

Kastampolidou, Kalliopi; Papalitsas, Christos; Andronikos, Theodore

## Article

# The distributed Kolkata Paise Restaurant game

Games

## Provided in Cooperation with:

MDPI – Multidisciplinary Digital Publishing Institute, Basel

*Suggested Citation:* Kastampolidou, Kalliopi; Papalitsas, Christos; Andronikos, Theodore (2022) : The distributed Kolkata Paise Restaurant game, Games, ISSN 2073-4336, MDPI, Basel, Vol. 13, Iss. 3, pp. 1-21,  
<https://doi.org/10.3390/g13030033>

This Version is available at:

<https://hdl.handle.net/10419/329944>

## Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

## Terms of use:

*Documents in EconStor may be saved and copied for your personal and scholarly purposes.*

*You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.*

*If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.*



<https://creativecommons.org/licenses/by/4.0/>

## Article

# The Distributed Kolkata Paise Restaurant Game

Kalliopi Kastampolidou <sup>\*,†</sup> , Christos Papalitsas <sup>†</sup>  and Theodore Andronikos <sup>†</sup> 

Department of Informatics, Ionian University, 49100 Corfu, Greece; c14papa@ionio.gr (C.P.); andronikos@ionio.gr (T.A.)

\* Correspondence: c17kast@ionio.gr

† These authors contributed equally to this work.

**Abstract:** The Kolkata Paise Restaurant Problem is a challenging game in which  $n$  agents decide where to have lunch during their break. The game is not trivial because there are exactly  $n$  restaurants, and each restaurant can accommodate only one agent. We study this problem from a new angle and propose a novel strategy that results in greater utilization. Adopting a spatially distributed approach where the restaurants are uniformly distributed in the entire city area makes it possible for every agent to visit multiple restaurants. For each agent, the situation resembles that of the iconic traveling salesman, who must compute an optimal route through  $n$  cities. We rigorously prove probabilistic formulas that confirm the advantages of this policy and the increase in utilization. The derived equations generalize formulas that were previously known in the literature, which can be seen as special cases of our results.

**Keywords:** Kolkata Paise Restaurant Problem; TSP; optimization; metaheuristics; probabilistic analysis; utilization



**Citation:** Kastampolidou, K.; Papalitsas, C.; Andronikos, T. The Distributed Kolkata Paise Restaurant Game. *Games* **2022**, *13*, 33. <https://doi.org/10.3390/g13030033>

Academic Editors: Yezekael Hayel and Ulrich Berger

Received: 14 March 2022

Accepted: 15 April 2022

Published: 20 April 2022

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



**Copyright:** © 2022 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

### 1.1. The Kolkata Paise Restaurant Problem

The Kolkata Paise Restaurant Problem (KPRP for short) is a repeated game that was named after the city Kolkata in India. In KPRP, there are  $n$  cheap restaurants (Paise Restaurants) and  $N$  laborers who choose among these places for their quick lunch break. If the restaurant they go to is crowded, they have to return to work hungry, since they do not have time to visit another restaurant. For a player to win, that is, to eat lunch, only one player should go to each restaurant. If two or more agents visit the same restaurant at the same time, one agent is chosen randomly, and only this agent is served. The player who eats has a payoff equal to 1, whereas all others who also chose this restaurant have a payoff equal to 0. Each agent prefers to go to an unoccupied restaurant rather than visit a restaurant where there are other agents as well. This realization, in turn, implies that the pure strategy Nash equilibria of the stage game are Pareto efficient. Consequently, there are exactly  $n!$  pure strategy Nash equilibria for the stage game. This, combined with the rationality of the players, leads to the conclusion that it is possible to sustain a pure strategy Nash equilibrium of the stage game as a sub-game perfect equilibrium of the KPRP.

### 1.2. The Traveling Salesman Problem

The Traveling Salesman Problem (TSP) is a famous optimization problem described as follows: a salesman has to visit all the nearby cities, starting from a specific city to which the salesman must return. The only constraints are that the salesman must start and finish at the same city and visit each other city only once. The visiting order is to be determined by the salesman each time the problem arises. The cities are connected through a railway or a road network, and the time it takes to move from one city to an adjacent one is modeled by the cost assigned to the corresponding edge of the graph. The salesman has just one purpose, and that is to visit all the cities with the minimum possible travel

cost. In this problem, the optimum solution is the fastest, shortest and cheapest solution. The TSP is easily expressed as a mathematical problem that typically assumes the form of a graph, where the nodes represent the cities that the salesman has to visit. The TSP is an NP-hard problem, and the results of the practical, heuristic solutions are not always optimal, but approximate [1]. The simplest “naive” solution to this problem is to try all possibilities and explore all paths, but the cost in time and complexity is so huge that it is practically impossible. To overcome that, when solving a TSP, the pragmatic focus is a near-optimal route instead of always the best route.

### 1.3. Related Work

#### 1.3.1. The Kolkata Paise Restaurant Problem

The Kolkata Paise Restaurant Problem (KPRP) was initially introduced in an earlier form in 2007 [2]. Its current formulation appeared in 2009 in [3,4]. Subsequently, many creative ideas from different lines of thought have been published, and even a quantum version of the game has arisen. In [3], the importance of diversity is emphasized, while herd behavior is penalized. Some of the strategies developed for the KPRP are discussed in [5], which also discusses problems where these strategies can be successfully applied. Ghosh et al. presented a dictator’s, or, as they call it, a social planner’s solution. In this solution, the agents form a queue, and the planner assigns each of them to a ranked restaurant depending on the queue of the first evening. The following evening, the agents go to the next-ranked restaurant, and the last in the queue goes to the first-ranked restaurant. This solution is called the fair social norm. In real life, each agent decides in parallel or democratically every evening, so this solution may be considered somewhat unrealistic. However, the parallel decision or democratic decision strategy is not as efficient as the dictated one, with the latter leading to one of the best solutions to this problem. Banerjee et al. in [6] offered a generalization of the problem in such a way that the cyclically fair norm is sustained. Each strategy is viewed as a sub-game of perfect equilibrium of the KPRP. In 2013, Ghosh et al. published an article about stochastic optimization strategies in the KPRP [7]. There, they point out that a stochastic crowd-avoiding strategy results in efficient utilization in the KPRP. Reinforcement learning was first introduced to the KPRP in [8], together with six revision protocols aiming at efficient resource utilization. These protocols combine local information with reinforcement learning. Each revision protocol has two variants, depending on whether or not customers who were once served by a restaurant remain loyal to that restaurant in all subsequent periods. Some of these protocols were experimentally tested and shown to improve the utilization rate. Another generalization was introduced by Yang et al. in [9], aiming at dynamic markets. They studied what happens when agents can either divert to another district or stay in the current one. Each agent may replace another agent with no prior knowledge of the game, following a Poisson distribution. Agarwal et al. in [10] showed that the KPRP could be reduced to a Majority Game. In the latter, capacity is not restricted, and agents aim at choosing with the herd. If many agents choose the same option, the utility decreases (see also [11,12]). Abergel et al. in [13] applied the KPRP to hospitals and beds. Among the local hospitals, the local patients choose those with the best ranking and compete with the other patients. If the patients are not treated in time, it is a clear case of social waste of service for the rest of the hospitals. Park et al. in [14] introduced the KPRP to the context of the Internet of Things (IoT) and IoT devices. They used a KPRP approach to develop a scheme for these devices because it allowed them to model situations where multiple resources are shared among multiple users, each with individual preferences. In [15], Sinha et al. proposed a phase transition behavior, where if two or more agents visit the same restaurant, one is randomly chosen to eat.

A significant trend, quite evident in the last two decades, is to enhance classical games using unconventional means. The most prominent direction is to cast a classical game in a quantum setting. Since the pioneering works of Meyer [16] and Eisert et al. [17], quantum versions for a plethora of well known classical games have been studied in the

literature. Starting from the most famous of all games, the Prisoners' Dilemma [17–20], many researchers have sought to achieve better solutions by employing quantumness (see the recent [20–22] and references therein), or other tools, such as automata [23]. It is not surprising that unconventional approaches to classical games are undertaken because they promise clear advantages over classical ones, or additional insights over traditional interpretations as in [24]. Another line of research is to turn to biological systems for inspiration. Most game situations can easily find analogs in biological and bio-inspired processes (see also [25]). A quantum version of the KPRP was proposed in [26], where the agents cannot communicate with each other and have to choose among  $m$  choices, but an agent wins by making a unique choice. Higher payoffs than the classical version were observed due to shared entanglement and the use of quantum operators. In Sharif's [27] review, quantum protocols for quantum games were introduced, including a protocol for a three-player quantum version of the KPRP. In [28], the authors studied the effect of quantum decoherence on a three-player quantum KPRP using tripartite entangled qutrit states. They observed that, in the case of maximum decoherence, the influence of the amplitude-damping channel dominates over depolarizing and flipping channels.

### 1.3.2. The Traveling Salesman Problem

The first appearance of the term “Traveling Salesman Problem” probably occurred between 1931 and 1932, and it was first studied by Karl Menger during the 1930s at Harvard and Vienna. The core of the TSP problem, however, was first mentioned over a century before in an 1832 German book [29]. The mathematical formulation was introduced by Sir William Rowan Hamilton and Thomas Kirkman.

Research efforts on the TSP and closely related problems include Ascheuer et al. [30], who addressed the asymmetric TSP-TW using more than three alternative integer programming formulations and more than ten neighborhood structures. Gutin and Punnen [31] studied the effect of sorting-based initialization procedures. The authors claimed that understanding the algorithmic behavior was the best way to find solutions since this would help in determining the best one out of those available.

