

Article

Quantum Game Application to Recovery Problem in Mobile Database

Magda M. Madbouly¹, Yasser F. Mokhtar² and Saad M. Darwish^{1,*} 

¹ Department of Information Technology, Institute of Graduate Studies and Research, Alexandria University, Alexandria 21526, Egypt; mmadbouly@hotmail.com

² IT Department, Egyptian Petrochemicals Co., Alexandria 4669, Egypt; yasserfakhry@hotmail.com

* Correspondence: saad.darwish@alexu.edu.eg

Abstract: Mobile Computing (MC) is a relatively new concept in the world of distributed computing that is rapidly gaining traction. Due to the dynamic nature of mobility and the limited bandwidth available on wireless networks, this new computing environment for mobile devices presents significant challenges in terms of fault-tolerant system development. As a consequence, traditional fault-tolerance techniques are inherently inapplicable to these systems. External circumstances often expose mobile systems to failures in communication or data storage. In this article, a quantum game theory-based recovery model is proposed in the case of a mobile host's failure. Several of the state-of-the-art recovery protocols are selected and analyzed in order to identify the most important variables influencing the recovery mechanism, such as the number of processes, the time needed to send messages, and the number of messages logged-in time. Quantum game theory is then adapted to select the optimal recovery method for the given environment variables using the proposed utility matrix of three players. Game theory is the study of mathematical models of situations in which intelligent rational decision-makers face conflicting interests (alternative recovery procedures). The purpose of this study is to present an adaptive algorithm based on quantum game theory for selecting the most efficient context-aware computing recovery procedure. The transition from a classical to a quantum domain is accomplished in the proposed model by treating strategies as a Hilbert space rather than a discrete set and then allowing for the existence of linear superpositions between classical strategies; this naturally increases the number of possible strategic choices available to each player from a numerable to a continuous set. Numerical data are provided to demonstrate feasibility.

Keywords: mobile computing; quantum game theory; decision making; mobile database recovery



Citation: Madbouly, M.M.; Mokhtar, Y.F.; Darwish, S.M. Quantum Game Application to Recovery Problem in Mobile Database. *Symmetry* **2021**, *13*, 1984. <https://doi.org/10.3390/sym13111984>

Academic Editors: Peng-Yeng Yin, Ray-I Chang, Youcef Gheraibia, Ming-Chin Chuang, Hua-Yi Lin, Jen-Chun Lee and Sergei D. Odintsov

Received: 27 July 2021

Accepted: 15 October 2021

Published: 20 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The mobile database system (MDS) is a client/server database management system provided via the internet that allows for the mobility of the whole processing environment. While the database itself may be static and spread across many sites, data processing nodes such as laptops, PDAs, and cell phones may be mobile and access required data from any place and at any time. A mobile host (MH) operating a client-server application may rapidly fail due to limited network resources. Client-server application failure recovery needs significant attention due to the scope of its usage. When applied to the mobile computing environment, traditional recovery methods such as checkpointing, logging, and rollback recovery suffer from many limitations [1–5].

Using checkpoint and message logging techniques, the mobile application may roll back to the last reliably stored state and resume execution with recovery assurances. Existing approaches operate under the assumption that MH disk storage is insecure and store checkpoint and log data at base stations [6–9]. The process for a mobile checkpoint may be coordinated or uncoordinated. To maintain a consistent and recoverable global checkpoint, distributed applications need MHs to coordinate their local checkpoints. Because the MH can independently checkpoint its local state, uncoordinated protocols are preferred

for mobile applications [7–9]. Recovery methods that are not coordinated are either non-logging or logging [4,5]. No-logging MH must generate a new checkpoint whenever the application's state changes. The logging method periodically generates checkpoints and records all write events between them. When an MH attempts to recover from a failure, it makes use of the checkpoint and any previously stored log data. The survey [1,9] compared performance with and without logging.

Numerous factors have an effect on recovery [1,3]: (1) the failure of the MH, for example, due to a bad wireless connection or inadequate battery capacity, is entirely random. If further failures occur, the transaction must roll back whenever MH recovers from a failure, increasing the total execution duration of the transaction; (2) Log Size: data transmission consumes twice the amount of energy required for data receipt. As a result, only important write events should be recorded to maintain a short log; (3) Memory Constraints: the base station controller (BSC) may need a significant amount of memory in order to store the log file for each MH. Calculating the average memory need for logs of different lengths and recovery techniques is essential; (4) Time required for recovery: the time required to recover a process after a failure varies according to the recovery method used and the technology used to capture write events; (5) Cost of log retrieval: the cost of reclaiming log information after the failure of a transaction is related to the amount of log dispersion. When a log is dispersed over several places, the costs of retrieval and recovery increase [10–13].

At present, academics are addressing wireless communication problems using game-theoretic methods [14,15]. In comparison to more conventional approaches, game theory offers a number of benefits. Game theory is concerned with a range of problems involving the strategic interaction of many individuals with conflicting goals in a competition. As a consequence, game theory is an inherently useful tool for describing the rational behavior of many players. Second, game theory may be utilized to model agent-agent interactions, to analyze equilibrium, and to develop distributed algorithms. Additionally, game theory is capable of analyzing hundreds of potential outcomes prior to determining the best course of action.

The Nash Equilibrium (NE) is the state of affairs in which no player can unilaterally enhance their reward, while the Pareto Optimum (PO) is the state of affairs in which no player can unilaterally increase their reward without impairing the advantage of another player. Both are optimal for the individual player, but the latter is usually preferable for the whole team [14,15]. Quantum game theory has developed as a paradigm for analyzing the competitive flow of quantum information in the recent past [16]. The phrase “quantum game” refers to novel applications of quantum information processing, such as competitive agent interactions. In contrast to conventional communication, applications may be created based on the interactions of entangled agents. Competitive von Neumann games, such as quantum auctions and voting, are enabled by entangled resources. These are in contrast to cooperative games, which allow agents to interact directly or indirectly via a third party. Entanglement is a quantum resource that may be utilized to optimize known game-theoretic equilibrium outcomes [16–19]. Figure 1 depicts the classical and quantum versions of a four-player game. Although players in the quantum version use an entangled state as a resource, neither version allows for player communication. The bar graph depicts the equilibrium payoffs for the minority game, while the quantum case's payoffs were determined experimentally using our technology.

Motivation and Contribution

In most mobile application recovery methods, environmental factors were not taken into account as potential influencers on the recovery process. As a result, using recovery methods in the real world is challenging [20]. The goal is to create a strategy that optimizes achievement by using the most effective recovery techniques available in light of the present circumstances. We selected quantum game theory over conflict analysis or interactive decision theory because it enables us to compare the recovery techniques'

available alternatives. The suggested model makes a significant contribution by enabling effective MDS recovery treatment by using a new smart strategy centered on players (various recovery procedures) inside the quantum game theory paradigm as a decision-maker for choosing the most efficient recovery process. Because the critical problem is not to choose one of the well-known recovery techniques, but to choose the strategy that is most appropriate in light of the changes made by the operating environment, which is often vague and unpredictable. In this respect, the present study will guarantee that the optimal approach to recovery through the quantum game theory model is selected based on its key parameters. This research examines many different types of recovery methods. These methods show the effect of different factors on the protocol's complicated efficiency. The suggested model demonstrates a high degree of adaptability via the use of cutting-edge recovery methods capable of substantially increasing performance.

