ISSN: 2690- 912X

Research Article

Journal of Genetic Engineering and Biotechnology Research

Agent-Based Management and Coordination of Aircraft at Intersections

Elhoucine Ouassam[1]*, Yassine Dabachine[1], Nabil Hmina[1] and Belaid Bouikhalene[1]

[1]Laboratory of Mathematics Innovation and Information Technology (LIMATI), Mathematics and computers Sciences, Sultan Moulay Slimane University, Morocco

[*]Corresponding Author
Elhoucine Ouassam, Laboratory of Mathematics Innovation and Information Technology (LIMATI), Mathematics and computers Sciences, Sultan Moulay Slimane University, Morocco

Submitted: 2023, Mar 02; **Accepted:** 2023, Sep 20; **Published:** 2023, Nov 23

Citation: Ouassam, E., Dabachine, Y., Hmina, N., Bouikhalene, B. (2023). Agent-Based Management and Coordination of Aircraft at Intersections. *J Gene Engg Bio Res, 5*(3), 198-205.

Abstract
This work presents a game-theoretic approach to address the problem of coordination and scheduling of aircraft at intersections, with the goal of avoiding conflicts and potential collisions. The proposed algorithm enables simple agents to work together in a way that leads to cooperative behaviors, resulting in equilibria that improve the overall efficiency of the system. The researchers tested and compared the game-theoretic approach with a centralized approach, specifically FirstCome-First-Serve (FCFS), using data from Mohammed 5 Casablanca airport. The initial results suggest that the game-theoretic model is promising, despite its higher complexity. The approach has the potential to improve the overall coordination and scheduling of aircraft, leading to a more efficient and safe system. The proposed game-theoretic approach is designed to improve the coordination and scheduling of aircraft at intersections, ultimately leading to a safer and more efficient system. The approach is shown to be promising in initial testing, offering a potentially superior alternative to centralized approaches like FCFS. This research highlights the potential benefits of game-theoretic models in addressing complex coordination problems in multi-agent systems.

Keywords: Agent-Based Modeling, Cooperative Agent, Distributed Computing, Game Formalism, Scheduling of Aircraft

1. Introduction

Conflict management and coordination are probably the most active fields of research in Distributed Artificial Intelligence and more particularly in Multi-Agent Systems as mentioned earlier in [1]. Many types of coordination mechanisms have been designed and developed and; aber of them use game theory as in [2, 3]. Games can be considered the simplest way to model conflict situations as shown by H. A. Simon and G. Y. Ke et al. [4, 5]. The initial use of mathematical game theory in the design of a multi-agent coordination mechanism goes back to Rosenschein in [6]. A multi-agent coordination mechanism is a system in which multiple autonomous agents work together to achieve a common goal. In such systems, each agent is responsible for a specific task and communicates with other agents to ensure that the overall goal is achieved. The coordination mechanism enables agents to share information, synchronize their actions, and resolve conflicts that may arise during the course of their activities. One of the most important challenges in multi-agent systems is to develop effective coordination mechanisms that can manage conflicts and ensure that agents act in a collaborative and efficient manner. Game theory is one approach that has been widely used to design such mechanisms. In game theory, agents are modelled as players who compete or cooperate with each other to achieve their objectives. By analyzing the strategies that agents can use to achieve their goals, game theory provides a way to design coordination mechanisms that can manage conflicts and ensure that agents act in a coordinated and effective manner. Coordination mechanisms based on game theory have been successfully used in various applications such as transportation, logistics, and robotics. For example, in the field of transportation, game theory has been used to develop intelligent traffic management systems that can optimize the use of road networks and reduce congestion. In logistics, game theory has been used to design efficient supply chain management systems that can optimize the allocation of resources and minimize costs. Thereafter, the ever-increasing research in Artificial Intelligence has allowed the development and implementation of many industrial and commercial applications that take advantage of the link between agents and game theory in [7]. In the field of airport traffic simulation, relatively few publications have appeared and only a few are related to the work presented here. This shows that, even if the work done to formalize and generalize multi-agent coordination methods is important, the mechanisms are often limited to specific applications. This work first presents the problem of coordination of simulated airport traffic and more particularly the case of conflicts at intersections. Work on a distributed coordination mechanism based on games and the notion of property is proposed in a second step. The implementation of this mechanism and the simulation results are presented to conclude.