The difficulty of tackling the TSP motivated researchers to explore other avenues. One such notable and particularly promising approach is based on metaheuristics. A metaheuristic is a high-level heuristic that is designed to recognize, build, or select a lower-level heuristic (such as a local search algorithm) that can provide a fairly good solution, particularly with missing or incomplete information or with limited computing capacity [32]. The term “metaheuristics” was coined by Glover. Metaheuristics can be used for a wide range of problems. It must be noted of course that metaheuristic procedures, in contrast to exact methods, do not guarantee a globally optimal solution [33]. Papalitsas et al. [34] designed a metaheuristic based on VNS for the TSP, with emphasis on time windows. Another quantum-inspired method, based on the original General Variable Neighborhood Search (GVNS), was proposed to solve the standard TSP [35]. This quantum-inspired procedure was also applied successfully to the solution of real-life problems that can be modeled as TSP instances [36]. A quantum-inspired procedure for solving the TSP with time windows was also presented in [37]. More recently, ref. [38] applied a quantum-inspired metaheuristic for tackling the practical problem of garbage collection with time windows that produced particularly promising experimental results, as further comparative analysis demonstrated in [39]. A thorough statistical and computational analysis of asymmetric, symmetric, and national TSP benchmarks from the well known TSPLIB benchmark library was conducted in [40]. Very recently, Papalitsas et al. expressed the TSPTW in the Quadratic Unconstrained Binary Optimization (QUBO for short) framework [41]. The QUBO formulation enables TSPTW to run on a quantum annealer and is a critical step towards the ultimate goal of running the TSPTW using pure quantum optimization methods. Stochastic optimization can be implemented through several metaheuristic processes. The obtained solution depends on the set of random variables [32]. Metaheuristic processes may find successful solutions with less computational effort than

accurate algorithms, iterative methods, or basic heuristic procedures by looking for a wide variety of feasible solutions [33]. Hence, metaheuristic procedures are extremely useful and practical approaches for optimization because they can guarantee good solutions in a small amount of time. For example, a problem instance with thousands of nodes can be run for 30–40 s and produce a solution with a 3–5% deviation from the optimal. In view of the small amount of time they require and of the good quality of the solution they produce, we advocate their functional use in the Distributed Kolkata Paise Restaurant game.

**Contribution.** In this paper, we approach the Kolkata Paise Restaurant Problem from an entirely new perspective. This provides the opportunity for an entirely new setting and the adoption of a novel approach that leads to a new and more efficient strategy and, ultimately, to greater utilization of the restaurants. For the first time, to the best of our knowledge, we focus on the spatial setting of the game, and we propose a more realistic and plausible topological layout for the restaurants. We perceive the restaurants to be uniformly distributed over the entire city area. This has profound ramifications on the topological layout of the game: the restaurants now get closer and, as their number  $n$  increases, the distances between nearby restaurants decrease. Due to the distribution of the restaurants, the resulting version of the game is aptly named the Distributed Kolkata Paise Restaurant Game. Thus, it is now realistic to assume that every agent has a second, a third, and maybe even a fourth chance. The agent is no longer a single destination and back traveler. The agent now resembles the iconic traveling salesman, who must pass through a network of cities and come back to the starting point, all while following the optimal route. This entirely new setting is formalized and rigorously analyzed via probabilistic tools. We derive general formulas that mathematically confirm the advantages of this policy and the increase in utilization. The proposed scheme demonstrably achieves utilization ranging from 0.85 to 0.95, and even beyond, from the first day. The steady-state utilization, to which the game rapidly converges, is, as expected, 1.0. The equations we derive generalize formulas that were previously presented in the literature, showing that the latter are special cases of our results.

#### 1.4. Organization

The structure of this paper is as follows. In Section 1, we provide a comprehensive description of the KPRP and the TSP. The rigorous formulations of the KPRP and the TSP are presented in Section 2. In Section 3, we give a thorough explanation and presentation of the distributed version of the game. We analyze mathematically the topological situation regarding the restaurants in Section 4. We formally prove the main results of the paper in Section 5. Finally, in Section 6, we summarize our results and discuss future extensions of this work.

## 2. Background

### 2.1. Formulation of the Standard Kolkata Paise Restaurant Problem

In its most usual formulation, the Kolkata Paise Restaurant Problem is a repeated game with infinite rounds. There is a set of players, typically called *agents* or *customers*, that is denoted by  $A = \{a_1, \dots, a_n\}$ , a set of restaurants that is denoted by  $R = \{r_1, \dots, r_n\}$ , and a utility vector  $u = (u_1, \dots, u_n) \in \mathbb{R}^n$  that is associated with the restaurants and is common to every agent. On any given day, all agents decide to go to one of the  $n$  restaurants for lunch. If just one agent arrives at a specific restaurant, she has lunch and is happy. If, however, two or more agents choose the same restaurant for lunch, just one of them eats. The one who eats is chosen randomly. So, in such a case, all but one are unhappy. The utility of a happy agent is 1 and 0 otherwise. In Chakrabarti et al. [3], the KPRP is modeled as a general one-shot restaurant game, where the set of agents is considered to be finite, and the utilities are ranked as follows:  $0 < u_n \leq \dots \leq u_2 \leq u_1$ . The set of agents  $A$  and the ranking of the utilities can be used to define the game. The latter can be represented as  $G(u) = (A, S, \Pi)$ , where  $A$  is the set of agents,  $S$  is the set of strategies available to all agents, and  $\Pi = (\Pi_1, \dots, \Pi_n)$  stands for the payoff vector. If the  $i$ th agent



$a_i$  decides to go to the  $j$ th restaurant  $r_j$ , then the corresponding strategy is  $s_i = j$ . Every day, each agent decides which of the  $n$  restaurants she visits. Given any strategy combination  $s = (s_1, \dots, s_n) \in S^n$ , the associated payoff vector is defined as  $\Pi(s) = (\Pi_1(s), \dots, \Pi_n(s))$ , where the payoff  $\Pi_i(s)$  of player  $a_i$  is  $\frac{u_{s_i}}{N_i(s)}$ , and  $N_i(s)$  is the total number of players who have made the same choice. The strategy combination is the restaurants the agents choose to eat at, and their payoff depends on their decision and the number of other agents who have made the same choice. In the literature, a game such as KPRP, where there are potentially infinite rounds and the same stage game is played in each round, is referred to as a *supergame* [42].

## 2.2. Formulation of the TSP

The TSP is an NP-hard problem of great significance to different fields, e.g., operational research and theoretical computer science. Usually, the TSP is represented by a graph. The fact that the TSP is NP-hard implies that there is no known polynomial-time algorithm for finding an optimal solution [43]. There are two types of models for the TSP: *symmetric* and *asymmetric*. The former is represented by a complete undirected graph  $G = (V, E)$  and the latter by a complete directed graph  $G = (V, A)$ . Assuming that  $n$  denotes the number of cities (nodes),  $V = \{1, 2, 3, \dots, n\}$  is the set of vertices,  $E = \{(i, j) : i, j \in V, \text{ where } i < j\}$  is the set of edges, and  $A = \{(i, j) : i, j \in V, \text{ where } i \neq j\}$  is the set of arcs. A cost matrix  $C = [c_{i,j}]$ , which satisfies the triangle inequality  $c_{i,j} \leq c_{i,k} + c_{k,j}$  for every  $i, j, k$ , is defined for each edge or arc. If  $c_{i,j}$  is equal to  $c_{j,i}$ , the TSP is *symmetric* (sTSP), otherwise it is *asymmetric* (aTSP). In particular, this is the case for problems where the vertices are points  $P_i = (X_i, Y_i)$  of the Euclidean plane, and  $c_{i,j} = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2}$  is the Euclidean distance. The triangle inequality holds if the quantity  $c_{i,j}$  represents the length of the shortest path from  $i$  to  $j$  in the graph  $G$  [44]. In the case of the symmetric TSP, the number of all possible routes covering all cities and corresponding to all feasible solutions is given by  $\frac{(n-1)!}{2}$  (recall that the number of cities is  $n$ ). The cost of the route is the sum of the costs of the edges followed.

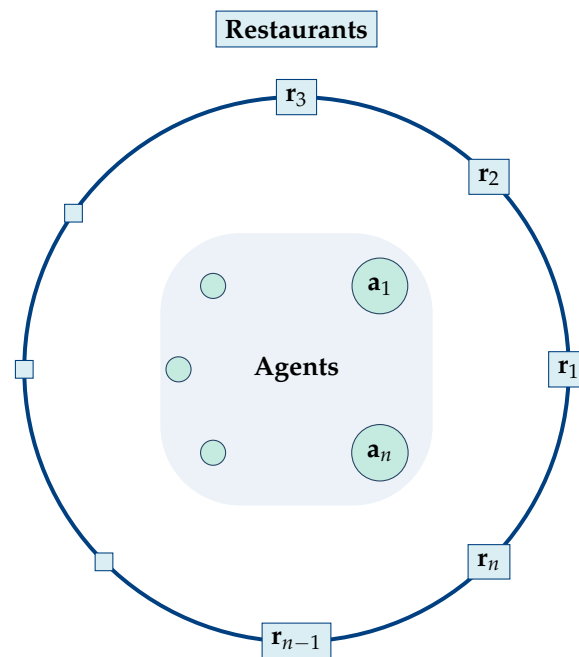
## 3. Formulation of the DKPRG

The Kolkata Paise Restaurant Problem (KPRP) involves multiple players ( $n$ ), each having multiple choices ( $N$ ). In its most general form, it is possible that  $n \neq N$ . In this paper, we follow the standard approach that the number of agents is equal to the number of restaurants, i.e.,  $n = N$ . We advocate a spatially distributed and, in our view, more realistic version of the KPRP by taking into account the topology of the restaurants and by allowing the agents to begin their routes from different starting points. We call our version the *Distributed Kolkata Paise Restaurant Game*, or DKPRG from now on. In the original formulation of the KPRP, one may point out the following underlying assumptions.

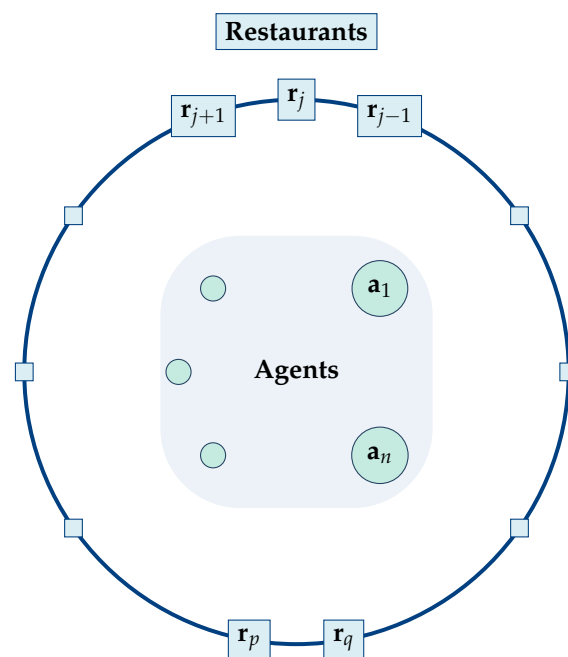
- (A1) The agents start from the same location.
- (A2) All restaurants are close enough to the point of origin of every customer that each customer can, in principle, go to any restaurant, eat there, and return to work on time.
- (A3) Every restaurant is sufficiently far away from every other restaurant to make prohibitive the possibility of any customer visiting a second restaurant if their first choice proved fruitless.