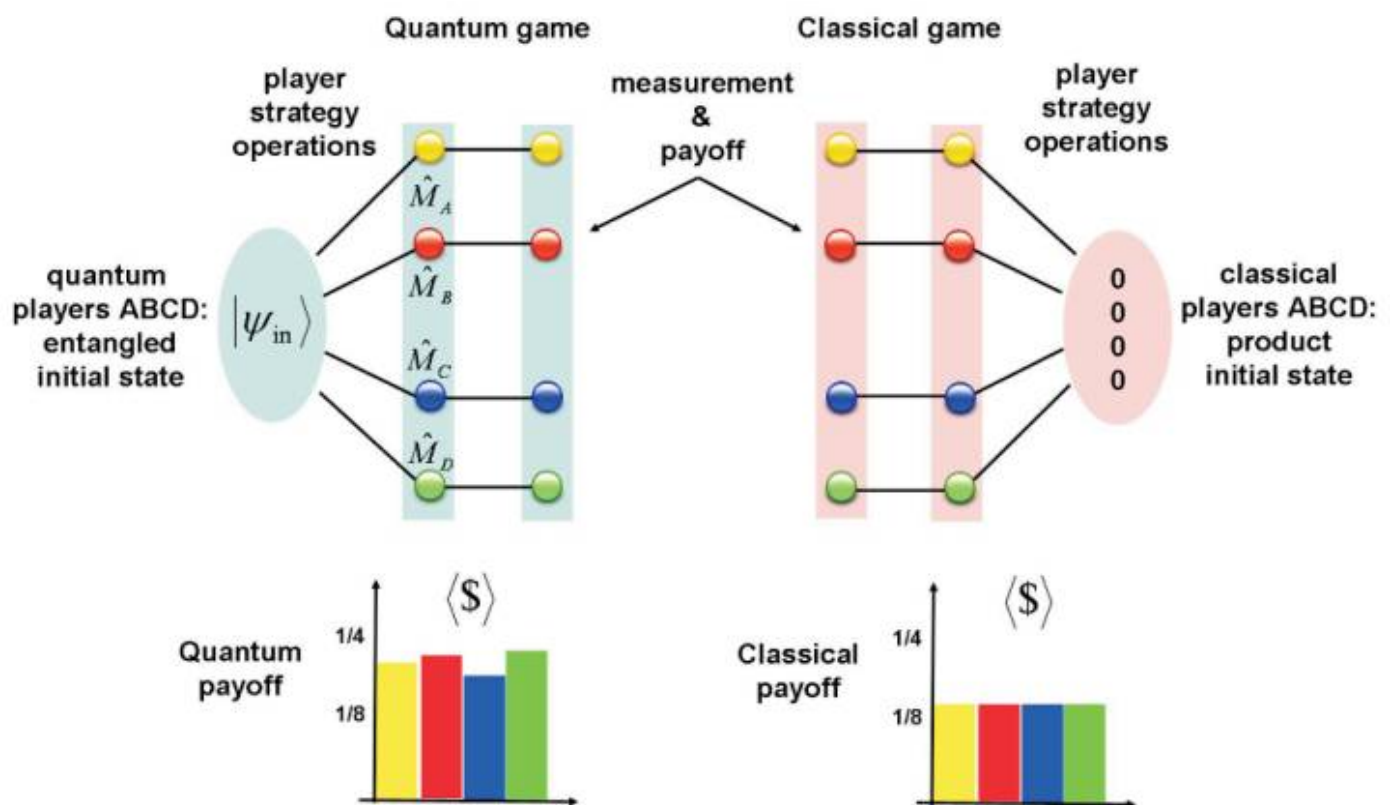


Figure 1. Diagram depicting the information flow in a four-player classical game (right) and a four-player quantum game (left).

To our knowledge, this is the first time that quantum game theory has been used to simulate recovery in a mobile database system. This work extends our conference paper [21] significantly by using quantum game theory rather than classical game theory to successfully deal with NE in order to increase the payoffs of well-known game-theoretic equilibrium. Additionally, three recovery algorithms were employed as players in the quantum game theory, rather than two in our earlier work, to allow a greater number of algorithms to join the competition, which enhances decision-making effectiveness under varying environmental circumstances. The transition from two-player to multiplayer games unquestionably results in more dynamic problem solving, and this mode is the most accurate representation of reality. Single-game dynamics are generally constrained, and trajectories may show a variety of limiting characteristics. When additional participants are involved, both qualitative and statistical dynamics may change. Analytic techniques are used to assess the suggested model's performance, and the results are given later in the article.

A basic notion in computer science, especially in the theoretical realm, is symmetry. Symmetry is crucial for algorithm design and analysis. The use of symmetry simplifies and speeds up probabilistic analysis of stochastic algorithms. With the symmetry idea at its core, the suggested model effectively deals with random parameter probabilities, as shown in the development of game theory's utilization functions.

In addition to this introduction, the following sections are included: Section 2 discusses current state-of-the-art mobile database recovery strategies, Section 3 introduces the proposed mobile database recovery model, Section 4 describes the criteria for evaluating the proposed mobile database recovery based on game theory and presents the results, and Section 5 concludes and suggests directions for future work.

2. State-of-the-Art

In recent years, mobile application failure recovery has grown in popularity. The suggested solutions use checkpointing, logging, or a combination of the two, while taking into account the inherent constraints of the mobile computing environment. Much research is based on a distributed uncoordinated checkpointing method, in which several MHs may achieve a globally consistent checkpoint without depending on coordination messages. Others presented a checkpointing-only method, which offers globally consistent checkpoints without requiring additional communications but is unique in that it uses time to synchronize checkpoint creation.

The authors of [22] described a technique for mobile database applications to recover from checkpointing and logging failures depending on mobility. Current methods use periodic checkpoints that are not dependent on user mobility. This system initiates checkpoints only after a certain number of mobility handoffs has occurred. The failure, log arrival, and mobility rates of the mobile host determine the optimum threshold. This enables modification of the checkpointing rate on a per-user basis. Additionally, depending on the checkpoint frequency, the last checkpoint may be situated a considerable distance from the mobile support station (MSS). Additionally, a significant number of logs across several MSSs may be scattered, resulting in a lengthy recovery time.

The authors of [23] suggested a technique for recovering applications in a mobile computing environment by combining movement-based checkpoints with message recording. A node's adaptability is used to decide if a checkpoint should be taken. This method was developed using a variety of factors, including the number of MH registrations in an area, the number of regions, and the number of handoffs. This approach is especially beneficial in large networks with many areas. In contrast, operating in restricted areas may result in extra expenses.

The authors of [24] developed a rollback recovery method that prioritized separate checkpoints and message recording. The algorithm is unique in that it manages message logs and checkpoints through mobile agents. Additionally, if a mobile node travels a great distance from its most recent checkpoint, the agents are able to move the checkpoint and message logs stored in distant mobile service stations. Thus, the time needed to retrieve a mobile node would never exceed a specified threshold. It is feasible to keep just one checkpoint in permanent storage by recording messages. The main advantage of this study is the modest size of the message log, which cannot be very large owing to the network's low message substitution rate. Additionally, if a process interacts often, it may decrease its checkpointing interval. Nonetheless, this method occurs in a small number of situations, resulting in increased network activity during recovery. Specifically, if the length of the mobility profile exceeds the number of different mobile service stations at any point, the logs must be consolidated into a single place.

The authors in [25] prepared a proposal for a contemporary checkpointing method that is suitable for mobile computing systems. This method is characterized by its dependability and efficiency in terms of time-space overhead associated with checkpointing and normal application execution. The work presented in [26] suggested a log management and low-latency no-blocking checkpointing system that utilizes a mobile-agent-based architecture

to reduce recovery time. By decreasing the amount of messages sent, this protocol reduced recovery time. On the contrary, particularly when many agents are needed, it may result in an increase in complexity, which may absorb some of the extra execution costs.

The authors of [27] developed a log management strategy for mobile computing systems that substantially lowers the total cost of failure recovery when compared to existing lazy and pessimistic approaches. Additionally, their approach enables recovery from a base station different than the one where it failed, lowering handoff costs, log replication costs, and the time required to recover from failure. The main benefit of their log management method is its ease of implementation, whereas the primary drawback is likely the recovery time if the home agent is situated a great distance from the mobile unit [28,29]. The authors of [30] described a recovery technique that is database and mobile device synchronization-dependent. As a consequence, the replication process guarantees that all organizations have consistent data. One drawback of this method is that, although it utilizes hash functions, it does not guarantee data integrity during transmission to the server, since both ends store the hash values in a database table.

The Need to Extend the Related Work

According to the review, the following are the current areas of research: (1) The majority of recovery studies employed a variety of techniques, including log management, checkpointing, movement-based checkpointing, and an agent-based logging scheme; (2) Because these techniques are so dissimilar, one cannot be used in place of another; this means that each algorithm has a distinct parameter set and different assumptions; (3) Despite the fact that some plans tried to merge several methods into a single contribution (hybrid method), they were damaged by the difficulties of selecting the optimal fusion from this pool of options. As a consequence, recovery costs may be high and the recovery mechanism may be excessively complicated; (4) The majority of schemes did not include environmental variables as influencing elements in the recovery process; and (5) As the demand for network applications grows, researchers are continuously developing new ways to solve the issue of high mobility or network connection loss owing to a variety of new or changing conditions. Thus, more fault-tolerant methods are needed to guarantee the continued functioning of mobile devices. As a consequence of the above, the use of recovery algorithms is constrained in a realistic manner. It is essential to design a plan that maximizes success via the selection of the most suitable recovery methods for the present situation. We selected quantum game theory over conflict analysis or interactive decision theory, as it enables us to compare the recovery possibilities available.

3. Quantum Game-Based Recovery Model

3.1. Mobile System Architecture

In a typical MDS design, a small database fragment is created from the main database on the MH. This design is meant to handle the accessibility limitations alleviated by MHs and mobile satellite services (MSS). If the MH is present in the cell serviced by the MSS, it may interact directly with another MH in the vicinity. The MH may freely move between cells, which each include a base station (BS) and a large number of MHs. Additionally, the BSs configured the stations to act as a wireless gateway, allowing them to communicate with the MHs and send data via the wireless network. Wireless communication is possible between the MHs and the BS, but not directly with the database server [11,31]. Figure 2 depicts the mobile system's architecture.

3.2. Recovery Modeling Using Quantum Game

The suggested method differs from prior MDS recovery attempts in that it takes into account a variety of important variables in the mobile environment during hand-offs or service failures, which change depending on the situation, while conventional recovery algorithms are predicated on specific assumptions about the environment and operate accordingly.