2. Proposal of a Game-Theoretic Mechanism Based on the Notion of Priority

2.1 Behavioral Simulation: Background

A conflict situation at an intersection can be considered a game. A game in is represented by a situation in which individuals (the players) must choose among several possible actions (strategies) in a predefined format (the rules of the game) [8]. These choices give an outcome to the game (the solution), which is associated with a positive or negative payoff for each participant. In the context of the behavioral simulation of airplanes passing through intersections, the players are the planes approaching or entering the intersection. The possible actions of the players can be to accelerate or brake (this is of course a subjective limitation). The main characteristic of a coordination mechanism in the present context is to constrain acceleration. The design of the coordination mechanism is then to define the rules and the method of resolution. In the case of conflict management at airport intersections, as mentioned in the rules are those that respect the landing and take-off times of the aircraft [9, 10]. However, this is far from always being respected, due to the numerous causes of delay. So what can be said about game theory in the context of airport traffic behavioral simulation? On the one hand, the assumptions made in game theory in are strong and hardly compatible with what we know about human behavior [11]. On the other hand, a large part of the work done in-game theory concerns the search for and study of equilibria to find a solution to games [12]. These two remarks are essentially due to an assumption that concerns the knowledge of the game that the players have. The game theorist as in often assumes that all players play the same game. This assumption is very difficult to validate, and may even be contrary to observable behavior in the context of airport traffic. Moreover, the possible multiplicity of equilibria can search searching computationally expensive. Moreover, since the situations of an intersection are highly dynamic, which implies a frequent re-evaluation of the situation, aircraft have little or no memory, and we could only consider one-turn games. Moreover, aircraft only perceive situations locally; in this case, the information is said to be incomplete (many works in psychology show that the level of resources devoted to the processing of interactions is limited and consequently that not all interactions can be processed). Similarly, the autonomy of the agents imposes a decision-making process that is independent of the others (at each cycle, the mobiles calculate their actions pseudo-parallel); in this case, the information is said to be imperfect. Given all these elements, two ideas emerge. First, we assume the definition and the choice of our resolution criterion and, consequently, the design of the matrix modeling the game. Secondly, everything related to the behavioral aspect, which is not taken into account by the game, is considered before the creation of the game. To do so, the notion of priority was used because of its primordial role for the pilot in his speed regulation strategy.

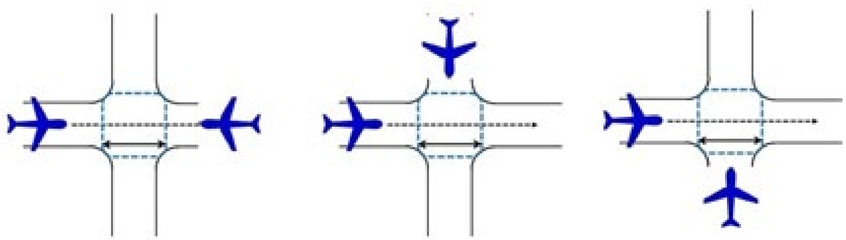

Figure 1: The Three Situations Involving Two Airplanes

2.2 Priority Relationship

The coordination mechanism then takes place in three stages. At each cycle, each pilot determines (or estimates) the priority relations he has with the other aircraft. Then he models, in the form of a game, the local situation represented by these relationships. Finally, he solves the game. This game is static (one turn only) and the dynamics of the system are expressed by the potential change of the priority relations (thus of the game) at each cycle. The mechanism is distributed and a given global situation can be, over time, interpreted differently by each aircraft. At this stage, we will assume that a pilot knows how to determine the priority relations he has with the other players. The principle of the al operation is based on a decision to be taken by the different agents at each cycle. Each player approaching an intersection must first determine with which other player he will play, then the game he will play. Finally, each player has to solve his game, i.e. choose the action that seems most interesting.