In the two-dimensional setting of Kolkata, the above assumptions taken together imply the topology depicted in Figure 1. There, the agents are concentrated within a very narrow region, which can be viewed as the center of a conceptual “circle”. The restaurants are located on this “circle” and, since *no two of them are allowed to be close*, they form a “regular polygon”. This precludes a situation such as the one shown in Figure 2. The spatial layout depicted there is strictly forbidden. The proximity of two, three, or more restaurants contradicts the impossibility of a second chance. In the standard KPRP, no agent is allowed a second chance. We write “circle” and “regular polygon” inside quotation marks because we are not dealing with a perfect geometric circle or a perfect regular polygon, but with two-

dimensional approximations resembling the aforementioned symmetric shapes. Clearly, this is a very special topological layout, one that is highly unlikely to be observed in practice. There is no compelling reason for the restaurants to exhibit this regularity or the agents to be confined to approximately the same location. On the contrary, it would seem far more reasonable to assume that the restaurants and perhaps the agents are uniformly distributed. Moreover, the usual assumption that the preference ranking of the restaurants is common to all customers seems a bit too special and probably too restrictive.



**Figure 1.** In the original formulation of the KPRP, the above topology is assumed implicitly.



**Figure 2.** In the original formulation of the KPRP, the above spatial layout is strictly forbidden.

With that motivation in mind, we propose to abolish all these assumptions. The resulting game is spatially distributed in terms of restaurants and is called the Distributed Kolkata Paise Restaurant Game (DKPRG). In our setting, each customer may have a unique

starting point that is, in general, different from the starting locations of the other customers. The starting locations can either be concentrated in a small region of the Kolkata city area, precisely like the standard KPRP, or they may be assumed to follow a random distribution. The fundamental difference from prior approaches is that now the restaurants are viewed as being *uniformly distributed* over the city of Kolkata. This uniform randomness in the placement of the restaurants implies that there must be restaurants near each other. This conclusion becomes inescapable, particularly in the case where the number of restaurants is large ( $n \rightarrow \infty$ ). As we show, the *expected distance* between “adjacent” restaurants is relatively short and decreases as the number  $n$  of restaurants increases. This leads to a *personalized* situation for each individual agent: each agent is faced with a *personalized* Traveling Salesman Problem. To every agent corresponds an individual graph. This graph has  $n + 1$  nodes, which are the locations of the  $n$  restaurants plus the location of the starting point of the customer. The costs assigned to the edges are also *personalized*; each agent combines an objective factor, namely, the spatial distances between the restaurants, with a subjective factor, her personal preferences. In the DKPRG, we forego the common preference restriction, and we let every customer have a distinct preference. Let us clarify, however, that being served, even at the least preferable restaurant, is more desirable than not being served at all. This, in turn, leads to a possibly unique ordering of the restaurants from the most preferable to the least preferable for *each agent*.

Hence, every agent is faced with a distinctive network topology that is the combined result of the inherent randomness of the spatial locations and the subjectiveness of the preferences. The topology of the restaurants has a further critical implication: a customer whose first choice is a particular restaurant now has, with very high probability, the opportunity to visit a second, a third, or even a fourth restaurant in the same area. For each agent, the time cost is dominated by the time taken to visit the first restaurant; the trip to other nearby restaurants in the same region incurs a negligible time cost due to their spatial proximity. The customer has a second (or even a third) chance to be served during the lunch break. Thus, an efficient, if not optimal, method for every customer to make well informed decisions regarding the first, second, third, etc., choice is to solve the Traveling Salesman Problem for her *personalized* graph. Although the TSP is an NP-hard problem, near-optimal solutions of practical value can be achieved in a very short time by employing metaheuristics. Starting from this perspective, we propose a distributed strategy that leads to an efficient global solution. All agents use a common high-level strategy that is fine-tuned according to their individual preferences. To enhance clarity, we explicitly state below the game settings that characterize the DKPRG variant.

- (1) The Distributed Kolkata Paise Restaurant Game is an infinitely repeated game.
- (2) The set of agents is denoted by  $A = \{a_1, \dots, a_n\}$ . The  $n$  agents (also referred to as *customers*) may have different starting locations. The set of restaurants is denoted by  $R = \{r_1, \dots, r_n\}$ . The  $n$  restaurants are uniformly distributed within the city area. All agents know the locations of the restaurants, but they need not know the starting locations of the other agents.
- (3) To each agent  $a \in A$  corresponds a distinct *personal preference ordering*  $P_a = (r_{j_1}, r_{j_2}, \dots, r_{j_n})$ , such that restaurant  $r_{j_1}$  is their first preference,  $r_{j_2}$  is their second preference, and so on, with  $r_{j_n}$  being the least preferable.
- (4) Each restaurant can accommodate only one customer at a time; if two or more customers arrive at a restaurant, only one can be served. The one to be served is chosen randomly.
- (5) Each agent  $a \in A$  perceives a *personalized graph*  $G_a = (V_a, E_a)$ .  $G_a$  is a complete undirected graph having  $n + 1$  vertices  $v_0, v_1, \dots, v_n$ , where  $v_0$  is the starting location of  $a$  and  $v_j$  is the location of restaurant  $j$ ,  $1 \leq j \leq n$ . The graph  $G_a$  is complemented with the (symmetric) *cost matrix*  $C_a$  that assigns to each edge  $(u, v)$  a cost  $c_{u,v}$ .
- (6) Each agent  $a \in A$  solves the corresponding TSP using an efficient metaheuristic that outputs a near-optimal *tour*  $T_a = (l_0, l_1, \dots, l_n, l_{n+1})$ , where  $l_0 = l_{n+1}$  is the starting point of  $a$  and  $l_k$ ,  $1 \leq k \leq n$ , is the index of the restaurant in the  $k$ th position of the tour.



- (7) Endowed with their individual tour  $T_a$ , all customers follow a *common* strategy. They travel from their starting location  $l_0$  to the restaurant  $r_{l_1}$ . If they are served, then they conclude their route successfully. If not, they proceed to  $r_{l_2}$ . If their attempt at eating lunch there also fails, then they proceed to  $r_{l_3}$ , and so on.
- (8) **The Revision Strategy.** The agents operate independently, and no communication takes place between any two of them. Therefore, each customer is completely unaware of the routes of the other customers. Every evening they revise their strategy, taking into account only what happened during the current day. If they were served at a specific restaurant this day, then tomorrow they go straight to the *same* restaurant. This applies even if this restaurant is not the first destination of their tour. Those who failed to eat lunch only know *which restaurants were left vacant*. The unserved agents construct and solve their new personalized TSP, this time using as vertices only the *vacant restaurants* (plus of course their starting location).

#### 4. Topological Considerations

We begin this section by fixing the notation and giving some definitions to clarify the most important concepts of our exposition.

**Definition 1.** The one-shot DKPRG takes place every day. We use the parameter  $t = 1, 2, \dots$ , to designate the day under consideration.

- To every agent  $a \in A$  corresponds a personalized network  $G_a = (V_a, E_a)$  with cost matrix  $C_a$ . Agent  $a$  follows the tour  $T_a = (l_0, l_1, \dots, l_n, l_{n+1})$ , which is the solution to the personalized TSP.
- The quality and efficiency of the strategy are measured by the utilization ratio  $f$ . This is the fraction of agents being served in a day, or, equivalently, the fraction of restaurants serving customers in a day.

An agent  $a$  who has opted to follow tour  $T_a = (l_0, l_1, \dots, l_n, l_{n+1})$  initially tries to eat lunch at restaurant  $r_{l_1}$ . If she succeeds, she eats and returns to her starting point. If not, she visits the next restaurant of the tour, i.e.,  $r_{l_2}$ . If she eats lunch there, she goes back to work. This goes on until either she is served or runs out of time, in which case she must interrupt the tour and return to work. If the time constraints allow her to pass through the *first*  $m$  restaurants in the tour, in the worst-case scenario of  $m - 1$  consecutive failures, then we say that  $T_a$  is an *m-stop* tour. In the rest of this work, we study the case where  $m \geq 2$  because the case where  $m = 1$  reduces to the standard treatment of the KPRP, which has already been analyzed extensively in the literature.

**Definition 2.** The tour  $T_a = (l_0, l_1, \dots, l_n, l_{n+1})$  is an *m-stop* tour,  $m \geq 2$ , if, in the worst case, agent  $a$  can visit restaurants  $l_1, l_2, \dots, l_m$  in this order without violating the time constraints. In such a tour,  $l_1$  is the first stop,  $l_2$  is the second stop, and so on, with  $l_m$  being the final *m*th stop. If  $\forall a \in A$ ,  $T_a$  is an *m-stop* tour, and the resulting game is the *m-stop* DKPRG.

Let us now explore the spatial ramifications of our assumption that the restaurants are uniformly distributed within the overall city area.

**Definition 3.** Given a region  $B$  on the plane, a random variable  $L$  that takes values in  $B$  has uniform distribution on  $B$ , if for any subregion  $C$  the following holds:

$$P(L \in C) = \frac{\text{area}(C)}{\text{area}(B)}, \quad C \subset B, \quad (1)$$

where  $P$  denotes the probability.