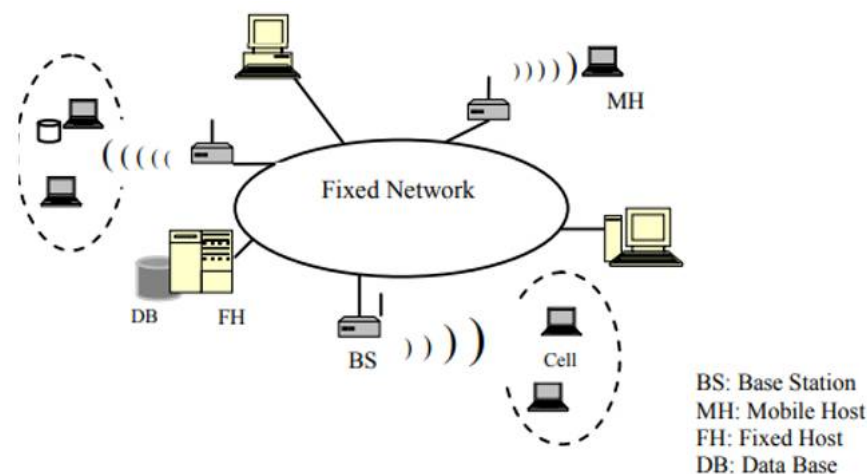


Figure 2. Mobile System Architecture.

The primary reasons for extending our prior work's two-player game to three players in this article are as follows: (1) reliance on a limited number of algorithms lowers the possibility of making good decisions, since it is possible for one of the algorithms to permanently dominate decision-making; (2) Allowing a greater number of algorithms to join the competition increases the efficacy of decision-making under a variety of environmental circumstances; (3) With a rise in the number of players, different degrees of complexity and mathematical calculations were used to address the recovery issue, resulting in enhanced capabilities for the proposed task; and (4) Several algorithms performed poorly in the present study, despite their success in earlier work. Thus, when the performance of certain algorithms deteriorated, they were eliminated from the competition; nevertheless, the entrance of others with superior results resulted in a substantial increase in performance, which is the purpose of presenting this study. The suggested model's architecture is shown in Figure 3. The following table (Table 1) summarizes the game assumptions utilized in the recovery modeling process.

Table 1. Game Assumptions.

Game	Parameter
Number of Players	Three Players
Game Type	Non Cooperative (No interaction between players)
Game Form	Strategic form
Evolutionary Game Theory	No
Strategy type	Pure not Mixed
Payoff Functions	TIME, Memory, Recovery Done Probability
The Winning Algorithm	The highest NE(Quantum Nash) in the reward matrix

Cooperative game theory (CGT) and non-cooperative game theory (NCGT) are two subfields of game theory. CGT elucidates how agents compete and cooperate to generate and capture value in unstructured interactions. NCGT simulates agents' activities, maximizing their usefulness based on a comprehensive description of each agent's motions and information. Cooperative games are ones in which players are convinced to follow a certain strategy via player dialogue and agreement. A strategy is a detailed plan of action that a player will follow in response to a variety of situations that may occur over the course of the game. On the other hand, non-cooperative games are ones in which participants select their own strategy of profit maximization. The main distinguishing feature is the absence of external authority to establish norms guaranteeing cooperative behavior. Without external authority (such as contract law), participants are unable to form coalitions and must compete alone [14,15].

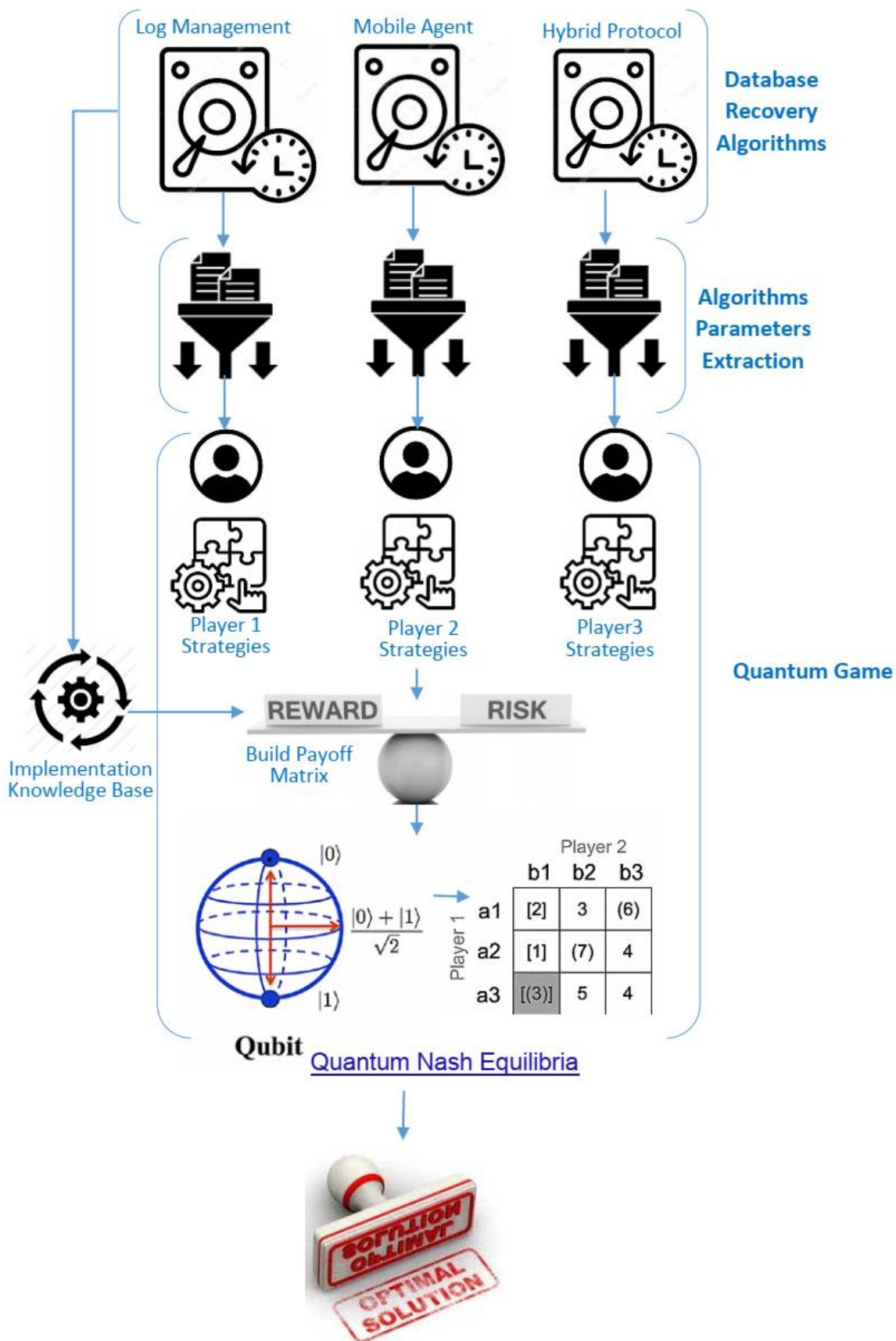


Figure 3. The Proposed MDS Recovery Model.

Non-cooperative games are often studied by trying to predict players' tactics and payoffs, as well as by finding Nash equilibria. Each player in NE is assumed to be aware of the other players' equilibrium tactics, and no one benefits from merely changing their strategy. If each player has chosen a strategy—a collection of actions based on previous game events—and no person can increase their expected payoff by changing their strategy while the other players retain theirs, then the current set of strategy choices characterizes NE.

The suggested game is modeled by static games with complete information, in which players simultaneously choose strategies and get rewards depending on the combination of actions taken. These types of games may be formalized using a normal-form representation [14]. This is a simple decision issue in which both players choose their actions concurrently (static game) and are rewarded for their mutual choices. Additionally, each player is fully aware of the values associated with his adversary's reward functions (complete information). The interrelationships of game theory are shown in Figure 4; for more information, see [14,15].

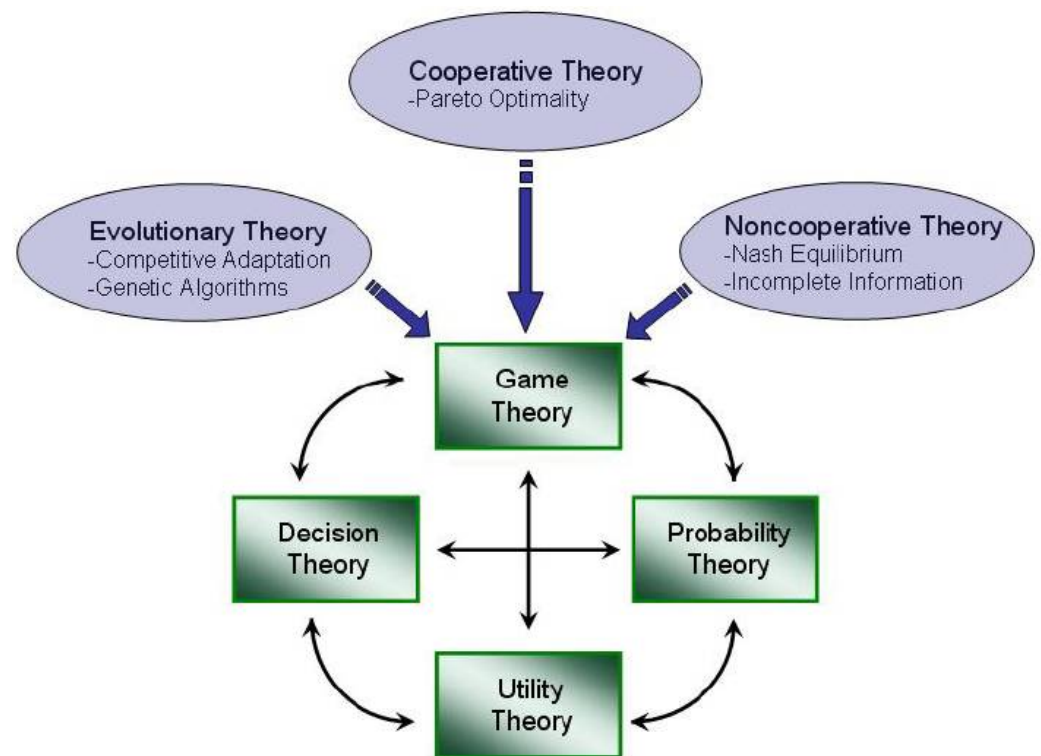


Figure 4. Game Theory Interrelationship Diagram.

