robotics

MDPI

Article

Minimum Energy Utilization Strategy for Fleet of Autonomous Robots in Urban Waste Management

Valeria Bladinieres Justo [1,†], Abhishek Gupta [2,*,†], Tobias Fritz Umland [1] and Dietmar Göhlich [2]

1 Electrical Engineering and Computer Science, Technische Universität Berlin, 10623 Berlin, Germany; bladinieresjusto@campus.tu-berlin.de (V.B.J.); t.umland@campus.tu-berlin.de (T.F.U.)
2 Methods of Product Development and Mechatronics, Technische Universität Berlin, 10623 Berlin, Germany; dietmar.goehlich@tu-berlin.de
* Correspondence: abhishek.gupta@tu-berlin.de
† These authors contributed equally to this work.

Abstract: Many service robots have to operate in a variety of different Service Event Areas (SEAs). In the case of the waste collection robot MARBLE (Mobile Autonomous Robot for Litter Emptying) every SEA has characteristics like varying area and number of litter bins, with different distances between litter bins and uncertain filling levels of litter bins. Global positions of litter bins and garbage drop-off positions from MARBLEs after reaching their maximum capacity are defined as task-performing waypoints. We provide boundary delimitation for characteristics that describe the SEA. The boundaries interpolate synergy between individual SEAs and the developed algorithms. This helps in determining which algorithm best suits an SEA, dependent on the characteristics. The developed route-planning methodologies are based on vehicle routing with simulated annealing (VRPSA) and knapsack problems (KSPs). VRPSA uses specific weighting based on route permutation operators, initial temperature, and the nearest neighbor approach. The KSP optimizes a route's given capacity, in this case using smart litter bins (SLBs) information. The game-theory KSP algorithm with SLBs information and the KSP algorithm without SLBs information performs better on SEAs lower than 0.5 km^2, and with fewer than 50 litter bins. When the standard deviation of the fill rate of litter bins is ≈10%, the KSP without SLB is preferred, and if the standard deviation is between 25 and 40%, then the game-theory KSP is selected. Finally, the vehicle routing problem outperforms in SEAs with an area of $0.5 \leq 5$ km^2, 50–450 litter bins, and a fill rate of 10–40%.

Keywords: urban service robots; robot fleet management; vehicle routing problem; simulated annealing; sustainable waste management; varying Service Event Areas; Knapsack problem; game theory

Citation: Justo, V.B.; Gupta, A.; Umland, T.F.; Göhlich, D. Minimum Energy Utilization Strategy for Fleet of Autonomous Robots in Urban Waste Management. *Robotics* **2023**, *12*, 159. https://doi.org/10.3390/robotics12060159

Academic Editor: Kagan Eugene

Received: 24 October 2023
Revised: 17 November 2023
Accepted: 22 November 2023
Published: 23 November 2023

1. Introduction

Service robots fall under the domain of advanced robotics [1] and can be described as robots that carry out beneficial tasks for humans or equipment, with the exception of applications related to industrial automation [2]. Developing a service robot with diverse functionalities and implementing them in a fleet amongst other robots in a dynamic environment and in varying operational Service Event Areas (SEA) are two separate but interrelated, complex problems. Incorporating sustainability as a governance aspect for operational management can have numerous meanings when compared with the defined sustainable development goals (SDGs) in robotics and automated services [3]. During a service operation, it is necessary that the energy consumption of the robots performing the service and their assistance infrastructure is minimal, thereby supporting SDG-12—responsible consumption and production [3,4]. The authors of [5] optimized same-day deliveries for multiple robots at various depots, along with considerations for diverse conditions to reduce the average travel distance of robots to customers by incorporating agile planning with pre-empty returns to depots. The authors of [6] introduced an effective virtual-region-based shape control scheme for guiding a swarm of robots through unknown occluded environments with multiple targets, thereby optimizing the overall performance of a multi-robot system. The authors of [7] presented an algorithm for multi-unmanned aerial vehicles (UAV) path planning that adapts to heterogeneous global navigation satellite systems (GNSSs) coverage, allowing cooperative strategies in challenging conditions and independent flight in open sky scenarios, with simulations demonstrating consistent computational efficiency across varying numbers of UAVs and waypoints. For an urban waste-management process, with the help of a fleet of autonomous urban service robots—MARBLE (Mobile Autonomous Robot for Litter Emptying) [8]—the operation requirements vary from one SEA to another.One single-route-planning algorithm for varying operational requirements might not provide an efficient operation management in the form of the least energy-consuming routes to the multiple co-operative MARBLEs over various target litter bins. This is due to the fact that one route-planning algorithm functions better under certain operational characteristics than other algorithms for such multi-agent problems [9,10]. The methodologies behind such algorithms serve as solution for specific requirements and not for all the requirements of the varying operation SEAs [11]. Developing a dedicated route-planning algorithm for every individual SEA is not just time-consuming, but is also a non-viable solution, as the characteristics of an SEA can change any given moment and it is not necessary for a new SEA to have characteristics that are similar to another SEA. To provide a solution for this scientific problem, this paper provides a solution that categorizes the characteristics of the operational SEAs based on their impact on the energy consumption of the service robots and its assistance infrastructure. The categorized characteristics are incorporated in various route-planning algorithms, and the results are compared on the basis of minimum operational energy consumption. The operational principle of minimizing energy consumption is designed to align with SDG-12, emphasizing responsible and sustainable energy use. Routes were simulated for three authentic SEAs in Berlin, each with diverse characteristics, as well as for multiple fictional SEAs.In Section 2, the various methodologies behind route-planning algorithms have been compared on the basis of the characteristics and requirements of the SEAs. In Section 3.1, the use-case MARBLE and the SEAs have been explained in detail. In Section 3.2, our methodology for the three SEAs with different characteristics to be emptied with the help of the fleet of MARBLE is explained for providing a minimum energy utilization strategy. In Section 3.3, the approach for choosing the algorithm with the lowest energy consumption for the characteristics of the SEA is provided. In Section 4, the results are demonstrated and compared in two different subsections. Section 4.1 shows the results of the simulated implementation of the method in actual Berlin scenarios. Section 4.2 shows results for various self developed use-cases and explains how the results for indefinite types of SEAs can be generalized. Finally, a conclusion and an outlook regarding future work has been outlaid in Section 5.