2.3 Modeling of Basic Two-Player Situations

Once the method of resolution is known, let us move on to the design of the matrices. To simplify we will only consider intersections in X as shown in Fig.1. Indeed intuitively, we can assume that a complex intersection is the sum of simple intersections. For situations involving two aircraft, we note that there are several types of situations. These situations modeled by games are matrices of size 2 × 2 whose cells are pairs of gains. Each cell of the matrix corresponds to an outcome of the game, which is a vector of wins. This matrix of payoffs is called the strategic form of the game as in [13]. As the possible longitudinal actions for an airplane are to accelerate or to brake (symbolized by the terms *Accelerate* and

Brake), a payoff matrix for an n-player game is an n-dimensional matrix of size 2n and whose payoff vectors are of size n. This implies 32 variables to be determined. An analysis not detailed in this paper was performed to consider only the necessary variables. An explanation of the variables introduced corresponding to the different situations/matrices is detailed in [14]. This leads to considering 9 variables for all four final matrices, with the following constraints: $\{x_1, y_1, y_2, d_1, d_2\}$, $y_3 > y_2$. We note: Prio(A,B) the priority relation such that A has priority over B and ¬Prio(A,B) the priority relation such that A does not have priority over B. The final matrices for two-player games are the following.

	Brake	Accelerate
Brake	(x_1,x_1)	(x_3,x_0)
Accelerate	$(0,x_3)$	(0,0)

Table 1: Payoff matrix for the two-player game ¬(*Prio*(*A,B*) ∧ *Prio*(*B,A*))

	Brake	Accelerate
Brake	$(-y_2,-y_1)$	$(y_6,0)$
Accelerate	$(0,y_3)$	(0,0)

Table 2: Payoff matrix for the two-player game ¬*Prio*(*A,B*) ∧ *Prio*(*B,A*)

	Brake	Accelerate
Brake	$(-y_1,-y_2)$	$(y_3,0)$
Accelerate	$(0,y_6)$	(0,0)

Table 3: Payoff matrix for the two-player game *Prio*(*A,B*) ∧¬*Prio*(*B,A*)

According to the perception of the intersection, we must consider the fact that the information is incomplete: each player does not take into account the gains of the others. Thus, a player chooses the action that maximizes his payoffs: he sums up the payoffs for each action. More precisely, agent A selects the corresponding decision *matrixm*$_{A/B}$ (the same is true for player B). Player A selects the strategy S_A such that:

S_A = *a* ⋆ *Accelerate,Brake* g A(B) = m –A/B″(a*,Accelerate) + m –A/B″(a*,Brake)″(1) Similarly, player B selects the strategy S_B by: $S_A = \{a\star \in \{\text{Accelerate,Brake}\} \mid g_A(B) = m_{A/B}(a\star,\text{Accelerate})+mA/B(a\star,\text{Brake})\}$

(2)

Two-player situations are now modeled and conflicts are handled in a”realistic” way. It is now possible to consider multi-player situations.

3. Generalization to Several Players

When a situation involves three or more players, the game matrix modeling is also based on prior iterate. For a situation with three players, 6 binary relations exist ($B \to C$, $C \to B$, $A \to B$, $B \to A$, $A \to C$, $C \to A$). Each three-player game, therefore, corresponds to three two-player games ($A - B$, $A - C$, and $B - C$), and there are 64 (2^6) possible three-player games.

	Brake	Accelerate
Brake	$(-z_3 + d_1,-z_1 + d_2)$	$(z_1,0)$
Accelerate	$(0,z_1)$	(0,−0)

Table 4: Payoff matrix for the two-player game *Prio*(*A,B*) ∧ *Prio*(*B,A*)

The two-player matrices can be aggregated into a single three-dimensional matrix whose cells are payoff vectors of size 3 as in [15].