3.2.1. Recovery Algorithms and Its Parameters

To demonstrate the technical importance of the system recovery model in the MDS, we evaluated the most widely used MD recovery algorithms in order to decide which methods should be explored further. We classified the recoverable algorithms in this situation according to their operation or features. As stated in [9–13], various groups differ in their approach to recovery. For our proposal, we selected three recovery protocols: log monitoring (as player 1) [27], mobile agent (as player 2) [24], and a hybrid method that combined movement-based check-pointing and message recording as a (as player 3) [22]. Because the real problem is not whether to adopt one of the well-known recovery techniques, but rather which strategy is most appropriate in light of the changes imposed by the operational environment, which is often unclear and changing. In this regard, the current research will ensure that the optimal recovery method is selected via the use of game theory and its essential variables.

To compete against other players, each player must create a set of strategies. To develop these methods, feature analysis and extraction are used for each chosen recovery procedure in order to determine the treatment's most strongest characteristics. Thus, in game theory, each selected protocol is defined by a player, and the way each protocol's variable is utilized determines the player's strategies. For instance, the first protocol (player 1) considered a variety of factors, including the log arrival rate, the handoff rate, the average log size, and the mobility rate. To summarize, the first player method involves retrieving the log file that was stored in the BS before to the failure, moving it to a new location linked to another BS, and updating the MH. The second protocol (player 2) makes extensive use of variables, including the number of processes in the checkpoint, a handoff threshold, and the length of the log. The second approach reduced recovery time by using a framework based on mobile agents. Here is a collection of processes in a list format. The home agent's list is included in the MH. The mobile agent traveled beside the MH and relayed information. The third process takes into account a variety of factors, including the total number of registrations in the area, the total number of regions, and the overall number of hand-offs. The work environment is split into several zones, and the checkpoint is only used once when MH enters and exits the region. For further information on how these protocols work, see [13–27].

It should be emphasized that any number of strategies for each player (protocol) may be produced by conducting any number of trials with various parameter values, although this increases the model's complexity. As a result, we believe it is important to choose several methods that reflect varying degrees of performance that may be depended upon in decision-making. To create the parameters for the necessary recovery algorithms using game theory as a decision-making method, we first apply the chosen protocols to the chosen key variables on which each protocol is reliant. Each algorithm is used for real database transactions in order to assess each player's strategy. A package is assessed that has an objective function for the total cost of recovery, which is computed differently for each method. In game theory, we build the payoff matrix for each protocol output value based on the previous stages. These outcomes are referred to as the utility or reward of each player. These payoffs or benefits are used to evaluate a player's level of satisfaction in a conflicting scenario.

In general, game theory may be summarized as follows: (1) a set of players (the negotiation algorithms chosen); (2) a pool of strategies for each player (the strategies take into account the assumed values of significant coefficients in each protocol, as well as possible environmental changes); and (3) the benefits or payoffs (utility) to any player for any possible list of the players' chosen strategies. It should be noted here that any number of strategies can be generated for each player (protocol) by running any number of experiments using different parameter values, but this of course increases the model's complexity. Therefore, we consider it best to choose a number of strategies to express different degrees of performance levels that can be relied upon in decision-making. In the suggested game-based recovery model, the assumption is that each player utilizes pure strategy and not a mixed one, as each strategy handles specific protocol parameters, and these parameters differ from one protocol to another. There are no general parameters used for all protocols. For pure strategies, it is far easier to obtain multiple solutions (of course if they exist) for the NE, and then select the best fitting one.

3.2.2. Build Knowledge Base

The suggested recovery model is predicated on the establishment of a knowledge base after the pre-implementation of each chosen recovery procedure in various simulated settings. The implementation knowledge base is created just once, and it is used to choose the optimal protocol based on the reward matrix and dominant equilibrium method. The decision is done here on the basis of the integration of three utility functions that serve as performance and evaluation benchmarks for the candidate protocols.

3.2.3. Build Payoff Matrix

In the proposed model’s game, see Table 2, a finite collection of players $N = 3$ was used. Three players are allocated $P_1, P_2,$ and P_3 in the three-player game. P_1 selects pure strategies from S_1, P_2 selects pure strategies from $S_2,$ and P_3 selects pure strategies from $S_3.$ If P_1 chooses the pure strategy i, P_2 chooses $j,$ and P_3 chooses $k,$ then the reward to P_1 is a_{ijk}, P_2 chooses $b_{ijk},$ and P_3 chooses $c_{ijk}.$ Define the $S_1 \times S_2 \times S_3$ payoff “cubes” $A, B,$ and C as [32,33]:

$$A = [a_{ijk}] \in R^{S1 \times S2 \times S3}, B = [b_{ijk}] \in R^{S1 \times S2 \times S3}, C = [c_{ijk}] \in R^{S1 \times S2 \times S3} \quad (1)$$

We create a matrix for each of player 3’s actions (strategies); accordingly, player 1 selects a row, player 2 selects a column, and player 3 selects a table. Table 3 illustrates the bi-matrix for a three-player game with its payoff. As a result, each third player strategy k is represented separately in a matrix together with its associated reward in terms of player 1 and player 2’s strategies. Here, a_{111} represents the payoff value for player 1’s plan or strategy given the return function $u_1(s_1, t_1, c_1)$ if (s_1, t_1, c_1) is selected, b_{111} denotes the payoff value for player 2’s strategy provided the reward function $u_2(s_1, t_1, c_1)$ if (s_1, t_1, c_1) is chosen, and c_{111} denotes the payoff value for player 3’s strategy with payoff function $u_3(s_1, t_1, c_1).$ The proposed recovery model’s game is a non-cooperative game in which all participants choose their own tactics in order to maximize their profit [15, 32,33]. These kinds of games are amenable to formalization via the use of normal-form representations [34]. In a normal-form game, player i ’s strategy S_i'' strictly trumps player i ’s strategy S_i' if and only if:

$$U_i (S_i'', S_{-i}) \geq U_i (S_i', S_{-i}) \quad (2)$$

for every list of S_{-i} of the other players that represents all players’ strategies except player $i.$

Table 2. The symbols used for description game theory model.

Symbol	Meaning
G	Quantum game model
N	Number of players
i	Game player
S	Player’s strategy
U	Payoff function or utility
B_i	Best response for a player i

Table 3. The bi-matrix for three players’ game.

Player	Player 3 Hybrid Method (Strategy k)				
	Strategy	t_1	t_2	...	t_h
Player 1 Log Management Method	s_1	$(a_{111}, b_{111}, c_{11k})$	$(a_{121}, b_{121}, c_{12k})$...	$(a_{1h1}, b_{1h1}, c_{1hk})$
	s_2	$(a_{211}, b_{211}, c_{21k})$	$(a_{221}, b_{221}, c_{22k})$...	$(a_{2h1}, b_{2h1}, c_{2hk})$
	
	s_m	$(a_{m11}, b_{m11}, c_{m1k})$	$(a_{m21}, b_{m21}, c_{m2k})$...	$(a_{mh1}, b_{mh1}, c_{mhk})$

Calculating the payoff in a game is complicated since it is dependent on the actions of other players. As a result, the strategy chosen by one player has an effect on the gain value of the other player. Three utility functions are included in this proposal as performance and assessment benchmarks for the candidate protocols. The functions are as follows: the amount of time consumed by the protocol during operation ($TIME_i$) for each strategy, the amount of memory consumed during operation ($MEMO_i$) for each strategy, and the rate

expressing the percentage of recovery work completed (“recovery completion”) for each strategy ($DONE_PROB_i$). The point of this step is to determine the reward that the player will gain if its strategy wins according to the mobile environmental conditions. Since all players are assumed to be rational, they make their preferred decisions that maximize their rewards (payoff). Consequently, one player’s strategy dominates another player’s strategy if it always provides a greater payoff to that player regardless of the strategy played by the opposing player. Therefore, it is very important to determine the method of calculating the return for each player’s strategy. Therefore, the aim of these functions is to evaluate every strategy by calculating an index (score) that represents its performance. Every function contains degrees to distinguish the better performance of each strategy with high degrees against the lower performance of all the utility functions.