2. State of the Art

For various autonomous service robots in the field of waste management, the biggest challenge lies not only in the the automation of the process, but also in the operational management, in the form of route planning for the Service Event Area [8,12,13]. Optimized route planning means the shortest path and least possible energy consumption for the overall waste-management service [14]. The route-planning algorithm can be solved as a vehicle routing problem using meta-heuristics like simulated annealing for single [15] and multiple actors [8]. The knapsack problem is a combinatorial optimization problem related to capacity constraints; it focuses on optimizing the weight of items in a specific sack. In previous work, the KSP was used for route planning with a single robot and smart litter bin infrastructure [16]. The KSP can be used with hardness results, as explained in [17]; they elaborate on arbitrary weights and profit functions. In the same research work, a new approach has been demonstrated, where the items can be shared between neighbors, only when at least one of an item's neighbors is also picked, based on asking whether that item can be chosen [17]. Further approaches have been researched, as discussed by [18], which integrates the knapsack problem with agent payoff functions to facilitate effective negotiations among the agents involved. Furthermore, [19] emphasizes the adoption of a bounded knapsack model, where agents base their decision making on current local data. A sharing system for exploring many targets with several robots is described in [20], along with a navigation algorithm for choosing the best path. The authors formulated the underlying issue as a set-partitioning problem and then use the branch-and-price method to resolve it [20]. Moreover, investigations have been performed on the association of the multi-agent systems and KSP approach, The authors of [21] analyze multi-agent approach for solving multidimensional multi-choice KSP. They solve this by decomposing the problem into sub-problems. Each sub-problem is solved by an agent utilizing negotiation skills, having a central power that evaluates and merges feasible solutions known as the coordination agent. The authors of [22] studies two main problems: how to optimally decide what goods to select in the KSP; how to divide the total revenue among the different agents. The second problem is approached with three types of cooperative games: pessimistic, optimistic, and realistic [22]. A spatial game representation of the knapsack problem has been presented in [23], where game player entities cooperate to maximize gain and reduces costs in order to reach a resolution using bargaining solutions. A similar approach has been applied in works [22,24,25] to maximize the gain. The vehicle routing under chance constraints with stochastic requirements is investigated in [26].

3. Materials and Methods

3.1. MARBLE and SEAs

We developed and tested a prototype of the robot MARBLE, depicted in Figure 1, which can autonomously empty the litter bins [8,27].

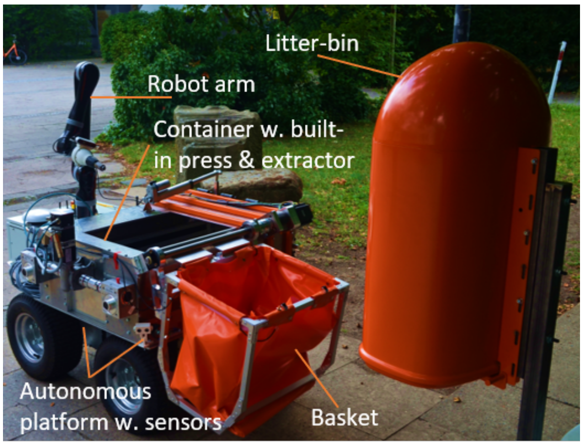
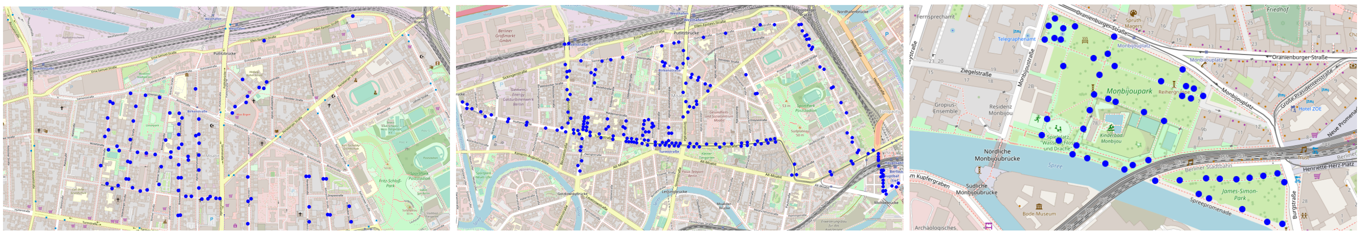


Figure 1. Modular design of the MARBLE.

With the help of various sensors like Lidar, ultrasonic sensors, depth cameras, and GPS, the robot can autonomously navigate through crowded places, formulating paths while avoiding dynamic and static obstacles. The robot arm and the depth camera help in the detection of the litter bin and its keyhole and thereafter in opening it. The garbage is collected in the basket, before being transferred to the container with a built-in press. The basket is designed and programmed to close the lid of the bin with the help of its rotational movement, similar to the motion of human arms. Due to its limited capacity of garbage collection, the robot has a garbage extractor function that assists in transferring the garbage to an external system, thereafter being able to empty other litter bins.

To achieve the goals of the service in various SEAs the robots will be working in a fleet with a conceptual assistance vehicle known as the mothership. The mothership assists the robots in handing over the compressed garbage after the maximum garbage storing capacity has been reached. The SEAs can have varying characteristics. They can contain smart litter bins with available information about the garbage to be emptied [16], or conventional litter bins with unknown filling levels. The focused characteristics in this paper are the area, density of the litter bins, and the filling rate of these in the respective SEA. Three real-world SEAs are depicted in Figure 2, with litter bins positions marked as blue dots.



(**a**) (**b**) (**c**)

Figure 2. Service event areas under consideration for autonomous emptying of litter bins. (**a**) SEA Alt-Moabit with 87 litter bins; (**b**) SEA Alt-Moabit with 214 litter bins; (**c**) SEA Monbijou Park.

3.2. Minimum Energy Utilization Strategy for Varying Service Event Areas

Figure 3 defines the approach of incorporating the Minimum Energy Utilization Strategy for the three SEAs mentioned. It takes the global positions of the marked litter bins into consideration and provides information if these are conventional or smart litter bins, along with their possible filling level. For every SEA we have incorporated a fixed number of three MARBLEs and a mothership for providing the service. The characteristics of the SEAs, such as the area, the density of the litter bins, the type of litter bins, and the filling level rate of the litter bins, are the characteristics incorporated in the Minimum Energy Utilization Strategy along with the number of robots and mothership. The developed VRPSA and standard KSP and game-theory-based KSP are used as route-planning algorithms for comparing the energy consumption due to the different SEAs' characteristics. Apart from the three mentioned SEAs in this section, we have also tested other dummy SEAs for validating our approach for utilization for other SEAs in future. The approach has been further explained in the next section.

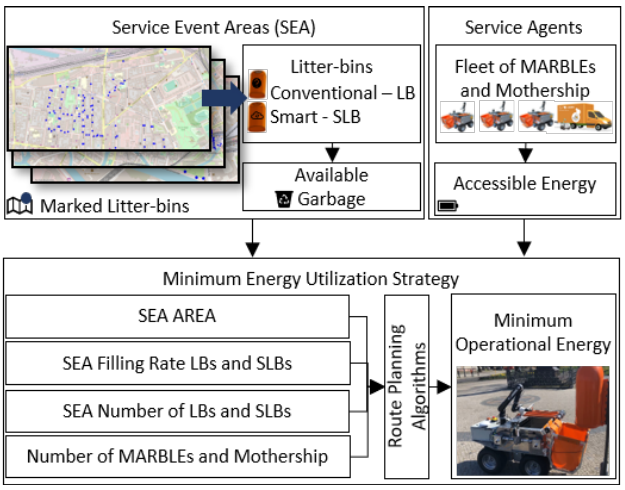


Figure 3. Incorporation of area-directed optimized energy utilization strategy for the fleet of MARBLE in various Service Event Areas.