Let player A be from a set of three players *A, B, C*. Player A has two priority relationships with player B and also two relationships with player C. Two relations lead to a two-player game among a set of four possible games. More generally, the aggregation method chosen is the sum of wins, and the formula for the n-player game is shown below. Assuming a set $J = 1,2,...,k,...,n$ of players, the payoff Gk of player k for an outcome $S = (S_1,S_2,...,S_k,...,S_n)$ in an n-player game is described by the following formula:

$n \backslash k$

$$\forall k \in J,\ Sk = k\star \in \text{Accelerate,Brake} \mid \text{Gk} = \text{Xgk(i)} \quad (3)$$

i=1

Once the game is determined, each player must choose their strategy according to the previous formulation. The method is the same as for two players (maximization of the sum of the gains relative to each action). This allows us to take into account that a player does not perceive the situation as a whole but only what is relative in their local environment. A player only considers what comes from their interactions with the others and does not know a priori the nature of the existing interactions between two other players. For example, in a three-player game, each player considers only four of the six relationships: those in which they intervene (for player A: $A \rightarrow B$, $B \rightarrow A$, $A \rightarrow C$, $C \rightarrow A$). Of course, with such a lack of information, the solutions of the games cannot always be optimal and some cases (situations) may lead to accidents. It is then necessary to choose the values of the payoffs of the two-player matrices to reach certain optimality in the resolution of the three-player games. To this end, a semi-formal study based on the solution of systems of in equations was carried out using the Scilab1 software [16]. For the three-player games, the values retained are such that only 6 interlocking cases out of 64 possible cases appear as follows.

	Brake	**Accelerate**
Brake	(1,1)	(1,0)
Accelerate	(0,1)	(0,0)

Table 5: Payoff matrix for the two-player game ¬*Prio*(*A*,*B*) ∧¬*Prio*(*B*,*A*)

The same approach can be performed for n-player games. These theoretical results are being, a mechanic based on these games has been realized and applied to the simulation model.

	Brake	**Accelerate**
Brake	(−1,−6)	(2,0)
Accelerate	(0,1)	(0,0)

Table 6: Payoff matrix for the two-player game *Prio*(*A*,*B*) ∧¬*Prio*(*B*,*A*)

	Brake	**Accelerate**
Brake	(−6,−1)	(1,0)
Accelerate	(0,1)	(0,0)

Table 7: Payoff matrix for the two-player game ¬*Prio*(*A*,*B*) ∧ *Prio*(*B*,*A*)

	Brake	**Accelerate**
Brake	(0,−6)	(1,0)
Accelerate	(0,1)	(0,0)

Table 8: Payoff matrix for the two-player game ¬*Prio*(*A*,*B*) ∧ *Prio*(*B*,*A*)

4. Application to Behavioral Traffic Simulation (Implementation of the Mechanism in the Traffic Simulation Tool)

As mentioned in implementing a multi-agent coordination mechanismin a pseudo-parallel discrete-time simulation model involving a large number of agents, some of which may be human, is not trivial. First of all, because the agents must interpret their environment and act together at the same time, and because a software agent and a human agent do not communicate in the same way, this implies important constraints [17]. Let us consider the different steps of coordination, by using games, of the actions of an agent with those of the others at an intersection. Each agent first estimates whether there is a game or not, i.e. whether it determines itself as a player for a game related to its intersection. Then, the player agent searches for other agents "linked" to the same intersection. Next, the player determines whether he is an active player (i.e., whether he will actively take part in the game during the current simulation step). If he is an active player, the agent determines the priority relationships he has with other players and players who may participate in his game, which determines the game he will play. Finally, the agent solves me and acts on the solution found. Before the actual game, steps related to perception are necessary as in [18, 19]. For example, to know if another agent is a potential player for his game, an agent must estimate if he and the other are at the same intersection. To reduce the computation time, each agent determines the intersection on which it plays and makes the information available to its associates. But this information is only available to them at the next simulation step, which can introduce a lag in the games played by the different agents. Following this, each player determines if he takes an active part in the game during the current cycle of the simulation by estimating his situation vis-`a-vis the other players. Indeed, the fewer active players there are, the faster the calculation is; if a player is momentarily blocked by another, it is useless for him to play to calculate an acceleration that will be zero anyway. Next comes the determination of priority relations as discussed in which is an essential step because it is on this step that the proper functioning of the coordination mechanism largely depends since it gives rise to the game that will be played by the agent during the current cycle. This is the stage that requires the most computation since it is mainly based on the interpretation of the local environment: passing numerical information into symbolic information. This information is then processed to obtain the different priorities which are finally aggregated [20, 21]. Once the active players are known and the associated priority relations are also known, an active player determines the player's likelihood to participate in his game. At this point, we should specify that for speed and simplicity, and to be more in line with real traffic only three other mobiles at most are considered. When the players and the priority relations are established, the agent then has the two-player matrices. All that remains for the agent to do is to create its n-player game and solve it. The final matrix, which is, in theory, the aggregation of the two-player matrices, is not explicitly computed by the agent; the agent can, by computational trickery, directly obtain the solution. Indeed, knowing that the resolution method only takes into account the payments relative to the agent and that the payments relative to the action Brake are always zero Fig.3, the agent only makes the sum of the payments relative to the action Accelerate. If this sum is positive, he chooses the latter strategy, otherwise, he chooses the Brake strategy as in [22, 23].