After analyzing and executing the protocols, it was determined that each algorithm operates within a time range of 0 to 5 s, implying that the value of the return function from the time measurement would be distributed as follows:

$$if \left\{ \begin{array}{ll} C_{1,i} \in]0.1, 0] & u_i = 6 \\ C_{1,i} \in]0.5, 0.1] & u_i = 4 \\ C_{1,i} \in]0.9, 0.5] & u_i = 2 \\ C_{1,i} \in]1, 0.9] & u_i = 0 \\ C_{1,i} \in]2, 1] & u_i = -2 \\ C_{1,i} \in]5, 2] & u_i = -4 \end{array} \right. \quad (3)$$

where $C_{1,i} = TIME_i$ is the time required to execute a strategy. $C_{2,i} = MEMO_i$, in the same context, is the amount of memory used by each protocol for a given strategy. The memory used during execution is expected to be between 0 and 4000 KB; therefore, the payoff values for the memory consumed by any strategy will be as follows:

$$if \left\{ \begin{array}{ll} C_{2,i} \in]500, 0] & u_i = 6 \\ C_{2,i} \in]1000, 500] & u_i = 4 \\ C_{2,i} \in]1500, 1000] & u_i = 2 \\ C_{2,i} \in]2000, 1500] & u_i = 0 \\ C_{2,i} \in]2500, 2000] & u_i = -2 \\ C_{2,i} \in]4000, 2500] & u_i = -4 \end{array} \right. \quad (4)$$

Finally, when calculating the completion level of the recovery process, $C_{3,i} = DONE_PROB_i$, where $DONE_PROB_i$ is utilized to determine if recovery occurred in accordance with the handoff rates threshold. As a result, the value of the possible return measure for this work ranges from 0% to 100% and is distributed as follows:

$$if \left\{ \begin{array}{ll} C_{3,i} \in]20\%, 0] & u_i = 1 \\ C_{3,i} \in]40\%, 20\%] & u_i = 2 \\ C_{3,i} \in]60\%, 40\%] & u_i = 3 \\ C_{3,i} \in]80\%, 60\%] & u_i = 4 \\ C_{3,i} \in]100\%, 80\%] & u_i = 5 \end{array} \right. \quad (5)$$

3.2.4. Quantum Nash Equilibrium for Selection

Thus, the player’s overall gain in this game is equal to the sum of the reward values associated with the variables ($C_{1,i}$, $C_{2,i}$, $C_{3,i}$). The ultimate solution may be obtained in one of two ways: (1) by achieving a single and exclusive dominant equilibrium method in the game, or (2) by using NE [15]. The strategies produced via the first method, dubbed iterated elimination of strictly dominated strategies, reflect the optimal actions that each player might rationally take, and therefore comprise the game’s (rational) solution. Regrettably, this alluring approach yields no prediction at all for some kinds of situations in which no strategy survives the elimination phase. In this situation, it is unclear which course of

action would be deemed reasonable and best. This is a unique game in which there are no absolutely dominant strategies.

NE is a game theory concept that determines the optimal result in a non-cooperative game in which each player has little incentive to change his or her initial strategy. Under the NE, a player wins nothing by departing from their initial strategy, guaranteeing that the strategies of the other players stay constant as well. A game may include several NE states or none at all [35]. Unfortunately, the majority of games lack dominating strategies. Thus, if there are many solutions (more than one NE) to a given issue, the alternative is to find another handling mechanism. To address this issue, all values in the payoff matrix are subject to an addition or subtraction mechanism based on one of the critical factors, such as execution time, so that more points may be awarded to the quickest element and vice versa (normalization and reduction phase). Finally, the updated payoff matrix is utilized to identify a more optimal solution (Pure Nash) that matches the various environmental factors.

Since all players are assumed to be rational, they make their preferred decisions which maximize their rewards. Consequently, one strategy for a player is dominant over another strategy for another player if it always provides a greater payoff to that player regardless of the strategy played by the opposing player. Therefore, it is very important to determine the method of calculating the return for each strategy for each player. A particular algorithm is selected when its strategy achieves the highest NE in the reward matrix.

In contrast to the classical situation, where the theory is incapable of making any unique prediction, the application of quantum formalism will show a new property: the emergence in entangled strategies of a NE reflecting the unique solution to the game. In the quantum version of this three-player game, players execute their strategies by applying the identity operators they possess with probability p , q , and r to the starting quantum state, respectively. The three players apply the inversion operator σ with probability $(1-p)$, $(1-q)$, and $(1-r)$. If ρ_{in} is the density matrix corresponding to the initial quantum state, then the final state after players have implemented their strategies is [17,18].

$$\rho_{fin} = \sum_{U=I,\sigma} P(H_A)P(H_B)P(H_C)H_A \otimes H_B \otimes H_C \rho_{in} H_A^\dagger \otimes H_B^\dagger \otimes H_C^\dagger \quad (6)$$

where either I or σ may be used as the unitary and Hermitian operator H . $P(H_A)$, $P(H_B)$, and $P(H_C)$ are the probability that players A , B , and C , respectively, will apply the operator H to the initial state. ρ_{in} is a convex combination of all quantum processes. Assume the arbitrator creates the following pure initial quantum state with three qubits (two strategies for each player for simplicity):

$$\begin{aligned} |\psi_{in}\rangle &= \sum_{i,j,k=1,2} C_{ijk}|ijk\rangle \\ \sum_{i,j,k=1,2} |C_{ijk}|^2 &= 1 \end{aligned} \quad (7)$$

where the quantum state's eight basis vectors are $|ijk\rangle$ for i , j , and k equal to one and two. The starting state may be thought of as a global state (in a $(2 \otimes 2 \otimes 2)$ -dimensional Hilbert space) of three quantum two-state systems or 'qubits'. The unitary operators I and σ are used by the player with conventional probability ρ_{in} included into its strategic.

As a consequence, rather of considering just a discrete and finite set of strategies, we will now consider their linear superposition by endowing the strategic space with the formal structure of a Hilbert space. As a result, pure quantum strategies may be constructed, which are characterized as linear combinations of pure classical strategies with complex coefficients. This must be interpreted as the probability of using a single pure classical method. It is worth noting that this interpretation of pure quantum strategies is identical to the classical concept of a mixed strategy introduced previously, because we are currently considering a restricted class of games (static games), which lacks typical quantum interference effects between amplitudes [36,37].

4. Discussion

4.1. Simulation Setup

The simulation is designed to evaluate the proposed model for MD recovery based on game theory. In this respect, we implemented the prototype NS2 program using two software packages: Matlab and NS2. By modeling discrete occurrences, the NS2 simulation software enables developers to improve their businesses in real time. Additionally, it supports a variety of protocols such as TCP, routing, and multicast across wired and wireless networks and runs on a variety of platforms including Linux and Windows [38,39]. As a result, we use this software to implement the stage of collecting data on the work environment at various levels in order to mimic changes in the work environment. Then, Matlab software is used to construct a game theory in order to assess various recovery procedures by inputting the output values from the simulation stage in order to find the optimal decision. For our solution, we used mobile log files of various sizes that included the process data that each method would obtain.

The settings for the NS2 simulation are summarized in Table 4. The MAC layer protocol of IEEE 802.11 for wireless large area networks is utilized here. A movement file provides the mobile client node's motions. The mobile client node transmission range is 250 m. Each cell has one base station. A random waypoint (RWP) model determines the starting node position and movement. The RWP model is based on random locations, speeds, and halt durations. The prototype was built in modules and tested on a Dell Inspiron N5110 laptop from Dell Computer Corporation in Texas. Processor: Intel(R) Core(TM) i5-2410M, 4.00 GB RAM, Windows 7 64-bit. The proposed model's efficiency is assessed using execution time and recovery probability. See [24,27] for more details. The results are the average of several repeated experiments for different initial location and movements of the nodes due to the using of RWP model. The most suitable values for the collection of protocol's factors (strategies) were picked from the literature, based on the assessment of selected recovery protocols, to represent the protocol's performance in a variety of settings.

4.2. Simulation Results

4.2.1. Experiment One

Aim: The first set of experiments was designed to evaluate the performance of the suggested recovery model in terms of actual execution time as a function of log file size. In general, the cost of recovery is very low for any scheme, since the full log information is stored at the current base station. When the MH travels a great distance from the initial BS, the difficulty of the recovery method used to locate and transmit the log file rises. Increasing the file size results in an increase in the transmission cost.

Main Results: As shown in Table 5, the suggested game theory-based recovery model delivers a faster execution time while increasing file size in a variety of mobile settings, depending on the simulation's changing nature.

Discussion: One reason for these results is that, since the proposed model is based on a knowledge base that was developed after the pre-implementation of each chosen recovery protocol in various simulated settings, it chooses the best appropriate recovery protocol for the present circumstance (variation of log file size). As a result, the proposed model requires less time to execute for numerous simulation runs.

4.2.2. Experiment Two

Aim: The second set of experiments evaluated the suggested recovery model's performance as a function of mobility rate in terms of actual execution time. The suggested recovery model is based on the creation of a knowledge base that is created once and used to choose the best suitable procedure based on the reward matrix and dominant equilibrium method. The decision is done here on the basis of the integration of three utility functions that serve as performance and evaluation benchmarks for the candidate protocols.

Table 4. Simulation Parameters.