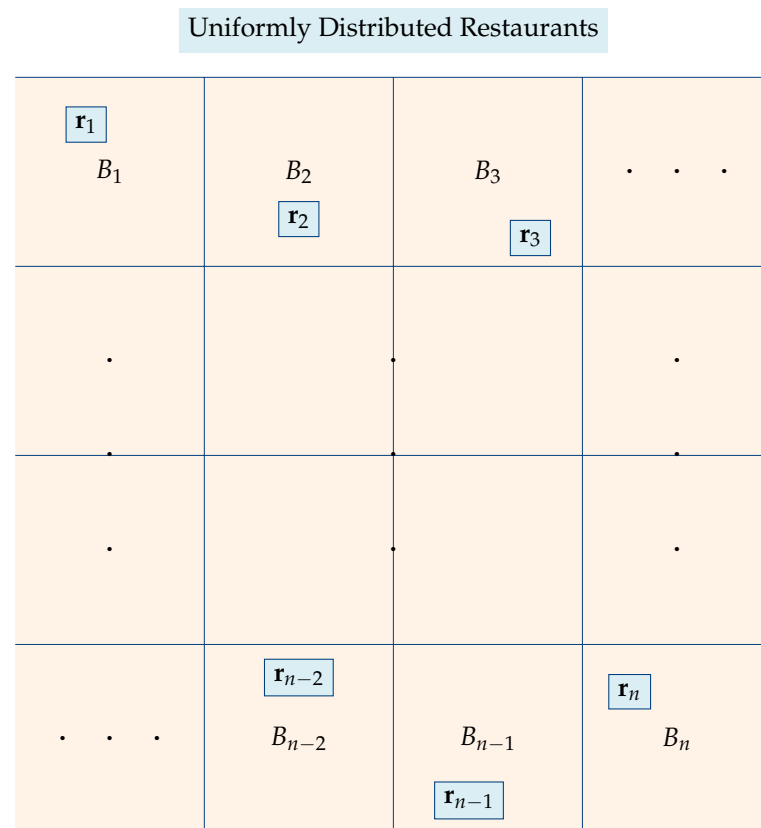
The above definition is adapted from [45]. For a more general and sophisticated definition in terms of measures, we refer the interested reader to [46].

In order to enhance the readability of this work, we have placed the proofs of all the results that follow in the Appendix A.

**Proposition 1.** *Assuming that the  $n$  restaurants are uniformly distributed over the whole city area, then, if the city area is partitioned into  $n$  regions of equal area, the expected number  $\overline{N_p}$  of restaurants in each region is exactly 1.*

$$\overline{N_p} = 1, 1 \leq p \leq n. \quad (2)$$

This topological layout of the restaurants is shown in Figure 3. The regions are drawn as squares, but this is just for convenience and to facilitate their graphic depiction. For very large values of  $n$ , partitioning a city into very small identical squares is a good approximation, as we know from the field of image representation. We emphasize that the use of squares does not lead to any loss of generality. From a theoretical point of view, the standard metric topology of  $\mathbb{R}^2$  is the same irrespective of whether the topological basis consists of squares or circular balls or diamond shape regions (a thorough analysis of this fact can be found in [47]).



**Figure 3.** If the restaurants are uniformly distributed, then the *expected number* of restaurants in each region is 1.

Let us observe that there is a meaningful notion of *distance* between *any two points* in the city area. In reality, this can be the geographical distance between any two locations. Consider, for instance, two points  $x$  and  $y$  with spatial coordinates  $(x_1, x_2)$  and  $(y_1, y_2)$ , respectively. A typical manifestation of the notion of distance is the *Euclidean* distance:  $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$  between  $x$  and  $y$ . We take for granted the existence of a distance function defined on every pair of points  $(x, y)$ , denoted by  $d(x, y)$ .

**Definition 4.**

- The distance between two regions  $B_p$  and  $B_q$  is defined as

$$d(B_p, B_q) = \inf\{d(x, y) : x \in B_p \text{ and } y \in B_q\}. \quad (3)$$

- Two regions  $B_p$  and  $B_q$  are adjacent if

$$d(B_p, B_q) = 0. \quad (4)$$

- The diameter (see also [47]) of a region  $B_p, 1 \leq p \leq n$ , is defined as

$$\text{diam} B_p = \sup\{d(x, y) : x, y \in B_p\}. \quad (5)$$

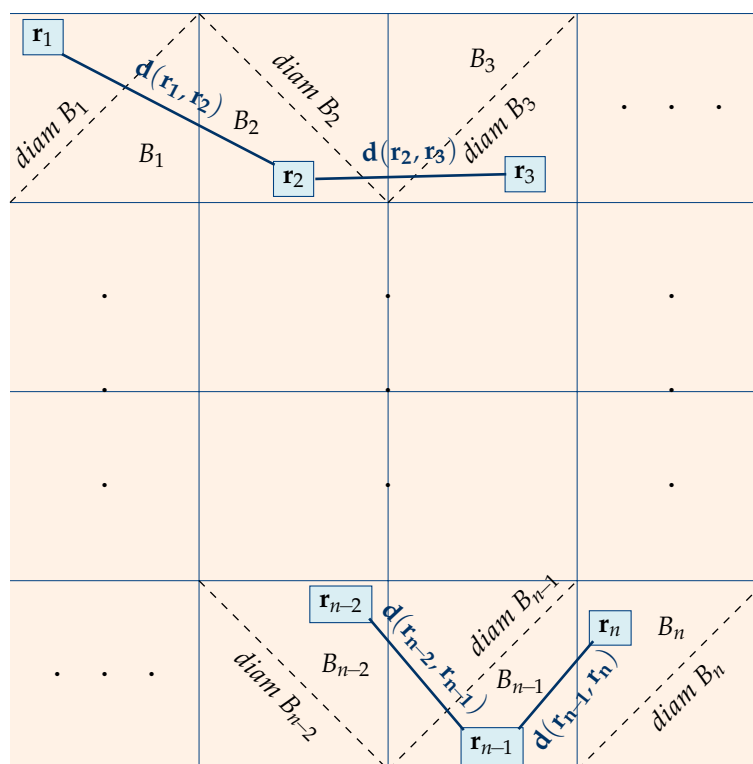
**Proposition 2.** Let the  $n$  restaurants be uniformly distributed over the city area and assume that the whole area is partitioned into  $n$  regions of equal area. If  $r_p$  and  $r_q$  are located at adjacent regions  $B_p$  and  $B_q, 1 \leq p \neq q \leq n$ , then

$$d(r_p, r_q) \leq \text{diam} B_p + \text{diam} B_q, \quad 1 \leq p \neq q \leq n. \quad (6)$$

As shown in Figure 4 the expected distance between restaurants which lie in adjacent regions becomes shorter, as the number of restaurants increases. The above upper bound can be simplified if we assume that all regions have the same geometric shape. This implies that  $\text{diam} B_1 = \dots = \text{diam} B_n = D$ , in which case inequality (6) becomes:

$$d(r_p, r_q) \leq 2D, \quad 1 \leq p \neq q \leq n. \quad (7)$$

In the special case where the regions are squares,  $D \propto \sqrt{\frac{2}{n}}$ .



**Figure 4.** The expected distances between restaurants located in adjacent regions.

The diameter of the regions decreases as  $n$  increases, and in the special case shown in the previous figures, the diameter decreases in proportion to  $\frac{1}{\sqrt{n}}$ . This means that *adjacent restaurants are very close to each other* as  $n \rightarrow \infty$ .

## 5. Mathematical Analysis of the Utilization

This section is devoted to the analytic estimation of the evolution of the game parameters and the daily utilization of the proposed strategy scheme.

### Definition 5.

- The expected number of agents who had lunch during day 1 is denoted by  $\overline{A}_1^s$  and the expected number of those who failed to do so is denoted by  $\overline{A}_1^u$ .
- The expected number of agents who ate for the first time during day  $t, t = 2, 3, \dots$ , is denoted by  $\overline{A}_t^s$ . The expected number of agents who failed to eat during day  $t, t = 2, 3, \dots$ , is denoted by  $\overline{A}_t^u$ .
- Symmetrically, the expected number of restaurants that served lunch during day 1 is denoted by  $\overline{R}_1^s$ , the expected number of restaurants that did not is denoted by  $\overline{R}_1^u$ , the expected number of restaurants that served a customer for the first time during day  $t, t = 2, 3, \dots$ , is denoted by  $\overline{R}_t^s$ , and the expected number of those who failed to do so is denoted by  $\overline{R}_t^u$ .
- The vacancy probability of day 1 is the probability that a restaurant did not accommodate any customer during day 1 and is designated by  $VP_1$ . The vacancy probability of day  $t, t = 2, 3, \dots$ , denoted by  $VP_t$ , is the probability that a restaurant that has not served any customer before day  $t$  did not serve a customer during day  $t$  either.
- In the  $m$ -stop DKPRG, the agents who actively play the game at the beginning of day  $t$ , seeking a restaurant at which to eat lunch, are called active players, and their expected number is denoted by  $n_t$ .
- The expected utilization of day  $t, t = 1, 2, \dots$ , designated by  $\overline{f}_t$ , is the fraction of the expected number of agents who were served during day  $t$ . The steady-state utilization is defined as  $f_\infty = \sup\{f_t : t \in \mathbb{N}\}$ .

The following analysis is based on the premise that *all  $n!$  tours are equiprobable*. We shall refer to it as the *equiprobability of tours assumption* (EPT for short). Given the discussion in the previous sections, this premise is well justified. An immediate consequence of the EPT assumption is the *equiprobability of each restaurant* appearing in any position. For easy reference, these facts are collected in the next Proposition 3, for which the proof is trivial and thus omitted.

**Proposition 3.** Assuming the equiprobability of tours, the following hold.

$$\forall a \in A, \forall r \in R, \forall k, 1 \leq k \leq n, P(r \text{ is in position } k \text{ of } T_a) = \frac{1}{n} \quad (8)$$

$$\forall a \in A, \forall r \in R, \forall k, 1 \leq k \leq n, P(r \text{ not in position } k \text{ of } T_a) = \frac{n-1}{n} \quad (9)$$

The above can be generalized to handle the case of a restaurant  $r$  appearing in one of  $w$  distinct positions  $k_1, k_2, \dots, k_w$ , where  $1 < w \leq n$ .

$$\forall a \in A, \forall r \in R, P(r \text{ is in one of positions } k_1, \dots, k_w \text{ of } T_a) = \frac{w}{n} \quad (10)$$

$$\forall a \in A, \forall r \in R, P(r \text{ not in any of positions } k_1, \dots, k_w \text{ of } T_a) = \frac{n-w}{n} \quad (11)$$

Since the probability that restaurant  $r \in R$  is in the  $k$ th position of the tour of agent  $a$  is  $\frac{1}{n}$ , the probability of the complementary event is  $\frac{n-1}{n}$ . If we deem as “success” the case where  $r$  is indeed in the  $k$ th position of  $T_a$  and as “failure” the case where  $r$  is not, then this situation is a typical example of a *Bernoulli trial*, having probability of success  $\frac{1}{n}$  (also

referred to as *parameter*, see [48]) and probability of failure  $\frac{n-1}{n}$ . Analogously, the probability that restaurant  $r \in R$  appears in one of  $w, 1 < w \leq n$ , distinct positions of the tour of agent  $a$  is  $\frac{w}{n}$ . The probability of the complementary event is  $\frac{n-w}{n}$ . Again, this is a Bernoulli trial, this time with parameter  $\frac{w}{n}$ . The fact that the  $n$  agents calculate their tours *independently* implies that  $n$  independent Bernoulli trials take place simultaneously, all with the same success and failure probabilities. This situation is described by the *binomial* distribution with parameters  $(n, p)$  (see [45,48] for details), denoted by  $\text{Bin}(n, p)$ , where  $p = \frac{1}{n}$  in the simple case of one position and  $p = \frac{w}{n}$  in the general case of  $w$  positions. Using well known formulas from probability textbooks, we may assert the following Proposition 4, the proof of which is also trivial.