5. Validation: First Results

In this chapter, we present the results obtained by applying the model obtained to our problem on data from the Mohammed 5 Casablanca airport. The simulator has been implemented in JAVA language and the experiments have been done on a machine, core i7 2.9 GHz and 16 GB of memory and using the GamBit tool [24]. Our validation successively focuses on the computation time-memory space requirements, and the speed of airplanes when approaching intersections.

6. Experimental Evaluation of Computing Time and Memory Requirements

Despite the simplicity of our example, we can see that the size of the state space is not negligible. The complexity is given by the formula $[2^{n2A} * 2^{v}]$, where A represents the number of aircraft and v is the number of channels. Figure 2 shows the evolution of s as a function of the size of the environment.

Therefore, an evaluation in terms of time/memory complexity Fig.3 shows that as the number of agents increases, so does the required memory space or the time in the worst case [25]. However, we consider that given the number

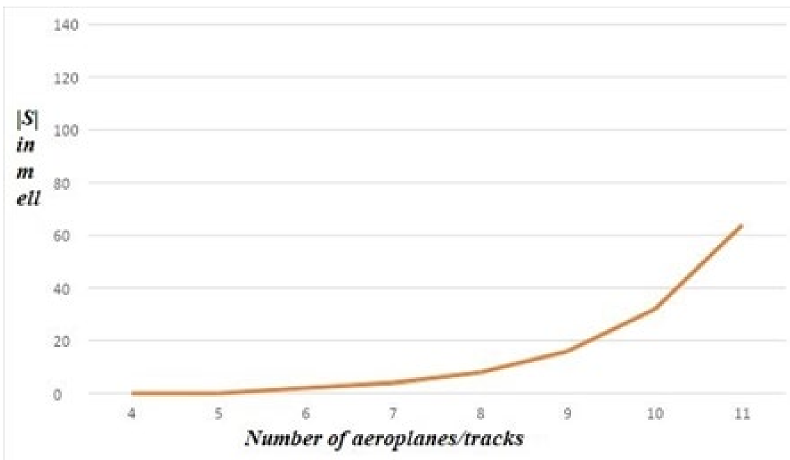

Figure 2: Evolution of the state space as a function of the number of airplanes/lanes.

