, for each one, calculates a total pass score based on five contextual factors: distance, opponent density, tactical alignment, style match, and AI preference. As shown in figure 15, the calculateOpponentDensity function performs a full scan of all opponents (m) using a distance-based filter to count nearby defenders, resulting in an overall time complexity of O(n × m) per simulation frame. While the other four factors are computed in constant time, the opponent density check forms the dominant term in the computational cost. Nonetheless, since football simulations operate under bounded player sets (typically $n \leq 10$ and $m \leq 11$), the algorithm remains computationally efficient and responsive in real time.

Fig. 13. Iterates over all teammates (n)

Fig. 14. Iterates over all opponents (m)

Building upon the algorithm complexity principles outlined in the theoretical framework, this implementation demonstrates a practical application of graph-based optimization that balances decision accuracy with performance constraints under realistic match conditions.

V. Conclusion

This study presents a graph-theoretic approach to pass selection in football simulations, inspired by tactical dynamics from real-world football and the gameplay structure of eFootball. By modeling the decision-making process as a directed weighted graph and evaluating passes based on five key contextual factors (distance, opponent density, team tactics, player style, and AI behavior) the system demonstrates how intelligent and adaptive passing decisions can be computed and visualized in real time.

Through simulation, we show that this approach allows dynamic interaction and transparency in decision logic, offering users insight into how football AI systems might evaluate pass quality in various match contexts. Additionally, an analysis of computational complexity reveals that the proposed algorithm maintains a manageable runtime of O(n × m), enabling responsive behavior even with real-time constraints.

These findings show the feasibility of combining discrete mathematics, particularly graph theory and complexity analysis, with soccer-specific domain knowledge to create intelligent, interpretable, and efficient decision systems. This model can serve as a basis for further to explore algorithm design and evaluation in applied contexts.

Future research directions could explore machine learning approaches to dynamically adjust the weighting factors (w_1 through w_5) based on real-time match context, opponent behavior patterns, and individual player performance metrics. Adaptive weighting systems could learn from successful and unsuccessful pass outcomes to continuously improve decision quality. Additionally, extending the model to consider sequential pass combinations and multi-step tactical planning, rather than individual pass decisions, could provide more sophisticated strategic planning capabilities that better reflect the complex chain-reaction nature of football tactics.

VIDEO LINK AT YOUTUBE AND GITHUB

Include link of your video on YouTube in this section:

https://youtu.be/NMBw84xooNA

Include link of your code on GitHub in this section:

https://github.com/hakamavicena/efootball-pass-optimizer.git

ACKNOWLEDGMENT

First of all, I would like to express my deepest gratitude to God, who has always been present and guided me through every condition. Secondly, I extend my heartfelt thanks to my family for their amazing support in all of my endeavors. Last but not least, I sincerely appreciate every aspect of the Discrete Mathematics Class IF1220. This includes our beloved lecturers, Mr. Rinaldi Munir and Mr. Arrival Dwi, the teaching assistants, and all of my fellow classmates who have made this class so insightful and engaging. I am truly happy and blessed to have been a part of this class, and I hope that all the knowledge I have gained both technical and life lessons will be useful in my future journey.

REFERENCES

[1] IBM Corporation, “What is Artificial Intelligence (AI)?”, [Online]. Available: https://www.ibm.com/think/topics/artificial-intelligence. [Accessed: 14-Jun-2025].

[2] Encyclopædia Britannica, “Football (Soccer)”, [Online]. Available: https://www.britannica.com/sports/football-soccer. [Accessed: 14-Jun-2025].

[3] Sportmonks, “Passing Accuracy – Football Glossary”, [Online]. Available: https://www.sportmonks.com/glossary/passing-accuracy/. [Accessed: 14-Jun-2025].

[4] A. L. Chmura, S. Barrett, and B. Born, “Match running performance and passing accuracy in elite football: Insights from spatial and physiological data,” Front. Physiol., vol. 15, 2024. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC11053140/#Sec3. [Accessed: 14-Jun-2025].

[5] R. Munir, “Materi Kuliah IF2120 Matematika Diskrit,” Institut Teknologi Bandung, [Online]. Available: https://informatika.stei.itb.ac.id/~rinaldi.munir/. [Accessed: 14-Jun-2025].

[6] TeamStats.net, “Football Formations: A Comprehensive Guide to the Most Popular Tactics in the Game,” [Online]. Available: https://www.teamstats.net/blog/football-formations-a-comprehensive-guide-to-the-most-popular-tactics-in-the-game. [Accessed: 17-Jun-2025].

[7] The Sports Reviewer, “Football Tactics: Possession vs Counter Attacking,” [Online]. Available: https://thesportsreviewer.com/football-tactics-possession-vs-counter-attacking/. [Accessed: 17-Jun-2025].

[8] Total Football Analysis, “Tactical Theory: A Comprehensive Guide to Direct Possession – Tactical Analysis & Tactics,” [Online]. Available: https://totalfootballanalysis.com/article/tactical-theory-a-comprehensive-guide-to-direct-possession-tactical-analysis-tactics. [Accessed: 17-Jun-2025].

[9] R. A. Syarif, “Analisis Strategi Optimal pada Permainan eFootball Menggunakan Relasi Fungsi dan Struktur Graf,” Institut Teknologi Bandung, 2024. [Online]. Avaiable: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/Makalah/Makalah-IF1220-Matdis-2024%20(15).pdf. [Accessed: 20-Jun-2025].

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 20 Juni 2025

Hakam Avicena Mustain - 13524075