**Proposition 4.** *Given a restaurant  $r$ , the probability of appearing in a specified position  $k$  exactly  $l$  times in total in  $n$  tours is given by*

$$\forall r \in R, \forall k, 1 \leq k \leq n, P(r \text{ appears } l \text{ times in position } k) = \binom{n}{l} \left(\frac{1}{n}\right)^l \left(\frac{n-1}{n}\right)^{n-l}. \quad (12)$$

*In the special case, where  $r$  never appears in position  $k$ , the above formula becomes:*

$$P(r \text{ never appears in position } k) = \binom{n}{0} \left(\frac{1}{n}\right)^0 \left(\frac{n-1}{n}\right)^n = \left(\frac{n-1}{n}\right)^n. \quad (13)$$

*More generally, the probability that restaurant  $r$  appears exactly  $l$  times in total in one of the  $w$  distinct positions  $k_1, \dots, k_w, 1 < w \leq n$ , is given by*

$$\forall r \in R, P(r \text{ appears } l \text{ times in positions } k_1, \dots, k_w) = \binom{n}{l} \left(\frac{w}{n}\right)^l \left(\frac{n-w}{n}\right)^{n-l}. \quad (14)$$

*The probability that  $r$  never appears in the designated positions is*

$$P(r \text{ never appears in positions } k_1, \dots, k_w) = \binom{n}{0} \left(\frac{w}{n}\right)^0 \left(\frac{n-w}{n}\right)^n = \left(\frac{n-w}{n}\right)^n. \quad (15)$$

According to the strategy scheme employed in the  $m$ -stop DKPRG, at the start of the second (third, etc.) day, the satisfied customers always go straight to the restaurant that *eventually* served them during the previous day. This strategy is followed by all agents, which guarantees that the customers who were satisfied the previous day remain satisfied today. Effectively, this implies that the satisfied agents have “won” the game and from now on they do not need to solve their personalized TSP. The game is played competitively by the unsatisfied agents of the previous day. We assume that they are aware of the unoccupied restaurants and, therefore, each one of them once again solves their personalized TSP to compute their tour. Of course, today the network of restaurants consists only of the *unoccupied* restaurants. The one-shot  $m$ -stop DKPRG of today is different from the one-shot game of the previous day in a critical factor: the number of “actively competing” players is significantly smaller. By the nature of the game, the number of active players at the *beginning* of stop 1 of the present day is equal to the number of unsatisfied customers at the end of the previous day. The expected number of active players varies with each passing day according to Theorem 1.

**Theorem 1.** *The daily progression of the  $m$ -DKPRG is described by the following formulas, where  $t$  stands for the day in question.*

$$VP_t = \left( \frac{n_t - m}{n_t} \right)^{n_t}, \quad n_t \geq m, \quad t = 1, 2, \dots, \quad (16)$$

$$\overline{R}_t^v = n_t \left( \frac{n_t - m}{n_t} \right)^{n_t}, \quad n_t \geq m, \quad t = 1, 2, \dots, \quad (17)$$

$$\overline{R}_t^r = n_t \left( 1 - \left( \frac{n_t - m}{n_t} \right)^{n_t} \right), \quad n_t \geq m, \quad t = 1, 2, \dots, \quad (18)$$

$$\overline{A}_t^u = n_t \left( \frac{n_t - m}{n_t} \right)^{n_t}, \quad n_t \geq m, \quad t = 1, 2, \dots, \quad (19)$$

$$\overline{A}_t^s = n_t \left( 1 - \left( \frac{n_t - m}{n_t} \right)^{n_t} \right), \quad n_t \geq m, \quad t = 1, 2, \dots, \quad (20)$$

$$n_1 = n, \quad (21)$$

$$n_{t+1} = n_t \left( \frac{n_t - m}{n_t} \right)^{n_t}, \quad n_t \geq m, \quad t = 2, 3, \dots, \quad (22)$$

$$\overline{f}_t = \frac{\sum_{d=1}^t n_d \left( 1 - \left( \frac{n_d - m}{n_d} \right)^{n_d} \right)}{n} = \frac{n - n_t \left( \frac{n_t - m}{n_t} \right)^{n_t}}{n}, \quad n_t \geq m. \quad (23)$$

Let us now make an important observation: Formula (17) that we derived above is completely general and subsumes more special formulas found in the literature. Take, for example, the special case where  $t = 1$  and  $m = 1$ . For these values, (17) computes the expected number of vacant restaurants at the end of day 1 for the standard one-stop KPRP. We can readily derive the expected utilization ratio for day 1, as shown below.

$$\overline{f}_1 = \frac{n - n \left( \frac{n-1}{n} \right)^n}{n} = 1 - \left( \frac{n-1}{n} \right)^n. \quad (24)$$

One assumption that is taken for granted in the literature is that the number of agents  $n$  tends to infinity. By recalling that  $\lim_{n \rightarrow \infty} \left( 1 + \frac{x}{n} \right)^n \rightarrow e^x$ , we see that  $\lim_{n \rightarrow \infty} \overline{f} \rightarrow 1 - e^{-1}$ , which is in complete agreement with a well known result of the literature. When  $n \rightarrow \infty$ , the following Corollary 1 provides useful approximations of the formulas of Theorem 1. Its proof follows almost verbatim the proof of Theorem 1 and is, therefore, omitted.

**Corollary 1.** *If we assume that  $n \rightarrow \infty$ , then the following approximations hold, where  $t$  is the day in question.*

$$n_{t+1} \approx ne^{-tm}, \quad t = 1, 2, \dots, \quad (25)$$

$$\overline{A}_t^u \approx ne^{-tm}, \quad t = 1, 2, \dots, \quad (26)$$

$$\overline{A}_t^s \approx n(1 - e^{-m})e^{-(t-1)m} = ne^{-(t-1)m} - ne^{-tm}, \quad t = 1, 2, \dots, \quad (27)$$

$$\overline{R}_t^v \approx ne^{-tm}, \quad t = 1, 2, \dots, \quad (28)$$

$$\overline{R}_t^r \approx ne^{-(t-1)m} - ne^{-tm}, \quad t = 1, 2, \dots, \quad (29)$$

$$VP_t \approx e^{-tm}, \quad t = 1, 2, \dots, \quad (30)$$

$$\overline{f}_t \approx 1 - e^{-tm}, \quad t = 1, 2, \dots, \quad (31)$$

$$\overline{f}_\infty \approx 1. \quad (32)$$

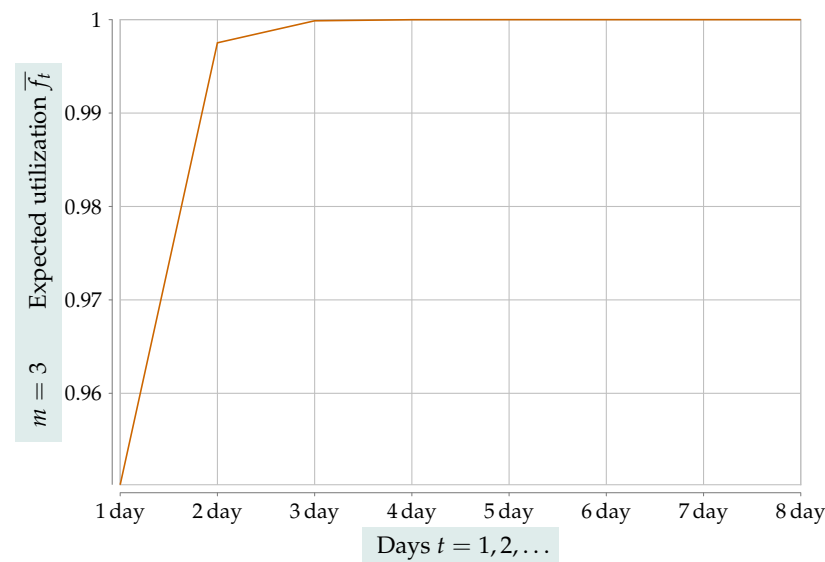
To demonstrate how the exact Formulas (16)–(23) reflect the daily evolution of the  $m$ -DKPRG, we study a 3-DKPRG game in the following example.

**Example 1.** *In this game, the number  $m$  of steps is 3, meaning that each agent may visit up to three restaurants if need be, and the number  $n$  of agents is  $10^9$ . Such an instance, with a large number of agents, can serve as the best demonstration of the dramatic improvement that can be obtained*

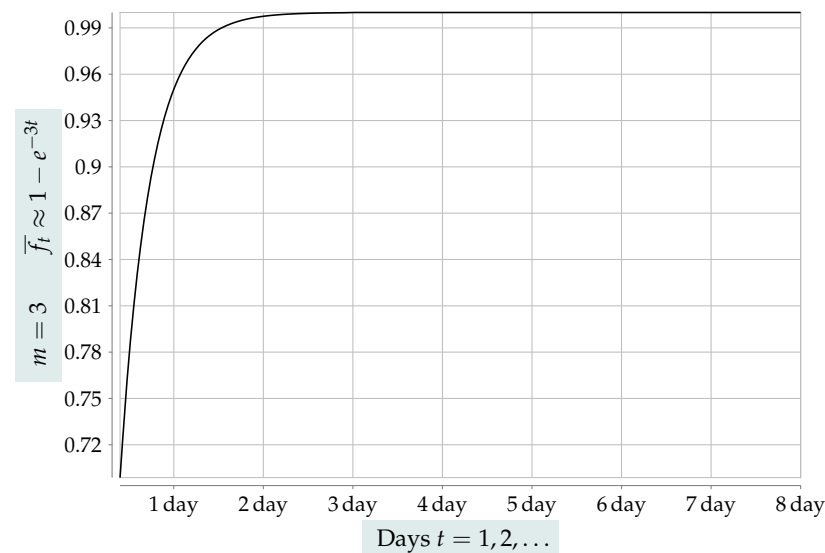


by an increase in the number of steps. Indeed, the numerical application of the Formulas (16)–(23) is depicted in Figure 5. A visual inspection corroborates this expectation, as one can see that all restaurants are utilized by the end of day 4, and the utilization at the end of the first day is already up to an impressive 0.95 and becomes 1, for all practical purposes, at the end of day 3.