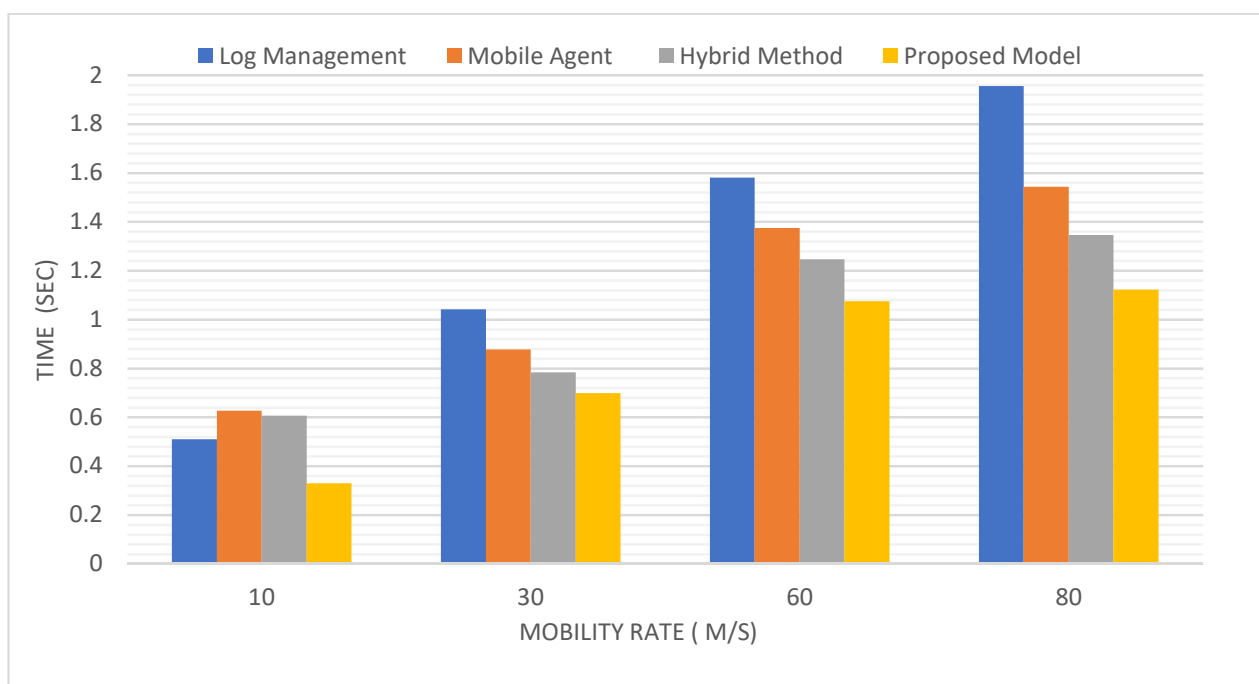
Variable	Meaning
Channel type	Channel/Wireless Channel
MAC type	Mac/802_11
Radio-propagation model	Propagation/Two Ray Ground
Network interface type	Phy/wireless Phy
Interface queue type	queue/drop tail/priqueue
Antenna model	Antenna/Omni Antenna
Link layer type	LL
Routing protocol	Destination-Sequenced Distance Vector (DSDV)
Coordinate of topology	670 m × 670 m
Max packet in <i>ifq</i>	500
Time to stop simulation	250

Table 5. Execution time (Seconds) as a function of log file size.

Algorithm/Log Size	5 KB	30 KB	60 KB	90 KB	120 KB	150 KB
Log Management	1	1.3	1.8	2.5	3.3	3.5
Mobile Agent	1.1	1.5	1.9	2.4	3	3.15
Hybrid Method	1.2	1.6	1.7	2.3	2.8	3
Proposed Model	1	1.4	1.8	2.3	2.7	2.75

Main Results: Figure 5 shows that when mobility rate rises, recovery cost increases. In addition, the current base station has all the log information, lowering the recovery cost. The recovery cost is greater when the mobile node is recovered in the same base station.

Discussion: The results in Figure 5 show that despite their success in limited regions, log management and agent-based methods usually suffer from increased implementation time in the long term. On the contrary, the hybrid approach may be more successful in a vast environment than in a small area since it only requires a checkpoint once before MH moves across regions, thus saving time.

**Figure 5.** Execution time as a function of mobility rate.

4.2.3. Experiment Three

Aim: The final set of experiments examined the connection between the likelihood of recovery and the handoff threshold. There is a favorable correlation between increasing the handoff rate and the likelihood of completing the recovery process. Handoff involves moving a MS from one base station to another. Algorithms with set parameters do not work well in varying system configurations. Handoff algorithms should take into consideration the communication system's peculiarities. If the preparation time of rapid handoff is longer than the WLAN sojourn time linked to mobile node speed, the handoff fails and packets are lost. If the mobile node speed is too slow, handoffs are initiated too late, reducing WLAN service duration. The handoff cost comprises the checkpoint status, message log, and acknowledgement.

Main Results: Figure 6 plots the completion of the recovery process against increases in the hand-off threshold rate. The recovery likelihood is significantly reduced when using the log management technique, but the hand-off threshold rate is raised. Alternatively, some techniques decreased gradually when the threshold value was increased.

Discussion: The results demonstrated that the log technique is applicable only in small work settings. Whereas the hybrid approach and the agent-based method both performed well for areas with multiple regions or regions located farther from the site of retrieval. As anticipated, the suggested model has a higher recovery probability in the long term when compared to the other methods in the various simulated settings.

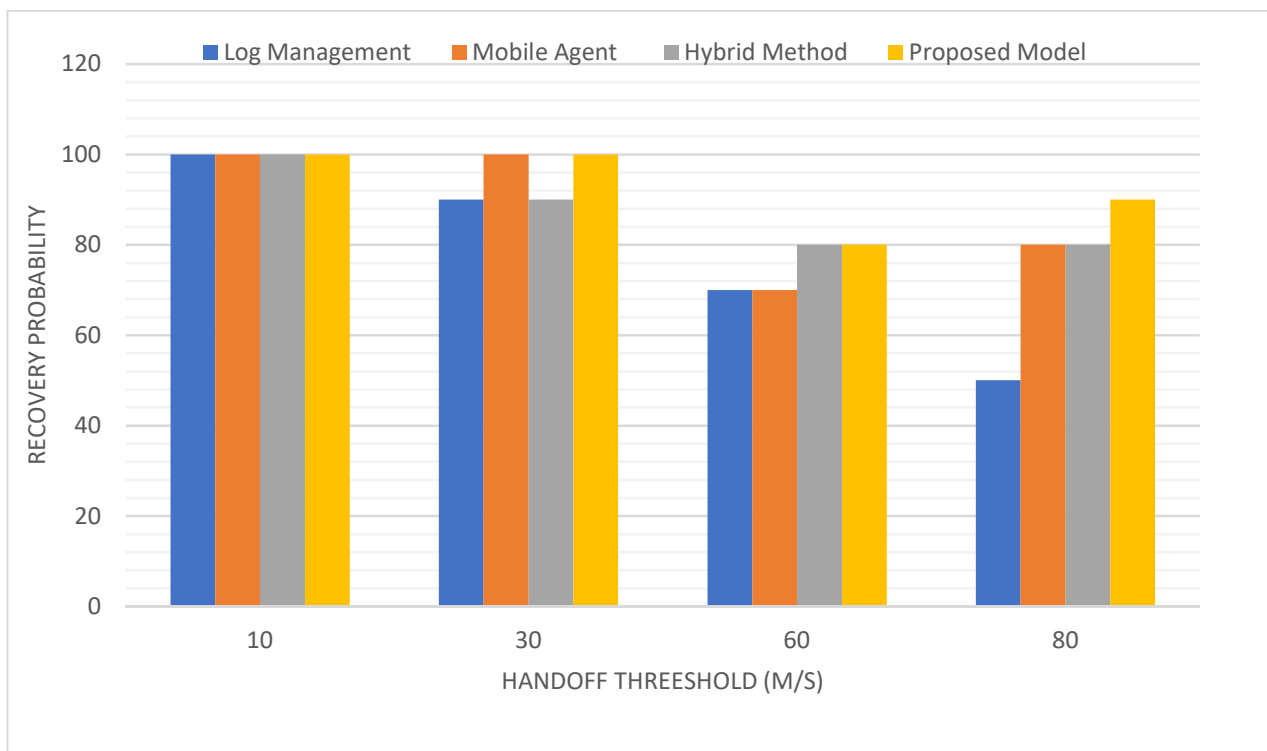


Figure 6. Recovery probability vs. handoff threshold.

4.2.4. Experiment Four

Aim: The next set of experiments was conducted to measure the efficiency of the proposed model compared to traditional recovery methods concerning the degree of complexity of the chosen strategies. In this case, the degree of complexity of the strategies was divided into three levels: low, medium, and high. The difficulty here is measured through utility functions. In the case of the log file size parameter, the range of size changes from small to medium to large is a measure of the complexity of the strategy. The same

is the case for the other parameters such as handoff rate and mobility rate. Herein, the evaluation is based on the total payoff value that is calculated as the sum of the three utility functions' outputs.

Main Results: The results in Figure 7 confirm the superiority of the suggested model. As previously stated, since the proposed model is based on a knowledge base that was created after the pre-implementation of each chosen recovery protocol in various simulated environments, it automatically picks the most appropriate recovery procedure for the present situation.