3.3. Determining Algorithms for Least Energy Consumption

Berlin's municipality is responsible for the city's waste collection. Automation with robots can help in reducing the need for workers the municipality is facing. A fleet of robots working together could make covering the large are of Berlin more efficient. In order to make this process sustainable but also efficient we optimize for both energy consumption and time. Currently, the municipality cannot determine a priori the quantity of garbage in a litter bin, and sometimes when collecting the rubbish, the litter bin is empty leading to unnecessary waste of energy, distance, and time. To address this issue, incorporating information from the surrounding environment or utilizing smart litter bins (SLBs) can lead to efficient route planning, helping to maximize resources and simplify the process. The MARBLE fleet can be highly productive if its route is optimized. By optimizing the route, the robots could reduce energy consumption, prolong battery life, and increase the overall efficiency of their operations. Further, the mothership requires collecting waste after a MARBLE reaches its built-in press capacity and/or battery limit. Combinatorial optimization problems focus on finding the optimal solution from a discrete configuration space within a finite set. They have been widely researched over the years but one of the main drawbacks is their complexity, as they belong to the class of NP or NP-hard problems, meaning they can be solved in polynomial time by a nondeterministic Turing machine. Some of the most common problems are the traveling salesman problem (TSP) using a single route for multiple destinations, the vehicle routing problem (VRP) focusing on multiple routes for multiple destinations, and the knapsack problem (KSP) concentrates on multiple destinations with a given volume. MARBLE's container, with built-in press and extractor, has a delimited capacity. The smart litter bins indicate the volume that the bins currently contain, which can help optimize MARBLE's and the mothership's route. In this paper, we built multiple operation areas and compare three algorithms: a vehicle routing problem with simulated annealing meta-heuristic [8]; a game theory knapsack problem with smart litter bins information; a knapsack problem without smart litter bins information. All three of these algorithms are designed to run as offline algorithms to determine the routes of the robots in advance from a central planning authority like the mothership.

3.3.1. Vehicle Routing Problem with Simulated Annealing Meta-Heuristic

A vehicle routing problem with a simulated annealing meta-heuristic has been developed in our previous work [8]. This approach requires an initial solution, a cooling schedule, a number of iterations, and permutation operators. This strategy utilizes a nearest neighbor heuristic as the initial solution, where each robot is iteratively assigned the closest not emptied litter bin for its initial route. The cooling schedule starts with an initial temperature, which decreases over time, and is used to determine the likelihood of randomly accepting a solution [8]. A new solution replaces the current solution either if it represents an improvement or if acceptance is determined by chance based on the temperature. As the number of iterations increases, the cooling schedule reduces the temperature, making it less probable for worse solutions to be accepted. The permutation operators are used to generate a new solution, they used inter- and intra-route operators. The intra-route operators relocate and 2-Opt alter the route of a single MARBLE, while the inter-route operators global relocate and global exchange transforms the route of two MARBLEs by exchanging stops between them. The operators are chosen randomly and they are weighed between 1 and 4, and the solution is accepted if the new cost of the route is lower than the current cost. In order to compare different methods, we will take from the previous work the combination with the best results. This includes the nearest neighbor's initial solution, the temperature of 50 °C, 100,000 iterations, and the weight operators of 4/1/1/4, relocate, 2-Opt, global relocate, and global exchange, respectively [8].

3.3.2. Knapsack Problem

A brute-force knapsack problem is proposed [16] to enhance the operational performance of a fleet of robots through a communication framework. The route planning involves selecting a batch of dustbins, solving the TSP for the batch, and accumulating distances and filling levels. It reports the filling levels to a central entity for better route planning, the data can be communicated with the help of a Long Range Wide Area Network (LoRaWAN) [28]. The paper presents promising results with lower energy consumption, fewer round trips, and increased capacity utilization compared to a baseline approach. This paper will use the same knapsack approach but with different input parameters for the objective, as discussed in the next sections.

3.3.3. Dynamic Knapsack Problem and Game Theory

The dynamic game theory knapsack problem aims to optimize the routes of the MARBLE robot fleet using the information retrieved from the smart litter bins, optimizing the capacity of the robot's built-in press, and minimizing the mothership's route. The knapsack problem (KSP) is a combinatorial optimization problem correlated to capacity constraints; it focuses on optimizing the weight of items in a sack with a given capacity [23]. MARBLE seeks to use the SLB information to optimize the capacity usage of the built-in press and extractor. Thus, minimizing the press energy consumption and reducing the mothership's interaction with MARBLE. The MARBLE robot route is computed as shown in Algorithm 1. It is performed individually for every member of the fleet. The knapsack function uses a dynamic programming approach to calculate a route for MARBLE with MARBLE's capacity C and the set of litter bins value V (calculated with Equation (4)) and filling levels $w \in [0, 1, \ldots, 99, 100]$, representing the integer percentage of capacity of the litter bin that is filled. For smart litter bins, the sensor provides the actual filling level as data. When not using smart litter bins, the fill level is estimated as 50%. An empty MARBLE can carry two full litter bins, which is equivalent to a fill-level of 200%. The starting point s is the point at which MARBLE starts a new route. This is either the initial starting point where MARBLE is positioned by the mothership or the endpoint of the last route MARBLE took until using its full capacity and being emptied by the mothership. The algorithm maximizes the path's value v, optimizing weight w and taking into consideration capacity C. The mothership optimizes its path by hierarchically choosing from the closest to the farthest point of robots that need assistance.

Algorithm 1: Dynamic programming knapsack problem solver

```
Data: capacity C, values V, filling levels w, set of unemptied litter bins S, starting
      point index s
Result: route of dustbins to empty
// initialization
 v ← V_s // s-th row of V
 n ← |S|
 k ← n ∗ n matrix
while v do
    for i = 0, i < n, i + + do
        for j = 0, j < C, j + + do
            if i = 0 & j = 0 then
                k[i][j] ← 0
            end
            else if w[i − 1] <= j then
                k[i][j] ←
                    max(v[i − 1] + k[i − 1][j − w[i − 1]], k[i − 1][j])
            else
                k[i][j] ← k[i − 1][j]
        end
        k_route ← k[i][j]
    end
    route ← k_route
end
 return route
```

A cooperative game is a game theory framework in which participants form coalitions and collaborate to achieve collective objectives by distributing benefits or costs among themselves [29]. In scenarios where a litter bin is situated at the intersection of two distinct paths, a collaborative game is employed to resolve the resulting conflict. Illustrated in Algorithm 2, this situation arises when a point o is shared by paths a and b. To address this, a multi-step process is undertaken. First, the paths are reevaluated using a knapsack approach, excluding point o. This yields recalibrated paths denoted as a_{new} and b_{new}, in addition to the initial paths, a and b. Subsequently, a cooperative game is initiated, aiming to optimize social welfare by maximizing overall utility. This cooperative game involves the comparison between the recalibrated paths (a_{new} and b_{new}) and the original paths (a and b), choosing the paths that minimize the joint route cost. The iterative process continues until there are no overlapping points like o within both paths a and b, thus effectively resolving conflicts and achieving an optimized route configuration.