The value of the approximate Formulas (25)–(32) lies in the fact that they can provide easy-to-compute and particularly good approximations for a large  $n$ . A simple comparison of the approximation shown in Figure 5 with Figure 6 ascertains their accuracy.



**Figure 5.** This figure depicts the expected utilization  $\bar{f}_t$ , as given by Equation (23), for the 3-DKPRG studied herein.



**Figure 6.** This figure depicts the approximate expected utilization  $\bar{f}_t \approx 1 - e^{-3t}$  as given by Equation (31) for  $m = 3$ .

## 6. Discussion and Conclusions

This work explored a completely new angle of the Kolkata Paise Restaurant Problem. The topological layout of the restaurants takes center stage in this new paradigm. Initially, we explicitly stated certain assumptions that are only implicitly present in the standard formulation of the game. Having done that, we undertook the radical step of going past them

and creating an entirely new setting. The critical examination of the topological setting of the game enhanced our perception regarding the locations of the restaurants and suggested a more realistic topological layout. We argued that their uniform distribution over the entire city area is the most logical, fair, and probable situation. As a result, we defined a new version of the game that is spatially distributed and is aptly named the Distributed Kolkata Paise Restaurant Game (DKPRG). The uniform probabilistic distribution of the restaurants enabled us to rigorously prove that, as their number  $n$  increases, the restaurants *get closer*, and the distance between adjacent restaurants decreases. In such a network, every customer has the opportunity to pass through more than one restaurant within the allowed time window. The agents now become traveling salesmen, and this led us to suggest the innovative idea that the TSP can be used to increase the chances of success in this game. This culminated in the development of a new and more efficient strategy where every agent uses metaheuristics to solve their personalized TSP, thereby achieving greater utilization.

After rigorously formulating the DKPRG, we proved completely general formulas that assert the increase in the utilization of our scheme. We established that utilization ranging from 0.85 to 0.95 is achievable. Apart from the exact formulas, we also derived the approximate Formulas (25)–(31). They can be quite useful because they are considerably easier to compute and are exceedingly good approximations for a large  $n$ . This fact is easily corroborated by comparing Figure 5 to the approximation shown in Figure 6. Let us remark that the derived equations generalize previously presented formulas in the literature.

It is worth mentioning that the fact that our strategy exhibits very rapid convergence to the steady state of utilization, 1.0, can potentially be used to address the following situation. An issue that remains and is common to almost all works in the literature is the simple matter that a near-optimal utilization may not, in general, be optimal for every agent individually. A socially efficient outcome, where every agent eats lunch and every restaurant has a customer to serve, is not necessarily optimal for the individual customer, in the sense that an agent may be served in a restaurant of low preference. A possible solution to this might be to reset the game periodically. We expect that adopting a reset period, i.e., setting a specific period of days after which the system is reset, and the game starts from scratch, may alleviate this drawback.

We believe that the proposed game can be successfully applied to many real-life situations of practical importance. For example, the distributed nature of the proposed game would make it particularly suitable for tackling the problem studied in [7]. There, a region or a country provides hospitals and beds in different parts of an area. However, there are cases of patients who prefer to go to higher-ranking hospitals at a greater distance than to lower-ranking hospitals at shorter distances. This results in a long waiting list as both local and non-local patients wait more, causing untimely treatment for patients (as the estimated number of patients increases) and a social waste of services for inaccessible hospitals. In the health care context, there are two more cases in which the game we suggest could offer a solution. The first concerns ambulance offload delays, i.e., when an ambulance arriving at the emergency department cannot leave the patient because there are no beds available. This solution can work on two levels, choosing the best route for the ambulance to arrive on time and contributing to better management of hospital resources in order for the treatment of the patient to start immediately [49]. At the same time, numerous problems concerning route optimization, resource allocation, and utilization appear in the area of public transport. The most common question about transportation is how limited resources should be utilized to maximize social benefits, taking into consideration the peak periods and the unequal distribution of resources [50]. A traffic problem that our proposal could tackle, similar to the previous one, is the school bus routing problem [51]. The difficulty of the school bus routing problem stems from the strict routine and timetable the buses have to follow, the requirement that the students need to arrive on time for their classes, and the calculation of the optimal route for the bus in order to collect all the students.

Another possible direction for future work could include extensive experimental tests and further investigation of other versions of the TSP. For instance, there exists a more

restrictive version of the TSP, the Traveling Salesman Problem with Time Windows (TSP-TW). TSP-TW is a constrained version of the TSP in which the salesman must visit the cities within a specific time window. This version is even more complicated and difficult to solve. However, the inherent time constraints built in to the TSP-TW may provide for even more realistic modeling of the DKPRG, so it is a research avenue that we believe is worth pursuing.

**Author Contributions:** Conceptualization, T.A. and K.K.; methodology, T.A.; validation, C.P. and K.K.; formal analysis, T.A.; investigation, C.P. and K.K.; writing—original draft preparation, K.K., C.P. and T.A.; writing—review and editing, C.P. and K.K.; visualization, K.K. and C.P.; supervision, T.A.; project administration, T.A. and K.K. 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:** Not Applicable, the study does not report any data.

**Acknowledgments:** The authors would like to thank Alexander Andronikos for his help in drawing the figures that appear in this manuscript and also express their sincere appreciation for the insightful comments and suggestions by the reviewers that enabled them to improve the initial version of this manuscript.

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

## Appendix A. Proofs

### Appendix A.1. Proof of Proposition 1

**Proof.** Let  $B$  stand for the whole city area and let  $B_1, \dots, B_n$  be the  $n$  regions. The game settings assert that:

1.  $B_1 \cup \dots \cup B_n = B$ ,
2.  $B_p \cap B_q = \emptyset$ , if  $1 \leq p \neq q \leq n$ , and
3.  $\text{area}(B_1) = \text{area}(B_2) = \dots = \text{area}(B_n) = \frac{\text{area}(B)}{n}$ .

From the fact that the  $n$  restaurants are uniformly distributed, we know from (1) that for every restaurant  $r_j$ ,  $1 \leq j \leq n$ , and every region  $B_p$ ,  $1 \leq p \leq n$ ,

$$P(r_j \in B_p) = \frac{\text{area}(B_p)}{\text{area}(B)} = \frac{1}{n}, \quad 1 \leq j, p \leq n. \quad (\text{A1})$$

We define the auxiliary random variables  $N_{pj}$ , where  $1 \leq p, j \leq n$ .

$$N_{pj} = \begin{cases} 1 & \text{if restaurant } r_j \text{ is located in region } B_p \\ 0 & \text{otherwise} \end{cases}. \quad (\text{A2})$$

By combining (A1) with (A2), we conclude that

$$N_{pj} = \begin{cases} 1 & \text{with probability } \frac{1}{n} \\ 0 & \text{with probability } \frac{n-1}{n} \end{cases}, \quad 1 \leq p, j \leq n. \quad (\text{A3})$$

Then, the random variable

$$N_p = \sum_{j=1}^n N_{pj} \quad (1 \leq p \leq n) \quad (\text{A4})$$

gives the number of restaurants in region  $B_p$ ,  $1 \leq p \leq n$ . We are not interested in the actual value of the random variable  $N_p$ , per se, but in its expected value  $E[N_p]$ . The latter can be easily computed as follows.

$$\overline{N_p} = E[N_p] \stackrel{(A4)}{=} E\left[\sum_{j=1}^n N_{pj}\right] = \sum_{j=1}^n E[N_{pj}] \stackrel{(A3)}{=} \sum_{j=1}^n \left(1 \cdot \frac{1}{n}\right) = 1 \quad (A5)$$

This proves that the expected number of restaurants in each region is 1.  $\square$

#### Appendix A.2. Proof of Proposition 2

**Proof.** Consider two adjacent regions  $B_p$  and  $B_q$ . By (4), this means that  $d(B_p, B_q) = 0$ , which in turn implies that  $\forall \varepsilon \exists x \in B_p \exists y \in B_q$  such that  $d(x, y) \leq \varepsilon$  (\*). In view of Proposition 1, one expects to find exactly one restaurant in  $B_p$  and exactly one restaurant in  $B_q$ . Let  $r_p$  and  $r_q$  be the restaurants located in  $B_p$  and  $B_q$ , respectively, and consider the distance  $d(r_p, r_q)$  between them. By the triangle inequality, which is a fundamental property of every distance function, we know that  $d(r_p, r_q) \leq d(r_p, x) + d(x, y) + d(y, r_q)$ ,  $\forall x \in B_p \forall y \in B_q$  (\*\*). From (\*) and (\*\*) we conclude that  $\forall \varepsilon \exists x \in B_p \exists y \in B_q$  such that  $d(r_p, r_q) \leq d(r_p, x) + d(y, r_q) + \varepsilon$  (\*\*\*). According to (5),  $d(r_p, x) \leq \text{diam} B_p$  and  $d(y, r_q) \leq \text{diam} B_q$ . These two relations combined with (\*\*\*), give that  $d(r_p, r_q) \leq \text{diam} B_p + \text{diam} B_q$ , as desired.  $\square$

#### Appendix A.3. Proof of Theorem 1

**Proof.** The proof of the given formulas is as follows.

1. We first prove that the vacancy probability at the *beginning* of stop  $z$ ,  $1 \leq z \leq m$ , of day  $t$  is

$$VP_{t,z} = \left(\frac{n_t + 1 - z}{n_t}\right)^{n_t}, \quad 1 \leq z \leq m. \quad (A6)$$

- At the beginning of stop 1 of day  $t$ , the expected number of restaurants that have not served any customer yet is equal to the expected number  $n_t$  of active agents. On day  $t$ , the game is all about the active agents and the restaurants that have never been utilized up to now. At this moment in time, all these restaurants are still unutilized, so vacancy is a certainty. Thus, indeed  $VP_{t,1} = 1$ .
- At the beginning of day  $t$ , the expected number of restaurants that have not served any customer yet is equal to the expected number  $n_t$  of active players. At the beginning of stop 2 of day  $t$ , the probability that restaurant  $r$  has not served lunch yet is precisely the probability that  $r$  never appears in position 1 in any tour of the active players. This last probability is given from (13), where of course  $n$  must now be replaced by  $n_t$ . Hence,  $VP_{t,2} = \left(\frac{n_t - 1}{n_t}\right)^{n_t}$ .
- The customers who have failed to eat lunch at their first destination immediately proceed to their second destination. However, an added complication arises now. It may be the case that  $r_{l_2}$  is already occupied from stop 1. In such a case  $r_{l_2}$  is completely unavailable. In view of this fact, we may conclude that the restaurants that are vacant at the beginning of stop 3 must satisfy two properties:
  - (a) they must be vacant at the beginning of stop 2, which means that must never appear in position 1 in any tour of the active players, and
  - (b) they must never appear in position 2 in any tour of the active players.