Discussion: As expected, the log management algorithm is preferred in the small area. However, this decision turns out to be unfavorable with a large log size, especially in distant regions, because the cost of transfer the log file becomes high, and thus the cost of recovery becomes more problematic. On the other hand, the hybrid method gave a good payoff compared to other algorithms whenever there was a multiplicity in the regions because it had taken the recovery point once. The same is true for agent-based recovery, as it is easy to find the recovery location by tracing the MH ids.

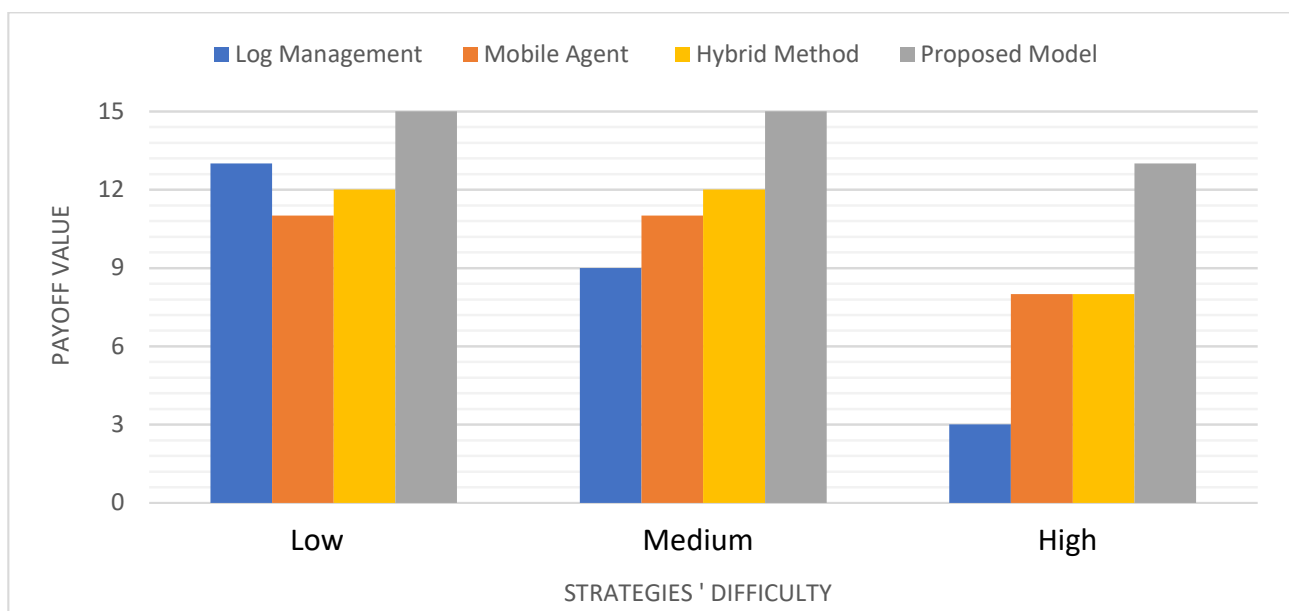


Figure 7. Comparison results in terms of total payoff.

4.2.5. Experiment Five

Aim: A comparative analysis has been done with an optimization model dealing with the recovery of HLR mobility databases presented in [40]. In this work, analytical performance results evaluation for the location updating scheme have been done for the mobility database failure recovery under some assumptions. The lost incoming calls cost is calculated versus the system probability distribution for the recovery time. Therefore, in our proposal, a simulation for a traffic analysis with random log file sizes is done into failure recovery time to calculate the lost packet rates.

Main Results: The results in Figure 8 show a significant decrease in the average lost packets in the proposed model compared to the comparative protocol under different failure recovery time.

Discussion: That is because the proposed model switched between a pool of recovery algorithms that may suitable for any environment rather than using a fixed recovery method that may not be suitable, which leads to the retrieval system not constantly updating its database in a proper time. Thus, it reduces the packet loss rate.

4.2.6. Experiment Six

Aim: The last set of experiments was implemented to compare the suggested recovery model (the extended version with three players) with our previous version (modelling by two players). The precision (accuracy) rate has been used for assessment according to the recovery probabilities for multi degrees of handoff threshold values that are considered as a very influential factor to complete the retrieval process.

Main Results: As expected, Form Figure 9, utilizing three players for game modelling, increases the precision rate by an average of 5% compared with two players for a game.

Discussion: This improvement comes from increasing competition through utilizing more than two recovery algorithms that gave the opportunity to select the best protocol from a pool of algorithms according to the current MDS environmental conditions. Therefore, some protocols used in our previous work [21] will be retreated in this work due to the entry of other protocols in the competition.

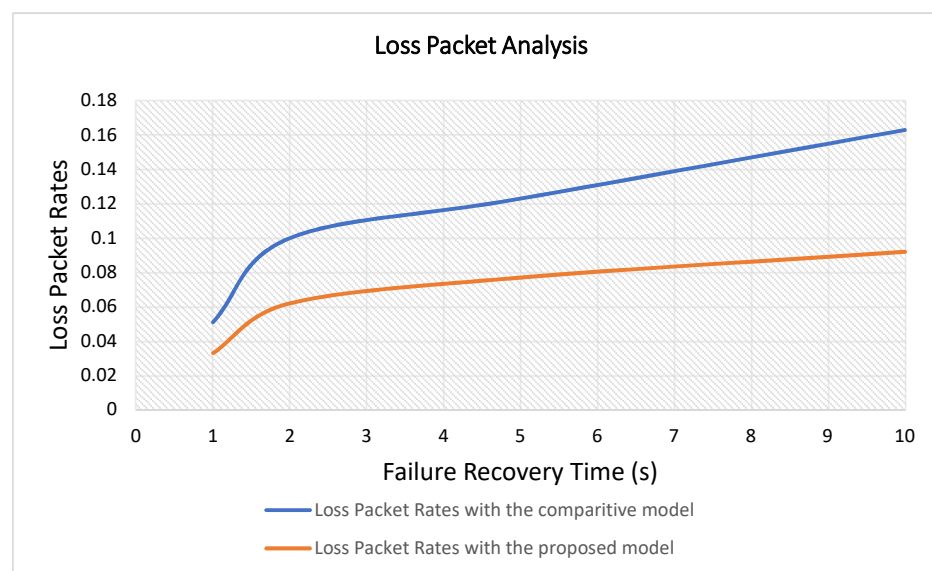


Figure 8. A loss packet analysis with a comparative model.

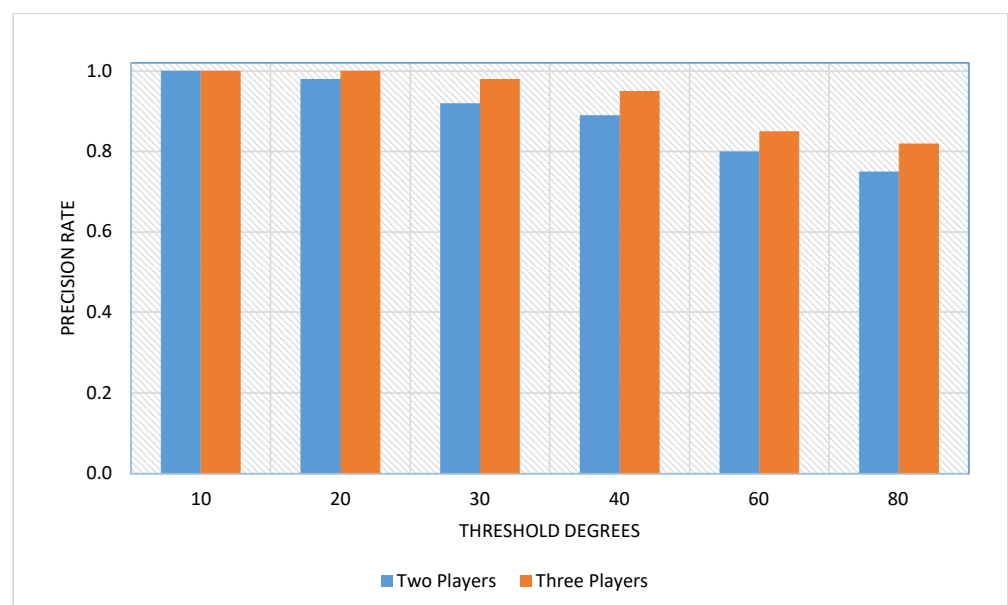


Figure 9. Accuracy analysis between two-player- and three-player-based game modeling.

5. Conclusions and Future Work

The purpose of this paper was to propose a new game theory model for determining the optimal recovery strategy in MSD. The novel method was compared against three of the most commonly used MDS recovery procedures in a competitive environment. Game theory is founded on the idea that each algorithm chooses the most appropriate strategy in terms of message delivery time and message count in order to determine the right recovery solution based on environmental factors. A key step of a quantum game-theoretic research is identifying which strategy to a recovery process is the superior solution to the strategies chosen by others. The proposed recovery model is based on the development of a knowledge base that is used to choose the best appropriate method based on the reward matrix and dominant equilibrium technique. The experimental findings demonstrate the superiority of the suggested recovery paradigm. In the future, it may be essential to include more recovery procedures to optimize the suggested model's performance. Additionally, a hybrid method based on game theory and the recently developed paradigm of cloud algorithms was utilized to improve the outcome. Furthermore, this enhanced the game model to allow interactions between players and utilized mixed strategies. Prior to that, some investigations should be made into the probability distribution of the behavior of the competing players (dealing with uncertainty). The application of the proposed model in the case of large systems and the discussion of its complexity will be also done in future work. Finally, the concept of mind-light-matter unity AI/QI in quantum-inspired computing can be utilized to enhance the suggested model.