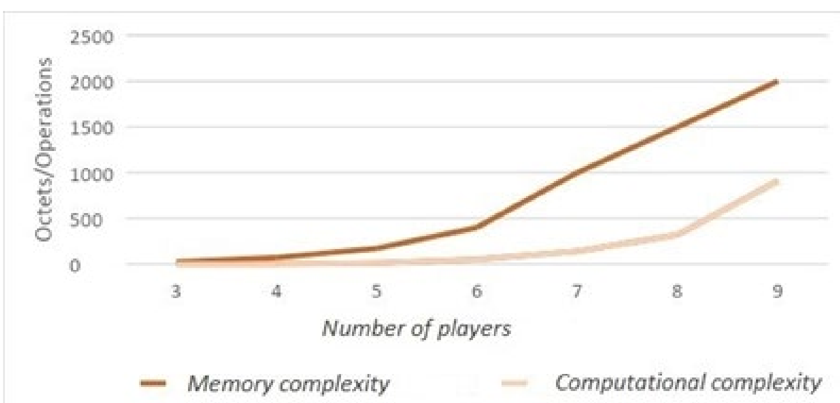

Figure 3: Computational Complexity of the Distributed Mechanism of agents (i.e. number of players playing the same game) directly in conflict, the mechanism gives satisfactory results.

To avoid any subjective evaluation, we compare the obtained solution with the one resulting from a centralized FCFS (first-come-first-served) control, considered here as an optimal bound for the relaxed problem. Thus, we notice that the construction of the matrices and the search for equilibrium increase considerably the computation time and the size of the required RAM, see Table 9. Specifically, we observe that the computation time measured for our algorithm is on average 1.5 times higher than that of the centralized approach.

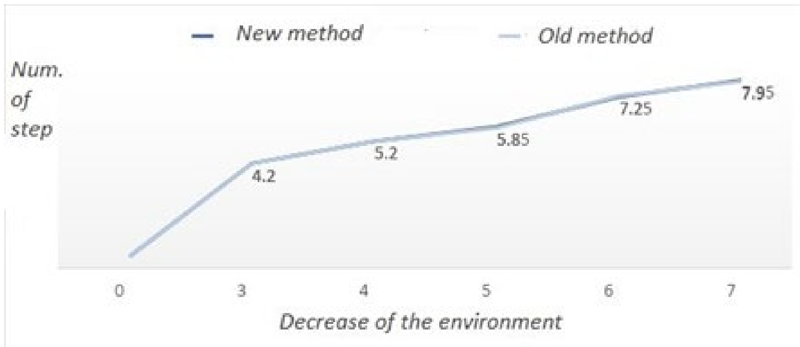

Figure 4: Comparison between the game theory and the FCFS method according to the number of steps needed to reach the objective.

Similarly, the space requirement is approximately 4 to 5 times higher than the reference approach.

ways	Memory	Time FCFS	Time TJ	Time FCFS TJ
2 Aeroplanes	2 ways	10.9	15.6	42
	3 ways	16	50.3	162
	4 ways	24	150	446
	5 ways	90	400	118
3 Aeroplanes	2 ways	9	30.2	94
	3 ways	23.7	100	390
	4 ways	60.3	285.6	100
	5 ways	172	720.5	250
4 Aeroplanes	4 ways	14.4	50.7	260
	5 ways	51.3	196.2	802
	6 ways	112.3	560.2	203

Table 9: Memory used (MB) and computation time (sec) for different numbers of objects, different sizes of the environment, and 3 agents.

7. Evaluation of the Agents' Joint Policies

As shown in fig.4 below, we have illustrated the change in the number of steps for both the old and the new methods, it is clear that the number of steps is about the same for both methods despite the change of the environment dimension.

8. Evaluation of Aircraft Speed on Approach to Intersections

The second validation concerns the speed of aircraft approaching an intersection [26]. For this, two aircraft arriving at an intersection must be considered, with one aircraft having priority over the other. The simulation results were compared to those of a centralized FCFS method Fig.5.