Therefore,  $VP_{t,3} = P(r \text{ never appears in positions 1 or 2 in } n_t \text{ tours}) \stackrel{(15)}{=} \left(\frac{n_t - 2}{n_t}\right)^{n_t}$ , which is again in agreement with (A6) when  $z = 3$ .

- The same reasoning can be employed to show that the vacancy probability  $VP_{t,z}$  at the beginning of stop  $z$  of day  $t$  is  $VP_{t,z} = \left(\frac{n_t+1-z}{n_t}\right)^{n_t}$ . Hence, we have proved the validity of (A6).
  - Finally, the probability that a restaurant  $r$  that has not served any customer up to day  $t$  is still vacant at the end of stop  $m$  of day  $t$ , is the probability that  $r$  never appears in positions 1, or, 2, or  $\dots$ , or  $m$  in any tour of the active players. Thus,  $VP_t = \left(\frac{n_t-m}{n_t}\right)^{n_t}$ , which verifies (16), as desired. For the Equation (16) to be meaningful, it must hold that  $n_t \geq m$ . The physical meaning of this restriction is that (16) is correct as long as there are at least as many active players as stops  $m$ . If on some day  $t$  we have that  $n_t \leq m$ , then the strategy we adhere to ensures that all  $n_t$  active players manage to eat lunch during day  $t$ .
2. The expected number of restaurants that remained vacant at the end of day  $t$  is given by  $\overline{R}_t^v$ . First, we express probabilistically those restaurants that remain vacant after all agents visit their first  $m$  destinations. We define the family of random variables  $R_{tj}^v$ ,  $1 \leq j \leq n_t$ . Specifically, if  $R_{tj}^v$  has the value 1, then restaurant  $r_j$  is *vacant* at the end of day  $t$ , whereas if  $R_{tj}^v$  is 0, then  $r_j$  is occupied.

$$R_{tj}^v = \begin{cases} 1 & \text{if } r_j \text{ is vacant at the end of day } t \\ 0 & \text{otherwise} \end{cases}, \quad 1 \leq j \leq n_t. \quad (\text{A7})$$

By combining Equations (A7) and (16), we deduce that

$$R_{tj}^v = \begin{cases} 1 & \text{with probability } \left(\frac{n_t-m}{n_t}\right)^{n_t} \\ 0 & \text{with probability } 1 - \left(\frac{n_t-m}{n_t}\right)^{n_t} \end{cases}, \quad 1 \leq j \leq n_t. \quad (\text{A8})$$

We define the random variable  $R_t^v$ , which counts the *the number of restaurants that are vacant* at the end of day  $t$ :

$$R_t^v = \sum_{j=1}^{n_t} R_{tj}^v. \quad (\text{A9})$$

We are interested not in the actual value of  $R_t^v$ , but in its expected value  $E[R_t^v]$ . In view of (A9) and the linearity of the expected value operator

$$\overline{R}_t^v = E[R_t^v] \stackrel{(\text{A9})}{=} E\left[\sum_{j=1}^{n_t} R_{tj}^v\right] = \sum_{j=1}^{n_t} E[R_{tj}^v] \stackrel{(\text{A8})}{=} \sum_{j=1}^{n_t} \left(\frac{n_t-m}{n_t}\right)^{n_t} \quad (\text{A10})$$

$$= n_t \left(\frac{n_t-m}{n_t}\right)^{n_t}, \quad (\text{A11})$$

which verifies (17).

3.  $\overline{R}_t^r$  denotes the expected number of restaurants that were visited by an agent by the end of day  $t$ . The random variables  $R_{tj}^r$ ,  $1 \leq j \leq n_t$ , indicate whether restaurant  $r_j$  is occupied or not at the end of day  $t$ .

$$R_{tj}^r = \begin{cases} 1 & \text{if } r_j \text{ is occupied at the end of day } t \\ 0 & \text{otherwise} \end{cases}, \quad 1 \leq j \leq n_t. \quad (\text{A12})$$

By combining (A12) and Equation (16), we deduce that

$$R_{tj}^r = \begin{cases} 1 & \text{with probability } 1 - \left(\frac{n_t-m}{n_t}\right)^{n_t} \\ 0 & \text{with probability } \left(\frac{n_t-m}{n_t}\right)^{n_t} \end{cases}, \quad 1 \leq j \leq n_t. \quad (\text{A13})$$

Having done that, we define the random variable  $R_t^r$ , which counts the *the number of restaurants that are occupied* at the end of day  $t$ .

$$R_t^r = \sum_{j=1}^{n_t} R_{tj}^r. \quad (\text{A14})$$

In view of (A14) and the linearity of the expected value operator

$$\begin{aligned} \overline{R}_t^r &= E[R_t^r] \stackrel{(\text{A14})}{=} E\left[\sum_{j=1}^{n_t} R_{tj}^r\right] = \sum_{j=1}^{n_t} E[R_{tj}^r] \stackrel{(\text{A13})}{=} \sum_{j=1}^{n_t} 1 \cdot \left(1 - \left(\frac{n_t - m}{n_t}\right)^{n_t}\right) \\ &= n_t \left(1 - \left(\frac{n_t - m}{n_t}\right)^{n_t}\right), \end{aligned} \quad (\text{A15})$$

which verifies (18).

4. The rules of the game stipulate that the number of customers who have not managed to eat lunch at the end of day  $t$  is equal to the number of restaurants that have not served any customer at the end of day  $t$ . Hence, their expected values are also equal, i.e.,  $\overline{A}_t^u = \overline{R}_t^r$  and (19) is proved.
5. Likewise, the adopted strategy ensures that the number of the active players who succeeded in eating lunch at the *end* of day  $t$  is equal to the number of restaurants that, although they had not served any agent up to day  $t$ , managed to accommodate a customer by the end of day  $t$ . Their expected values are also equal, i.e.,  $\overline{A}_t^s = \overline{R}_t^r$  and (20) is proved.
6. We are now in a position to compute the expected number of active players.
  - At the beginning of the first day, there are  $n$  active players.
  - The adopted strategy in the  $m$ -DKPRG ensures that the number of agents who have not eaten lunch at the end of day  $t$  is always equal to the number of active players on day  $t + 1$ . Thus, the expected number of active agents on day  $t + 1$  is equal to the expected number of unsatisfied agents at the end of day  $t$ :

$$n_{t+1} = \overline{A}_t^u. \quad (\text{A16})$$

By combining the previous result (A16) with (19), we derive

$$n_{t+1} = n_t \left(\frac{n_t - m}{n_t}\right)^{n_t}, \quad (\text{A17})$$

which establishes the validity of (22), as desired.

7. The expected utilization  $\overline{f}_t$  for day  $t = 1, 2, \dots$ , is the ratio of the expected number of agents who were served during day  $t$ . This is equal to the expected number of customers who ate lunch on day 1, plus the expected number of the *additional* customers who ate lunch on day 2, and so on. The *additional* agents on day  $t$  are precisely those agents who had failed to eat lunch prior to day  $t$ , but succeeded on day  $t$ . Their expected number is  $\overline{A}_t^s$ , which is given by Equation (20). Hence, the total number of agents who have eaten lunch up to and including day  $t$  is given by

$$\sum_{d=1}^t \overline{A}_d^s \stackrel{(\text{20})}{=} \sum_{d=1}^t n_d \left(1 - \left(\frac{n_d - m}{n_d}\right)^{n_d}\right), \quad t = 1, 2, \dots. \quad (\text{A18})$$

An equivalent way to compute this number is by subtracting from the total number of agents  $n$  the expected number of agents who failed to eat lunch on day  $t$ , which is  $\overline{A}_t^u$  given by Equation (19). Thus,



$$n - \overline{A}_t^u \stackrel{(19)}{=} n - n_t \left( \frac{n_t - m}{n_t} \right)^{n_t}, \quad t = 1, 2, \dots \quad (\text{A19})$$

Together the two Formulas (A18) and (A19), allow us to conclude that

$$\overline{f}_t \stackrel{(A18)}{=} \frac{\sum_{d=1}^t n_d \left( 1 - \left( \frac{n_d - m}{n_d} \right)^{n_d} \right)}{n} \stackrel{(A19)}{=} \frac{n - n_t \left( \frac{n_t - m}{n_t} \right)^{n_t}}{n}, \quad (\text{A20})$$

which establishes the validity of (23), as desired.