Author Contributions: Conceptualization, M.M.M., S.M.D. and Y.F.M.; methodology, S.M.D. and Y.F.M.; software, Y.F.M.; validation, M.M.M. and S.M.D.; formal analysis, M.M.M., S.M.D. and Y.F.M.; investigation, M.M.M. and S.M.D.; resources, S.M.D. and Y.F.M.; data curation, S.M.D. and Y.F.M.; writing—original draft preparation, M.M.M., S.M.D. and Y.F.M.; writing—review and editing, M.M.M. and S.M.D.; visualization, Y.F.M.; supervision, M.M.M. and S.M.D.; project administration, S.M.D. and Y.F.M.; funding acquisition, Y.F.M. 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: The study did not report any data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rathika, R.; Prakash, T.; Aloysius, A. A survey on mobile computing security. *Int. J. Res. Anal. Rev.* **2019**, *6*, 645–649.
2. Anandhakumar, M.; Ramaraj, E.; Venkatesan, N. Query issues in mobile database systems. *Asia Pac. J. Res.* **2014**, *1*, 24–36.
3. Ibikunle, F.; Adegbenjo, A. Management issues and challenges in mobile database system. *Int. J. Eng. Sci. Emerg. Technol.* **2013**, *5*, 1–6.
4. Su, H. The processing technology in mobile database transaction system. *Int. J. Database Theory Appl.* **2015**, *8*, 51–60. [[CrossRef](#)]
5. Patil, P. Mobile computing: Issues and limitations. *Int. J. Comput. Sci. Mob. Appl.* **2016**, *4*, 1–6.
6. Nisar, J.; Trumboo, O.N. Database for mobile application. *Int. J. Trend Sci. Res. Dev.* **2018**, *2*, 853–854. [[CrossRef](#)]
7. Mobarek, A.; Abdelrhman, S.; Abdel-Mutal, A.; Adam, S.; Elbadri, N.; Ahmed, T.M. Transaction processing, techniques in mobile database: An overview. *Int. J. Comput. Sci. Appl.* **2015**, *5*, 1–10. [[CrossRef](#)]
8. Selvarani, D.; Ravi, T. A survey on data and transaction management in mobile databases. *Int. J. Database Manag. Syst.* **2012**, *4*, 1211. [[CrossRef](#)]
9. Ramaj, V.; Januzaj, Y.; Januzaj, Y. Transactions management in mobile database. *Int. J. Comput. Sci. Issues* **2016**, *13*, 39–44.
10. Ahmed, I.; Karvonen, H.; Kumpuniemi, T.; Katz, M. Wireless communications for the hospital of the future: Requirements challenges and solutions. *Int. J. Wirel. Inf. Netw.* **2020**, *27*, 4–17. [[CrossRef](#)]
11. Kumar, P. Fault tolerance approach in mobile distributed systems. *Int. J. Comput. Appl.* **2015**, *975*, 8887.
12. Chandna, A. Mobile database design: A key factor in mobile computing. In Proceedings of the 5th National Conference on Computing for Nation Development, New Delhi, India, 10–11 March 2011; pp. 1–4.

13. Kumar, V. Mobile database recovery. In *Fundamentals of Pervasive Information Management Systems*, 2nd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2013; pp. 219–238.
14. Murthy, D.; Jack, N. Game theoretic modelling of service agent warranty fraud. *J. Oper. Res. Soc.* **2017**, *68*, 1399–1408. [[CrossRef](#)]
15. Bonau, S. A Case for Behavioral Game Theory. *J. Game Theory* **2017**, *6*, 7–14.
16. Quezada, L.; Dong, S. Quantum version of a generalized Monty Hall game and its possible applications to quantum secure communications. *Ann. Physik* **2021**, *533*, 2000427. [[CrossRef](#)]
17. Zabaleta, O.; Barrangú, J.; Arizmendi, C. Quantum game application to spectrum scarcity problems. *Phys. A Stat. Mech. Its Appl.* **2017**, *15*, 455–461. [[CrossRef](#)]
18. Babu, S.; Mohan, U.; Arthanari, T. Modeling cooperation as a quantum game. *Int. Game Theory Rev.* **2020**, *22*, 2040001. [[CrossRef](#)]
19. Schmid, C.; Flitney, A.P.; Wiecek, W.; Kiesel, N.; Weinfurter, H.; Hollenberg, L.C. Experimental implementation of a four-player quantum game. *New J. Phys.* **2010**, *12*, 063031. [[CrossRef](#)]
20. Satria, D.; Park, D.; Jo, M. Recovery for overloaded mobile edge computing. *Future Gener. Comput. Syst.* **2017**, *70*, 138–147. [[CrossRef](#)]
21. Mokhtar, Y.F.; Darwish, S.M.; Madbouly, M.M. An enhanced database recovery model based on game theory for mobile applications. *Adv. Intell. Syst. Comput.* **2021**, *1261*, 16–25.
22. Saraswat, B.K.; Suryavanshi, R.; Yadav, D.S. A comparative study of checkpointing algorithms for distributed systems. *Int. J. Pure Appl. Math.* **2018**, *118*, 1595–1603.
23. Silberschatz, A.; Korth, H.F.; Sudarshan, S. Recovery system. In *Database System Concepts*, 7th ed.; McGraw-Hill Education: New York, NY, USA, 2020; pp. 907–957.
24. George, S.E.; Chen, I.R. Movement-based checkpointing and logging for failure recovery of database applications in mobile environments. *Distrib. Parallel Databases* **2008**, *23*, 189–205. [[CrossRef](#)]
25. Jaggi, P.; Singh, A. Log based recovery with low overhead for large mobile computing systems. *J. Inf. Sci. Eng.* **2013**, *29*, 969–984.
26. Chowdhury, C.; Neogy, S. Checkpointing using mobile agents for mobile computing system. *Int. J. Recent Trends Eng.* **2009**, *1*, 26–29.
27. Mansouri, H.; Badache, N.; Aliouat, M.; Pathan, A. A new efficient checkpointing algorithm for distributed mobile computing. *Control. Eng. Appl. Inform.* **2015**, *17*, 43–54.
28. Mahmoodi, M.; Baraani, A.; Khayyambashi, M. Recovery time improvement in the mobile database systems. In Proceedings of the International Conference on Signal Processing Systems, Singapore, 15–17 May 2009; pp. 688–692.
29. Pamila, M.J.; Thanushkodi, K. Log management support for recovery in mobile computing environment. *arXiv* **2009**, arXiv:0908.0076.
30. Chowhan, R.S. Mobile agent programming paradigm and its application scenarios. *Int. J. Curr. Microbiol. Appl. Sci.* **2018**, *7*, 3269–3273. [[CrossRef](#)]
31. Belghiat, A.; Kerkouche, E.; Chaoui, A.; Beldjehem, M. Mobile agent-based software systems modeling approaches: A comparative study. *J. Comput. Inf. Technol.* **2016**, *24*, 149–163. [[CrossRef](#)]
32. Kekgathetse, M.; Letsholo, K. A survey on database synchronization algorithms for mobile device. *J. Theor. Appl. Inf. Technol.* **2016**, *86*, 1–9.
33. Bhagat, A.R.; Bhagat, V.B. Mobile database review and security aspects. *Int. J. Comput. Sci. Mob. Comput.* **2014**, *3*, 1174–1182.
34. Liang, X.; Yan, Z. A survey on game theoretical methods in human–machine networks. *Future Gener. Comput. Syst.* **2019**, *92*, 674–693. [[CrossRef](#)]
35. Shaha, H.; Kakkada, V.; Patela, R.; Doshi, N. A survey on game theoretic approaches for privacy preservation in data mining and network security. *Procedia Comput. Sci.* **2019**, *155*, 686–691. [[CrossRef](#)]
36. Orbay, B.Z.; Sevgi, L. Game theory and engineering applications. *IEEE Antennas Propag. Mag.* **2014**, *56*, 256–267.
37. Gerasimou, G. Dominance-solvable multi-criteria games with incomplete preferences. *Econ. Theory Bull.* **2019**, *7*, 165–171. [[CrossRef](#)]
38. Trivedi, H.; Mali, M. A Review on Network Simulator and its Installation. *Int. J. Res. Sci. Innov.* **2014**, *1*, 115–116.
39. Gayathri, C.; Vadivel, R. An overview: Basic concept of network simulation tools. *Int. J. Adv. Res. Comput. Commun. Eng.* **2017**, *6*, 19–22.
40. Fang, Y.; Chlamtac, I.; Fei, H. Failure recovery of HLR mobility databases and parameter optimization for PCS networks. *J. Parallel Distrib. Comput.* **2000**, *60*, 431–450. [[CrossRef](#)]