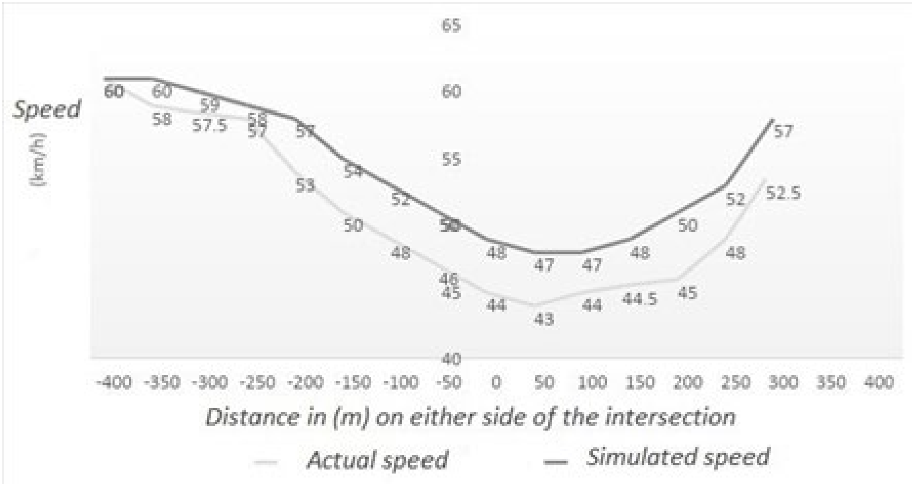

Figure 5: Average speeds at an X-intersection by a priority aircraft in the presence of another non-priority mobile.

9. Conclusion

This study aimed to investigate the potential of using cooperative games to coordinate a multi-agent system. The problem of coordination was identified as a challenging task, and we demonstrated the effectiveness of their approach through an illustrative example involving two airplanes at an intersection in an X-shaped pattern. The results of this model, coupled with the notion of Nash equilibrium, were found to be comparable, if not better, than a centralized approach. Looking forward, we plan to extend this research in two directions. In the short term, they aim to apply this model to a larger number of agents to test its scalability. In the medium and long term, they propose to evaluate the robustness of their approach in more complex intersections. Overall, this research contributes to the ongoing efforts to develop effective coordination mechanisms for multi-agent systems. By demonstrating the potential of cooperative games and Nash equilibrium in this context, this study offers insights into the design and implementation of distributed coordination mechanisms that are scalable and efficient. As such, it has implications for a wide range of real-world applications, including traffic management, logistics, and supply chain management.

Authors' Declaration

Conflicts of Interest: None. - We hereby confirm that all the Figures and Tables in the manuscript are mine ours. Besides, the Figures and images, which are not mine ours, have been given the permission for re-publication attached with the manuscript.

Authors' Contributions

Author name Elhoucine Ouassam role Conception, design, acquisition of data, analysis, interpretation, drafting the MS. Author name Yassine Dabachine role Conception, design, acquisition of data, analysis, interpretation, drafting the MS. Author name Nabil Hmina role revision and proofreading. Author name Belaid Bouikhalene role revision and proof reading.

References

1. Wu, J. J., Khan, H. A., Chien, S. H., & Wen, C. H. (2022). Effect of customization, core self-evaluation, and information richness on trust in online insurance service: Intelligent agent as a moderating variable. Asia Pacific Management Review, 27(1), 18-27.
2. Nguyen, T. H., & Jung, J. J. (2022). ACO-based traffic routing method with automated negotiation for connected vehicles. Complex & Intelligent Systems, 1-12.
3. Zulfiqar, M., Kamran, M., & Rasheed, M. B. (2022). A blockchain-enabled trust aware energy trading framework using games theory and multi-agent system in smat grid. Energy, 255, 124450.
4. Liu, C., Hua, C., & Chen, J. (2022). Efficient supervision strategy for illegal dumping of construction and demolition waste: A networked game theory decision-making model. Waste Management & Research, 40(6), 754-764.
5. Lopez, V. G., Lewis, F. L., Liu, M., Wan, Y., Nageshrao, S., & Filev, D. (2022). Game-theoretic lane-changing decision making and payoff learning for autonomous vehicles. IEEE Transactions on Vehicular Technology, 71(4), 3609-3620.
6. Choudhary, P. (2022). Polynomial time algorithms for tracking path problems. Algorithmica, 84(6), 1548-1570.
7. Wang K., Wang Y., Du H., Nam K. (2021). Game-Theory-Inspired Hierarchical Distributed Control Strategy for Cooperative Intersection Considering Priority Negotiation. IEEE Trans on Veh Tech. juill 70(7):6438-49.
8. Jiang, Y., Liu, Z., Hu, Z., & Zhang, H. (2021). A priority-based conflict resolution strategy for airport surface traffic considering suboptimal alternative paths. IEEE Access, 9, 606-617.
9. Yang, K., Yang, H., Zhang, J., & Kang, R. (2022). Effects on Taxiing Conflicts at Intersections by Pilots' Sensitive Speed Adjustment. Aerospace, 9(6), 288.
10. Zhu, M., Anwar, A. H., Wan, Z., Cho, J. H., Kamhoua, C. A., & Singh, M. P. (2021). A survey of defensive deception: Approaches using game theory and machine learning. IEEE Communications Surveys & Tutorials, 23(4), 2460-2493.
11. Dahiya, A., & Gupta, B. B. (2021). A reputation score policy and Bayesian game theory based incentivized mechanism for