Algorithm 2: Game theory: cooperative game

Data: space S, point o, path a, path b
Result: $a_{\text{new}}, b_{\text{new}} \in S$ w.r.t. a, b and o
// initialization
while $\exists o \in]a, b[$ **do**
  $a_{\text{new}}, b_{\text{new}} \leftarrow knapsack(a, b) \notin o$
  $a_{\text{new}}, b_{\text{new}} \leftarrow coo_{\text{game}}(a, b, a_{\text{new}}, b_{\text{new}})$
end

3.4. Calculating Energy Consumption

The energy cost function in Equation (1) was defined in our previous work [8]. The first term reflects MARBLE's cost, and the second is the mothership's cost. The energy expenditure of MARBLE is α and for mothership is β. The energy consumed during the litter bin opening process is presented as e_{open}, and $\text{bins}(i)$ is the number of litter bins emptied by robot i. The energy consumed while driving is defined by $\alpha \cdot \text{len}(i)$. With $\text{len}(i)$ being the length of the route driven by robot i. These give the total cost of the MARBLE robots and the cost of the mothership, defining the energy cost of a route s.

$$f(s) = \left(\sum_{i=0}^{r} \alpha \cdot \text{len}(i) + e_{\text{open}} \cdot \text{bins}(i) \right) + \beta \cdot \text{len}(\text{path}_{\text{MS}}(s)) \quad (1)$$

We updated the values of these components to be more in line with our prototype. $\alpha = 122\,\frac{\text{Wh}}{\text{km}}$ and $e_{\text{open}} = 18.7$ Wh represent the energy required to move MARBLE 1 km and open one bin respectively. The energy required to move the mothership for 1 km is estimated as $\beta = 270\,\frac{\text{Wh}}{\text{km}}$. In addition, we added $\xi = 92$ W to include the energy used by the onboard components of MARBLE (excluding the built-in press) with $t(i)$ representing the time taken for route completion. This updated cost function can be seen in Equation (2).

$$f(s) = \left(\sum_{i=0}^{r} \alpha \cdot \text{len}(i) + e_{\text{open}} \cdot \text{bins}(i) + \xi \cdot t(i) \right) + \beta \cdot \text{len}(\text{path}_{\text{MS}}(s)) \quad (2)$$

The values for α, e_{open} and ξ were obtained through the measurement of their processes on the robot prototype. β is an assumption made from the concept design of the mothership. To keep the results from [8] comparable, we recalculated the cost for the resulting routes with the new cost function.

3.4.1. Service Event Area

The municipality works in areas that are composed of different factors. The generalization of an SEA to measure the efficiency of MARBLE can be defined with many characteristics but we focus on three specifically: the area, the number of litter bins, and the level of filling of the litter bins. We made this selection for two reasons: Firstly, we wanted to be able to have a reference to adapt the solution to different SEAs quickly. Thus, we chose geographic parameters that were easy to gather to characterize the SEA by. Secondly, the Berlin municipality does not have data for the filling level of its litter bins; thus, we wanted to be able to adapt to different variations in this distribution. The density of an operating area is defined as the number of litter bin divided by the area ($\frac{N_{LB}}{A}$), and the filling level of the litter bin is defined by the standard deviation σ of its distribution. We decided to test the route planning solutions in the areas of 0.5, 1, 2.5, and 5 km^2. The number of litter bins in the testing areas N_{LB} varies from 50 to 450 with an increment of 50 litter bins. The fill level of the litter bins is modeled as a normal distribution. The mean is always set to 50% with $\sigma = [10\%, 25\%, 40\%]$ used to quantify the variability of a set of litter bins to measure the performance of the algorithms. Thus, the different distributions are represented by their standard deviation.

3.4.2. Smart litter bins

In the smart litter bin study [16], the filling level of each litter bin is assigned with its respective weight, while the maximum capacity of the robot serves as the upper weight boundary. The objective function of each dustbin, denoted as R_{ks}, is determined through Equation (3). In this formulation, f represents the dustbin's filling level, d_{db} signifies the distance to a reference dustbin, adjusted by a factor K_d, and the ϵ set to 10×10^{-5} to avert division by zero concerns.

$$R_{ks} = \frac{(1 \times 10^5)}{(f_{max} - f + \epsilon + K_d d_{db})} \tag{3}$$

In this work, we changed the objective function to the value V_{ij} in Equation (4). It takes the smart litter bin filling information w from bins i and j divided by the overall filling of all the smart litter bins w. This is normalized with the distance between SLBs i and j. Otherwise, the optimization would depend solely on the capacity of the built-in press and the distance cost would increase severely, making the route an unviable option. The revenue of each dustbin is defined in the Formula 3, based on dustbin's filling level and distance.

$$V_{ij} = \frac{(w_i + w_j)/(\sum w)}{(d_{ij})} \tag{4}$$

When testing an algorithm with smart litter bins, we no longer assume them to be filled to 50%. Instead, we can use the actual fill level which is modeled as described in Section 3.4.1.

4. Results

This section presents a thorough evaluation of three distinct route-planning algorithms (compared in Table 1) within various SEAs. The first algorithm is a dynamic game-theory-based knapsack problem integrating smart litter bin data (Algorithm I); the second is a dynamic knapsack problem without smart litter bin information (Algorithm II) based on [16]; the third is a vehicle routing problem enhanced with simulated annealing meta-heuristic from [8] (Algorithm III). In Algorithms II and III, there is a uniform assumption that all the litter bins are filled to a mean of 50%, which is based on the information provided by the municipality of Berlin. The testing encompasses three MARBLE robots operating in the designated SEAs described in Section 3.2, and Algorithm-3's outcomes are drawn from three separate trials due to its inherent non-deterministic nature. The performance evaluation is conducted using the objective defined in Equation (2). The foundation of this study rests on the premise that the optimal solution for waste collection across different SEAs requires tailored strategies. Thus, the following sections of the paper delve into examining their applicability within the broader context of both general route planning and the specific operational landscape of the MARBLE robots and their service operations. All results in this section were obtained in simulations run on a Lenovo Thinkpad E15 with an Intel i7-1165G7 CPU and 16 GB of RAM. The implementation was performed in Python 3.6 with the SEAs as custom files in the format of the TSP Library and usage of the numpy library for list-based operations [30].