□

## References

- Feillet, D.; Dejax, P.; Gendreau, M. Traveling Salesman Problems with Profits. *Transp. Sci.* **2005**, *39*, 188–205. [\[CrossRef\]](#)
- Chakrabarti, B.K. Kolkata restaurant problem as a generalised el farol bar problem. In *Econophysics of Markets and Business Networks*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 239–246.
- Chakrabarti, A.S.; Chakrabarti, B.K.; Chatterjee, A.; Mitra, M. The Kolkata Paise Restaurant problem and resource utilization. *Phys. A Stat. Mech. Appl.* **2009**, *388*, 2420–2426. [\[CrossRef\]](#)
- Ghosh, A.; Chakrabarti, A.S.; Chakrabarti, B.K. Kolkata Paise Restaurant problem in some uniform learning strategy limits. In *Econophysics and Economics of Games, Social Choices and Quantitative Techniques*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 3–9.
- Chakrabarti, B.K.; Chatterjee, A.; Ghosh, A.; Mukherjee, S.; Tamir, B. *Econophysics of the Kolkata Restaurant Problem and Related Games*; Springer International Publishing: Berlin/Heidelberg, Germany, 2017. [\[CrossRef\]](#)
- Banerjee, P.; Mitra, M.; Mukherjee, C. Kolkata paise restaurant problem and the cyclically fair norm. In *Econophysics of Systemic Risk and Network Dynamics*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 201–216.
- Ghosh, A.; Biswas, S.; Chatterjee, A.; Chakrabarti, A.S.; Naskar, T.; Mitra, M.; Chakrabarti, B.K. Kolkata paise restaurant problem: An introduction. In *Econophysics of Systemic Risk and Network Dynamics*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 173–200.
- Ghosh, D.; Chakrabarti, A.S. Emergence of distributed coordination in the Kolkata paise restaurant problem with finite information. *Phys. A Stat. Mech. Appl.* **2017**, *483*, 16–24. [\[CrossRef\]](#)
- Yang, P.; Iyer, K.; Frazier, P.I. Mean field equilibria for competitive exploration in resource sharing settings. In Proceedings of the 25th International Conference on World Wide Web, Montreal, QC, Canada, 11–15 April 2016; pp. 177–187.
- Agarwal, S.; Ghosh, D.; Chakrabarti, A.S. Self-organization in a distributed coordination game through heuristic rules. *Eur. Phys. J. B* **2016**, *89*, 266. [\[CrossRef\]](#)
- Milchtaich, I. Congestion games with player-specific payoff functions. *Games Econ. Behav.* **1996**, *13*, 111–124. [\[CrossRef\]](#)
- Martin, L. Extending Kolkata Paise Restaurant Problem to Dynamic Matching in Mobility Markets. *Jr. Manag. Sci.* **2019**, *4*, 1–34. [\[CrossRef\]](#)
- Abergel, F.; Chakrabarti, B.K.; Chakraborti, A.; Ghosh, A. *Econophysics of Systemic Risk and Network Dynamics*; Springer: Berlin/Heidelberg, Germany, 2012.
- Park, T.; Saad, W. Kolkata paise restaurant game for resource allocation in the internet of things. In Proceedings of the 51st Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 29 October–1 November 2017; pp. 1774–1778.
- Sinha, A.; Chakrabarti, B.K. Phase transition in the Kolkata Paise Restaurant problem. *Chaos Interdiscip. J. Nonlinear Sci.* **2020**, *30*, 083116. [\[CrossRef\]](#)
- Meyer, D.A. Quantum strategies. *Phys. Rev. Lett.* **1999**, *82*, 1052. [\[CrossRef\]](#)
- Eisert, J.; Wilkens, M.; Lewenstein, M. Quantum games and quantum strategies. *Phys. Rev. Lett.* **1999**, *83*, 3077. [\[CrossRef\]](#)
- Li, A.; Yong, X. Entanglement Guarantees Emergence of Cooperation in Quantum Prisoner's Dilemma Games on Networks. *Sci. Rep.* **2014**, *4*, 6286. [\[CrossRef\]](#)
- Deng, X.; Zhang, Q.; Deng, Y.; Wang, Z. A novel framework of classical and quantum prisoner's dilemma games on coupled networks. *Sci. Rep.* **2016**, *6*, 23024. [\[CrossRef\]](#) [\[PubMed\]](#)
- Giannakis, K.; Theocharopoulou, G.; Papalitsas, C.; Fanarioti, S.; Andronikos, T. Quantum Conditional Strategies and Automata for Prisoners' Dilemmata under the EWL Scheme. *Appl. Sci.* **2019**, *9*, 2635. [\[CrossRef\]](#)
- Andronikos, T.; Sirokofskich, A.; Kastampolidou, K.; Varvouzou, M.; Giannakis, K.; Singh, A. Finite Automata Capturing Winning Sequences for All Possible Variants of the PQ Penny Flip Game. *Mathematics* **2018**, *6*, 20. [\[CrossRef\]](#)
- Andronikos, T.; Sirokofskich, A. The connection between the PQ penny flip game and the dihedral groups. *Mathematics* **2021**, *9*, 1115. [\[CrossRef\]](#)
- Giannakis, K.; Papalitsas, C.; Kastampolidou, K.; Singh, A.; Andronikos, T. Dominant Strategies of Quantum Games on Quantum Periodic Automata. *Computation* **2015**, *3*, 586–599. [\[CrossRef\]](#)
- Andronikos, T.; Stefanidakis, M. A two-party quantum parliament. *Algorithms* **2022**, *15*, 62. [\[CrossRef\]](#)

25. Kastampolidou, K.; Nikiforos, M.N.; Andronikos, T. A Brief Survey of the Prisoners' Dilemma Game and Its Potential Use in Biology. In *Advances in Experimental Medicine and Biology*; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 315–322. [\[CrossRef\]](#)
26. Sharif, P.; Heydari, H. Strategies in a symmetric quantum Kolkata restaurant problem. *AIP Conf. Proc.* **2012**, *1508*, 492–496. [\[CrossRef\]](#)
27. Sharif, P.; Heydari, H. An introduction to multi-player, multi-choice quantum games: Quantum minority games & kolkata restaurant problems. In *Econophysics of Systemic Risk and Network Dynamics*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 217–236.
28. Ramzan, M. Three-player quantum Kolkata restaurant problem under decoherence. *Quantum Inf. Process.* **2013**, *12*, 577–586. [\[CrossRef\]](#)
29. Voigt, B.F. Der Handlungsreisende, wie er sein soll und was er zu thun hat, um Aufträge zu erhalten und eines glücklichen Erfolgs in seinen Geschäften gewiss zu sein. In *Commis-Voageur*, Ilmenau; Verlag Schramm: Kiel, Germany, 1981.
30. Ascheuer, N.; Fischetti, M.; Grötschel, M. Solving the Asymmetric Travelling Salesman Problem with time windows by branch-and-cut. *Math. Program.* **2001**, *90*, 475–506. [\[CrossRef\]](#)
31. Gutin, G.; Punnen, A.P. *The Traveling Salesman Problem and Its Variations*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006; Volume 12.
32. Bianchi, L.; Dorigo, M.; Gambardella, L.M.; Gutjahr, W.J. A survey on metaheuristics for stochastic combinatorial optimization. *Nat. Comput.* **2009**, *8*, 239–287. [\[CrossRef\]](#)
33. Blum, C.; Roli, A. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv. (CSUR)* **2003**, *35*, 268–308. [\[CrossRef\]](#)
34. Papalitsas, C.; Giannakis, K.; Andronikos, T.; Theotokis, D.; Sifaleras, A. Initialization methods for the TSP with Time Windows using Variable Neighborhood Search. In Proceedings of the 6th International Conference on Information, Intelligence, Systems and Applications (IISA 2015), Corfu, Greece, 6–8 July 2015; pp. 1–6. [\[CrossRef\]](#)
35. Papalitsas, C.; Karakostas, P.; Kastampolidou, K. A Quantum Inspired GVNS: Some Preliminary Results. In *Advances in Experimental Medicine and Biology*; Springer International Publishing: Berlin/Heidelberg, Germany, 2017; pp. 281–289. [\[CrossRef\]](#)
36. Papalitsas, C.; Karakostas, P.; Andronikos, T.; Sioutas, S.; Giannakis, K. Combinatorial GVNS (General Variable Neighborhood Search) Optimization for Dynamic Garbage Collection. *Algorithms* **2018**, *11*, 38. [\[CrossRef\]](#)
37. Papalitsas, C.; Karakostas, P.; Giannakis, K.; Sifaleras, A.; Andronikos, T. Initialization methods for the TSP with Time Windows using qGVNS. In Proceedings of the 6th International Symposium on Operational Research, OR in the Digital Era—ICT Challenges, Thessaloniki, Greece, 8–10 June 2017.
38. Papalitsas, C.; Andronikos, T. Unconventional GVNS for Solving the Garbage Collection Problem with Time Windows. *Technologies* **2019**, *7*, 61. [\[CrossRef\]](#)
39. Papalitsas, C.; Andronikos, T.; Karakostas, P. Studying the Impact of Perturbation Methods on the Efficiency of GVNS for the ATSP. In *Variable Neighborhood Search*; Springer International Publishing: Berlin/Heidelberg, Germany, 2019; pp. 287–302. [\[CrossRef\]](#)
40. Papalitsas, C.; Karakostas, P.; Andronikos, T. A Performance Study of the Impact of Different Perturbation Methods on the Efficiency of GVNS for Solving TSP. *Appl. Syst. Innov.* **2019**, *2*, 31. [\[CrossRef\]](#)
41. Papalitsas, C.; Andronikos, T.; Giannakis, K.; Theocharopoulou, G.; Fanarioti, S. A QUBO Model for the Traveling Salesman Problem with Time Windows. *Algorithms* **2019**, *12*, 224. [\[CrossRef\]](#)
42. Mertens, J.F. Supergames. In *Game Theory*; Springer: Berlin/Heidelberg, Germany, 1989; pp. 238–241.
43. Rego, C.; Gamboa, D.; Glover, F.; Osterman, C. Traveling salesman problem heuristics: Leading methods, implementations and latest advances. *Eur. J. Oper. Res.* **2011**, *211*, 427–441. [\[CrossRef\]](#)
44. Johnson, R.; Pilcher, M.G. *The Traveling Salesman Problem*; Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., Shmoys, D.B., Eds.; John Wiley & Sons: Hoboken, NJ, USA, 1985.
45. Pitman, J. *Probability*; Springer: New York, NY, USA, 1993. [\[CrossRef\]](#)
46. Klenke, A. *Probability Theory*; Springer: London, UK, 2014. [\[CrossRef\]](#)
47. Munkres, J. *Topology*; Prentice Hall, Inc.: Upper Saddle River, NJ, USA, 2000.
48. DasGupta, A. *Fundamentals of Probability: A First Course*; Springer: Berlin/Heidelberg, Germany, 2010.
49. Kobusingye, O.C.; Hyder, A.A.; Bishai, D.; Joshipura, M.; Hicks, E.R.; Mock, C. Emergency medical services. In *Disease Control Priorities in Developing Countries*, 2nd ed.; World Bank Publications: Washington, DC, USA, 2006.
50. Gkiotsalitis, K.; Wu, Z.; Cats, O. A cost-minimization model for bus fleet allocation featuring the tactical generation of short-turning and interlining options. *Transp. Res. Part C Emerg. Technol.* **2019**, *98*, 14–36. [\[CrossRef\]](#)
51. Schittekat, P.; Kinable, J.; Sörensen, K.; Sevaux, M.; Spieksma, F.; Springael, J. A metaheuristic for the school bus routing problem with bus stop selection. *Eur. J. Oper. Res.* **2013**, *229*, 518–528. [\[CrossRef\]](#)