DDoS attacks mitigation and cyber defense. Future Generation Computer Systems, 117, 193-204.
12. Agi, M. A., Faramarzi-Oghani, S., & Hazır, Ö. (2021). Game theory-based models in green supply chain management: a review of the literature. International Journal of Production Research, 59(15), 4736-4755.
13. Rokhforoz, P., Gjorgiev, B., Sansavini, G., & Fink, O. (2021). Multi-agent maintenance scheduling based on the coordination between central operator and decentralized producers in an electricity market. Reliability Engineering & System Safety, 210, 107495.
14. Zhang, Y., Yang, Q., An, D., & Zhang, C. (2021, May). Coordination between individual agents in multi-agent reinforcement learning. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 35, No. 13, pp. 11387-11394).
15. Zhao, H., Meng, X., & Wu, S. (2021). Distributed edge-based event-triggered coordination control for multi-agent systems. Automatica, 132, 109797.
16. Terán, J., Aguilar, J., & Cerrada, M. (2017). Integration in industrial automation based on multi-agent systems using cultural algorithms for optimizing the coordination mechanisms. Computers in Industry, 91, 11-23.
17. Liang, Z., Zhou, K., & Gao, K. (2019). Development of virtual reality serious game for underground rock-related hazards safety training. IEEE access, 7, 118639-118649.
18. Tabak, M., de Vette, F., van Dijk, H., & Vollenbroek-Hutten, M. (2020). A game-based, physical activity coaching application for older adults: design approach and user experience in daily life. Games for health journal, 9(3), 215-226.
19. Lowing D., Techer K. (2022). Priority relations and cooperation with multipleactivity levels. J of Math Eco. 14 juill 102740.
20. Lin, C., Kou, G., Peng, Y., & Alsaadi, F. E. (2020). Aggregation of the nearest consistency matrices with the acceptable consensus in AHP-GDM. Annals of Operations Research, 1-17.
21. Li, X., Li, J., Hu, X., & Yang, J. (2020). Line-cnn: End-to-end traffic line detection with line proposal unit. IEEE Transactions on Intelligent Transportation Systems, 21(1), 248-258.
22. Lin, W. C., Lin, C. L., Hsu, P. M., & Wu, M. T. (2014). Realization of anti-lock braking strategy for electric scooters. IEEE Transactions on Industrial Electronics, 61(6), 2826-2833.
23. Savani, R., & Von Stengel, B. (2015). Game Theory Explorer: software for the applied game theorist. Computational Management Science, 12, 5-33.
24. Alwen, J., Blocki, J., & Pietrzak, K. (2017, April). Depth-robust graphs and their cumulative memory complexity. In Advances in Cryptology–EUROCRYPT 2017: 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30–May 4, 2017, Proceedings, Part III (pp. 3-32). Cham: Springer International Publishing.
25. Lemonnier, S., Brémond, R., & Baccino, T. (2015). Gaze behavior when approaching an intersection: Dwell time distribution and comparison with a quantitative prediction. Transportation research part F: traffic psychology and behaviour, 35, 60-74.
26. Ostendorf, B. (2011). Overview: Spatial information and indicators for sustainable management of natural resources. Ecological indicators, 11(1), 97-102.