4.1. Results for Application Areas in Berlin

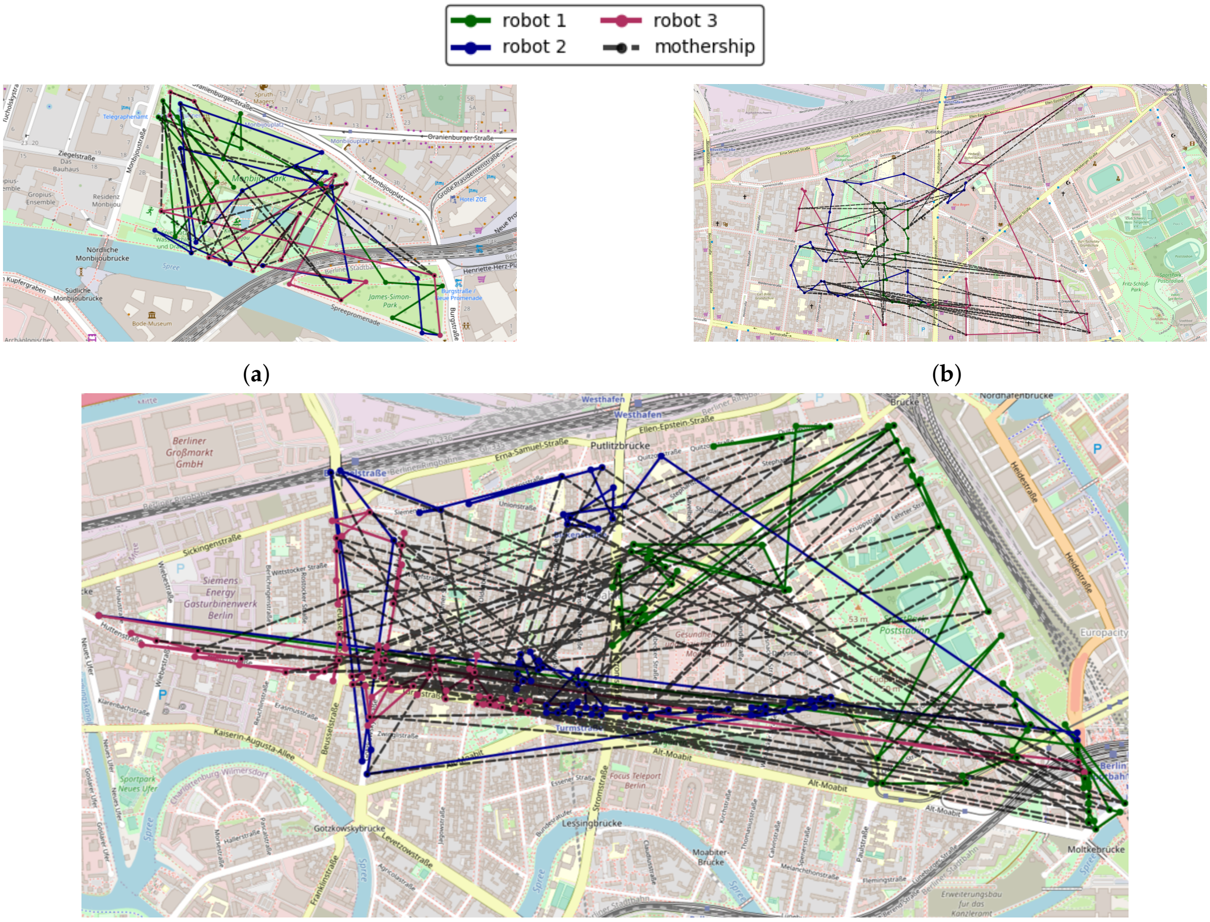
The operational scenarios chosen for evaluation Monbijoupark, Moabit87, and Moabit214 represent three distinct event areas where the Berlin municipality is actively engaged.

These SEAs were tested with the three algorithms mentioned in Section 4. The results presented in Table 1 reflect the route planning optimization of three MARBLEs. In Monbijoupark, Algorithm I has the best energy performance. Algorithms II and III have similar performances; their results against the best performer are +10.2% and +9.7%, respectively. In contrast, in Moabit87 and Moabit214, Algorithm III has the best performance. Algorithm I has a considerable gap with +46.5% and +20.7%, and Algorithm II reduces the difference with +20.7% and +2.6%. Algorithm I aims to minimize the number of times the robots have to be emptied. This results in longer trips for the robots while reducing the amount of stops for the mothership. In small areas, such as the Monbijoupark, this leads to an improvement in performance in regard to the energy consumption. As the area grows, the distance driven becomes more important, allowing Algorithm 3 to be more efficient than Algorithm I.

Table 1. Energy consumption for Service Event Areas in Berlin

Scenario	Area km^2	Num. LB	Fill Rate σ	Alg.	Energy kWh	Computation Time s
Monbijoupark	0.1	51	30	I	4.2	0.13
				II	4.6	0.15
				III	4.6	8.92
Moabit87	1.7	87	30	I	18.9	0.45
				II	17.4	0.52
				III	12.9	13.45
Moabit214	5.5	214	30	I	99.0	4.98
				II	84.1	5.47
				III	82.0	24.31

The optimal route for all three real-world SEAs can be seen in Figure 4. The Monbijoupark was solved with Algorithm I, Moabit-87 with Algorithm II and Moabit-220 was solved with Algorithm III. The different effects of the algorithms can be seen. In Figure 4a and Figure 4b, we can see that fewer mothership stops are favored; however, a clear clustering of litter bins can be seen in Figure 4c, which causes the mothership to travel more often between the different robots while minimizing the individual robot path lengths.



(c)

Figure 4. Best solutions for real-world SEAs. (**a**) Monbijoupark; (**b**) Moabit87; (**c**) Moabit214.

4.2. General Results

Earlier, the algorithms underwent testing within functional areas pertinent to Berlin's municipality. This section extends the scope to generalize the SEAs, with the intention of exploring algorithm performance across diverse operational domains. This exploration is guided by specific parameters, including area size, number of litter bins, and the fill rate of these litter bins. The Service Event Areas in this section were created through a randomized allocation of a specified number of dustbins within predefined geographical area.

The distribution of the energy cost between the MARBLEs and the mothership is dependent on the algorithm used. The energy cost is separated between MARBLEs and mothership with a filling level $\sigma = 25\%$, demonstrated in Figure 5. Algorithms I and II explain that, in a 0.5 km^2 area, the MARBLEs' cost is $\approx$66.5% less than the mothership cost, and Algorithm III has the same cost for both. For an area of 1 km^2, the algorithms have different trends. Algorithms I and II double the costs of the mothership by 46% against the MARBLEs, and in Algorithm III the cost is reversed, having a reduction of 54% in the route of the mothership.

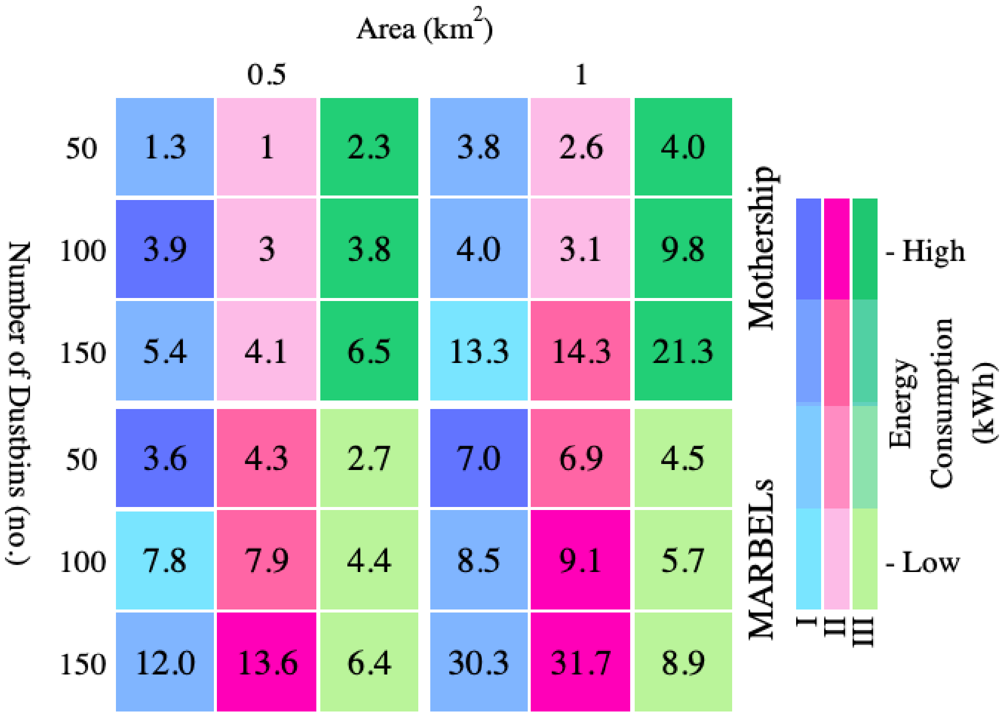


Figure 5. Energy consumption of MARBLEs and mothership.

Figure 6 offers a detailed examination of the performance of the three distinct algorithms, under varying conditions encompassing different quantities of dustbins (ranging from 50 to 450), varying geographical areas (spanning from 0.5 to 5 km^2), and litter bin fill rate as $\sigma = 10, 25, 40$. The primary metric of evaluation is energy consumption, quantified in kilowatt-hours (kW h) as the energy consumption of three MARBLEs and a mothership. In the analysis, it becomes evident that Algorithm I consistently records higher energy consumption when compared to Algorithms II and III across the entire spectrum of scenarios. Notably, Algorithm III consistently outperforms the other two algorithms by achieving the lowest energy consumption levels, irrespective of the number of dustbins or the size of the area. Furthermore, a distinct trend emerges: as the number of dustbins and the size of the area both increase, the energy usage reported by Algorithm III proportionally diminishes.

Table 2 presents a comprehensive summary of algorithm applicability in various Service Event Areas (SEAs). The table is organized into distinct parameter ranges that optimize the performance of Algorithms I, II, and III. Algorithm I, designed for the game theory knapsack problem with access to SLBs information, excels in minimizing energy consumption within SEAs spanning an area of less than 0.5 km^2, containing fewer than 50 litter bins, and attaining a fill rate in the range of 20–40%. Algorithm II, the knapsack problem without smart litter bins (SLB) information, demonstrates superior routing efficiency when operating within SEAs covering an area of less than 0.5 km^2, involving fewer than 50 litter bins, and with a fill rate of approximately 10%. Algorithm III, optimized for solving the vehicle routing problems using simulated annealing, exhibits versatility across a broader spectrum of SEAs. It operates effectively within areas ranging from 0.5 to 5 km^2, accommodating 50–450 litter bins and a filling range between 10 and 40%. These findings provide valuable insights into the algorithm-selection process when addressing real-world environmental optimization challenges in various SEAs.

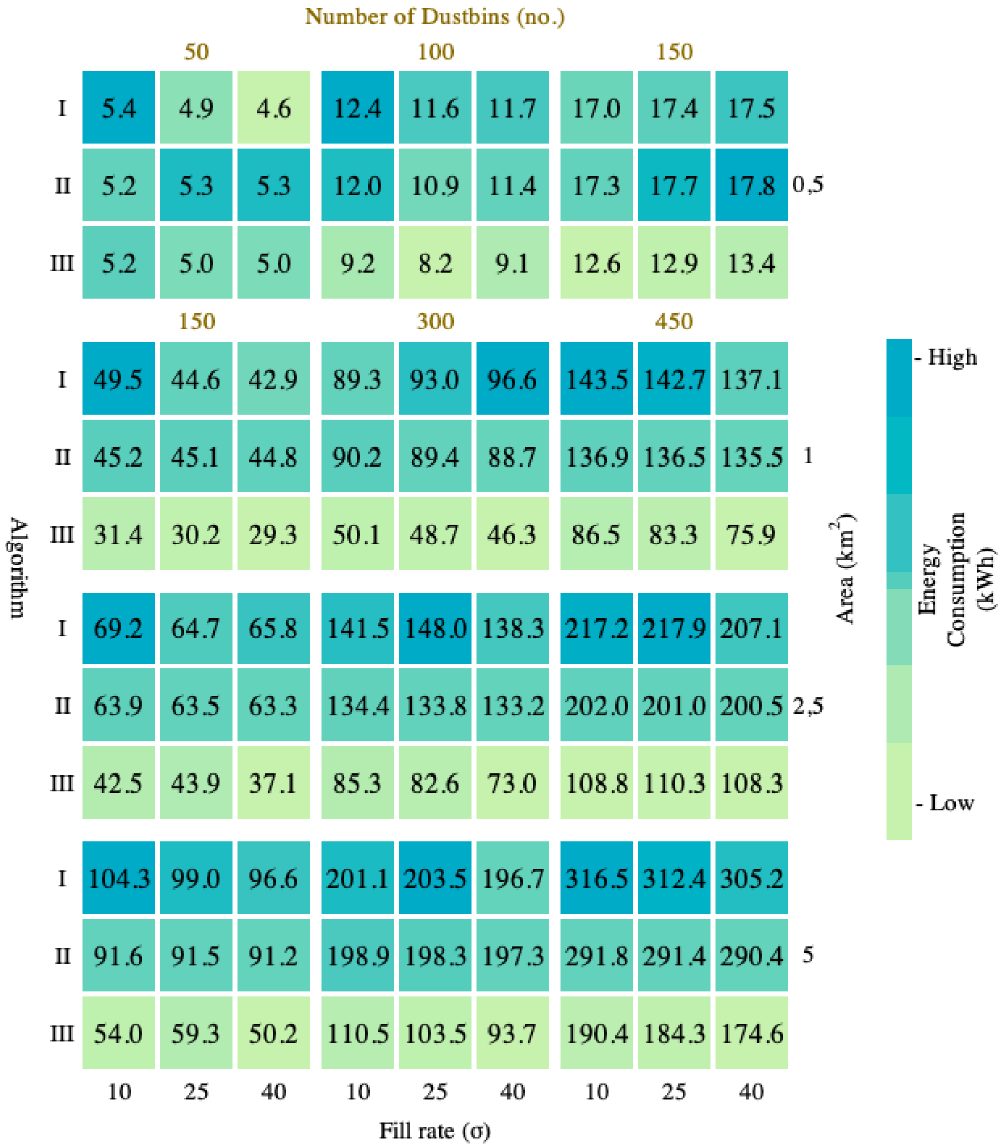


Figure 6. Energy consumption in Service Event Areas.

Table 2. Service Event Areas parameters abstraction for route planning methodologies.

Algorithm	Area A km^2	Num. LB N_{LB}	Fill Rate σ %
I	$A \leq 0.5$	$N_{LB} \leq 50$	$25 \leq \sigma \leq 40$
II	$A \leq 0.5$	$N_{LB} \leq 50$	$\sigma \approx 10$
III	$0.5 \leq A \leq 5$	$50 \leq N_{LB} \leq 450$	$10 \leq \sigma \leq 40$

5. Conclusions and Outlook

This work provides an optimization solution in terms of task allocation to robots in varying SEA with diverse characteristics, such as the area, the number of litter bins, and the litter bin fill rate. We tested and integrated three route-optimization methodologies through simulating the approach for a fleet of urban service robots, MARBLE, and a mothership, that autonomously empty the litter bins on the streets of Berlin. The most promising algorithm for a solution that minimized the overall cost was deducted with the help of numerous experiments. The optimal algorithm, determined through experiments, employs game theory's knapsack problem by integrating smart litter bin data. This approach notably reduces energy consumption and reduces the energy consumption of the mothership. The smart litter bin strategy demonstrates a 9–10% energy reduction. For smaller SEAs, the knapsack problem with or without smart litter bin data is suitable, depending on filling rate. When an area is $x \leq 0.5\,\text{km}^2$ and there are $y \leq 50$ litter bins, the knapsack problem without SLBs information and the game theory knapsack problem with SLBs information are best suitors. Simulated annealing vehicle routing excels in medium-sized SEAs (0.5–5 km^2) with 50–450 bins and 10–40% filling, offering a reduced-energy-use solution.

Limitations and Outlook

The algorithms used in this paper have certain limitations, such as the restriction to simulate real-world scenarios with fidelity. A constraint lies in their treatment of scenario paths. Rather than capturing the intricate trajectories scenarios may take, these algorithms simplify the paths by representing them as linear connections. This simplification compromises the ability to closely mirror the complexities inherent in reality. In addition, approaches such as a vehicle routing problem with simultaneous pickup and delivery and a particle swarm optimization in [31] can be investigated to further improve the performance of the mothership's route. The game theory knapsack problem considers that all the litter bins are smart. The MARBLEs' routes are planned even if the litter bins are not 100% full. The automatic garbage fill alerting system [32] would notify a central system its status when it reaches its capacity, and a swarm of robots would collect the garbage. This research focused on stable characteristics that do not undergo changes over time. Specifically, it examined elements such as the unvarying area and fixed quantity of litter bins within a designated region. The constant filling level characteristics for the litter bins, given the absence of sensors to detect fluctuations, was also assumed. Including a more dynamic estimation of the fill levels could yield an improvement in route planning quality. Incorporating other complex characteristics of the SEAs such as pedestrian densities can also be incorporated in future research, as this would provide a more dynamic approach in terms of real-time-based route planning for varying pedestrian densities [33]. Also, the position of the litter bins as compared to other neighboring litter bins (for example, with a radius of 10 m) can also play an influential role on the filling rates. The approach can be used for various other urban service robots like delivery robots, street-cleaning robots, snow-removal robots, and security robots, as for them the SEAs are ever-changing as they are for MARBLEs. The results obtained in this work are only from simulation data since there is only one working prototype of MARBLE at this point. To validate our results in future research, it is necessary to utilize real-world testing with multiple robots when possible.

Author Contributions: Conceptualization, V.B.J., A.G. and T.F.U.; Methodology, A.G.; Software, V.B.J., A.G. and T.F.U.; Validation, V.B.J., A.G. and T.F.U.; Formal analysis, V.B.J., A.G. and T.F.U.; Investigation, V.B.J., A.G. and T.F.U.; Resources, V.B.J. and A.G.; Data Curation, V.B.J., A.G. and T.F.U.; writing—original draft preparation, V.B.J. and A.G.; writing—review and editing, V.B.J., A.G., T.F.U. and D.G.; visualization, V.B.J., A.G. and T.F.U.; supervision, A.G. and D.G.; project administration, A.G. and D.G.; funding acquisition, D.G. All authors have read and agreed to the published version of the manuscript.

Funding: Project MARBLE is funded within the Berlin Program for Sustainable Development—BENE sponsored by the European Regional Development Fund (#1247-B5-O). We also acknowledge support by the German Research Foundation and the Open Access Publication Fund of TU Berlin.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

SEA	Service Event Area
MARBLE	Mobile Autonomous Robot for Litter Emptying
VRP	vehicle routing problem
VRPSA	vehicle routing problem with simulated annealing
KSP	knapsack problem
TSP	traveling salesman problem
LB	litter bin
SLB	smart litter bin
SDGs	Sustainable Development Goals
TU	Technische Universität (University of Technology)
LoRaWAN	Long Range Wide Area Network
GNSSs	global navigation satellite systems
UAVs	unmanned aerial vehicles

References

1. Gonzalez-Aguirre, J.A.; Osorio-Oliveros, R.; Rodríguez-Hernández, K.L.; Lizárraga-Iturralde, J.; Morales Menendez, R.; Ramírez-Mendoza, R.A.; Ramírez-Moreno, M.A.; Lozoya-Santos, J.d.J. Service robots: Trends and technology. *Appl. Sci.* **2021**, *11*, 10702. [CrossRef]
2. *ISO 8373*; Robots and Robotic Devices—Vocabulary. Technical Report ISO 8373. ISO: Geneve, Switzerland, 2012.
3. Mai, V.; Vanderborght, B.; Haidegger, T.; Khamis, A.; Bhargava, N.; Boesl, D.B.; Gabriels, K.; Jacobs, A.; Moon, A.; Murphy, R.; et al. The Role of Robotics in Achieving the United Nations Sustainable Development Goals—The Experts' Meeting at the 2021 IEEE/RSJ IROS Workshop [Industry Activities]. *IEEE Robot. Autom. Mag.* **2022**, *29*, 92–107. [CrossRef]
4. Grosinger, J. Fostering sustainable consumption through the development of proactive human-centered robot systems. In Proceedings of the IEEE IROS 2021 Workshop on the Role Robot, Achieving UN's Sustainable Development Goals, Prague, Czech Republic, 27 September–1 October 2021; p. 1.
5. Kronmueller, M.; Fielbaum, A.; Alonso-Mora, J. On-demand grocery delivery from multiple local stores with autonomous robots. In Proceedings of the 2021 International Symposium on Multi-Robot and Multi-Agent Systems (MRS), Cambridge, UK, 4–5 November 2021; pp. 29–37.
6. Roy, D.; Maitra, M.; Bhattacharya, S. Exploration of multiple unknown areas by swarm of robots utilizing virtual-region-based splitting and merging technique. *IEEE Trans. Autom. Sci. Eng.* **2021**, *19*, 3459–3470. [CrossRef]
7. Causa, F.; Fasano, G.; Grassi, M. Multi-UAV path planning for autonomous missions in mixed GNSS coverage scenarios. *Sensors* **2018**, *18*, 4188. [CrossRef] [PubMed]
8. Gupta, A.; van der Schoor, M.J.; Bräutigam, J.; Bladinieres Justo, V.; Umland, T.F.; Göhlich, D. Autonomous Service Robots for Urban Waste Management-Multiagent Route Planning and Cooperative Operation. *IEEE Robot. Autom. Lett.* **2022**, *7*, 8972–8979. [CrossRef]
9. Poeting, M.; Schaudt, S.; Clausen, U. A comprehensive case study in last-mile delivery concepts for parcel robots. In Proceedings of the 2019 Winter Simulation Conference (WSC), National Harbor, MD, USA, 8–11 December 2019; pp. 1779–1788.
10. Sembiring, N.; Sipayung, R.I. Developing multi-agent systems using agent-based and geographic information system simulation. In Proceedings of the 2020 3rd International Conference on Mechanical, Electronics, Computer, and Industrial Technology (MECnIT), Medan, Indonesia, 25–27 June 2020; pp. 114–117.
11. Mahmud, M.S.A.; Abidin, M.S.Z.; Buyamin, S.; Emmanuel, A.A.; Hasan, H.S. Multi-objective route planning for underwater cleaning robot in water reservoir tank. *J. Intell. Robot. Syst.* **2021**, *101*, 1–16. [CrossRef]
12. Tomitagawa, K.; Anuntachai, A.; Chotipant, S.; Wongwirat, O.; Kuchii, S. Performance Measurement of Energy Optimal Path Finding for Waste Collection Robot Using ACO Algorithm. *IEEE Access* **2022**, *10*, 117261–117272. [CrossRef]
13. Ahmad, S.; Jamil, F.; Iqbal, N.; Kim, D. Optimal route recommendation for waste carrier vehicles for efficient waste collection: A step forward towards sustainable cities. *IEEE Access* **2020**, *8*, 77875–77887. [CrossRef]
14. Ghahramani, M.; Zhou, M.; Molter, A.; Pilla, F. IoT-based route recommendation for an intelligent waste management system. *IEEE Internet Things J.* **2021**, *9*, 11883–11892. [CrossRef]
15. Bräutigam, J.; Gupta, A.; Göhlich, D. Simulated Annealing-based Energy Efficient Route Planning for Urban Service Robots. In Proceedings of the 2022 26th International Conference on Methods and Models in Automation and Robotics (MMAR), Miedzyzdroje, Poland, 22–25 August 2022; pp. 294–299.
16. Gupta, A.; Kremer, P.; Park, S.; Göhlich, D. Intelligent route planning for autonomous service robots using communicating smart dustbins. In Proceedings of the ICC 2022-IEEE International Conference on Communications, Gangnam-gu, Republic of Korea, 16–20 May 2022; pp. 1202–1207.
17. Borradaile, G.; Heeringa, B.; Wilfong, G. The 1-neighbour knapsack problem. In Proceedings of the International Workshop on Combinatorial Algorithms, Victoria, BC, Canada, 20–22 July 2011; pp. 71–84.

18. Nicosia, G.; Pacifici, A.; Pferschy, U. On Multi-Agent Knapsack Problems. In Proceedings of the CTW, Paris, France, 2–4 June 2009; pp. 44–47.
19. Shrestha, P.; Joshi, B. Multi-agent based heterogeneous power management system. *Int. J. Innov. Comput. Inf. Control* **2021**, *17*, 245–257.
20. Dönmez, E.; Kocamaz, A.F. Multi target task distribution and path planning for multi-agents. In Proceedings of the 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), Malatya, Turkey, 28–30 September 2018; pp. 1–7.
21. Htiouech, S.; Alzaidi, A. Smart agents for the multidimensional multi-choice knapsack problem. *Int. J. Comput. Appl.* **2017**, *174*, 5–9. [CrossRef]
22. Arribillaga, R.P.; Bergantiños, G. Cooperative and axiomatic approaches to the knapsack allocation problem. *Ann. Oper. Res.* **2021**, *318*, 805–830. [CrossRef]
23. Wong, K.K.L. Bridging game theory and the knapsack problem: A theoretical formulation. *J. Eng. Math.* **2015**, *91*, 177–192. [CrossRef]
24. Benabbou, N.; Perny, P. Solving multi-agent knapsack problems using incremental approval voting. In Proceedings of the Twenty-second European Conference on Artificial Intelligence, Amsterdam, The Netherlands, 29 August–2 September 2016; pp. 1318–1326.
25. Lau, H.C.; Zhang, L. Task allocation via multi-agent coalition formation: Taxonomy, algorithms and complexity. In Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence, Sacramento, CA, USA, 3–5 November 2003; pp. 346–350. [CrossRef]
26. Noorizadegan, M.; Chen, B. Vehicle routing with probabilistic capacity constraints. *Eur. J. Oper. Res.* **2018**, *270*, 544–555. [CrossRef]
27. Göhlich, D.; Syré, A.M.; van der Schoor, M.J.; Jefferies, D.; Grahle, A.; Heide, L. Design Methodologies for Sustainable Mobility Systems. In *Design Methodology for Future Products*; Krause, D., Heyden, E., Eds.; Springer: Berlin/Heidelberg, Germany, 2022; pp. 123–144. [CrossRef]
28. SEMTECH. What Is LoRa. Available online: https://www.semtech.com/lora/what-is-lora (accessed on 13 February 2023).
29. Serrano, R. Cooperative games. In *Complex Social and Behavioral Systems: Game Theory and Agent-Based Models*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 49–60.
30. Reinhelt, G. TSPLIB: A Library of Sample Instances for the TSP (and Related Problems) from Various Sources and of Various Types. 2014. Available online: http://comopt.ifi.uniheidelberg.de/software/TSPLIB95 (accessed on 18 October 2023).
31. Ai, T.J.; Kachitvichyanukul, V. A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. *Comput. Oper. Res.* **2009**, *36*, 1693–1702. [CrossRef]
32. Patel, D.; Kulkarni, A.; Udar, H.; Sharma, S. Smart Dustbins for Smart Cities. *Int. J. Trend Sci. Res. Dev.* **2019**, *3*, 1828–1831. [CrossRef]
33. Mokhtari, K.; Ayub, A.; Surendran, V.; Wagner, A.R. Pedestrian density based path recognition and risk prediction for autonomous vehicles. In Proceedings of the 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Naples, Italy, 31 August–4 September 2020; pp. 517–524.