A Cloud-based Dynamic Waste Management System for Smart Cities

Conference Paper · November 2016

DOI: 10.1145/3001913.3006629

2 authors:

Sadia Sharmin
Bangladesh University of Engineering and Technology

Sikder Tahsin Al-Amin
University of Houston

A Cloud-based Dynamic Waste Management System for Smart Cities

Sadia Sharmin
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka-1000, Bangladesh
sadiasharmin@cse.buet.ac.bd

Sikder Tahsin Al-Amin
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka-1000, Bangladesh
stahsin.cse@gmail.com

ABSTRACT

A smart city is a vision to adopt multiple information and communication technology (ICT) solutions in the management of public affairs. Waste management problem is acute in the cities and urban areas now a days. Number of trucks roaming around, collecting waste at any time, excessive manpower requirement and inefficient monitoring are some of the difficulties we face with the conventional waste collection approach. The purpose of our work is to introduce a smart and intelligent waste management system that is able to handle the process dynamically and cost effectively. In our approach weight and volume of waste thrown in the waste bins are collected by economical sensors and then sent to cloud server using a micro-controller and GPRS. This data is used to find the waste collection schedule to maximize the collection. Location of vehicles and waste bins are used to find the shortest possible collection route for each truck which is implemented by Ant Colony Optimization(ACO) technique. The system is adaptable to dynamic changes i.e. routes blocked during waste collection process. The whole process can be monitored centrally and thus provide a high quality service to the citizens of a smart city.

Keywords

Smart City, Waste Management, Ant Colony Optimization

1. INTRODUCTION

One of the major environmental problems now a days is solid waste management. With the increase of population, proper management of solid waste becomes more severe for maintaining sustainable environment. Improper management may cause hazards to inhabitants. Specially in municipal areas where waste are disposed often in open dumps and landfills, may create problems to public health as well as the environment. Lack of budget, manpower and proper monitoring system for tracking activities are making this problem more difficult. As the world is now evolving towards smart city, the goal of building a smart city is to improve quality of life by using urban informatics and technology [7]. ICT (Information and Communication Technology) allows city officials to interact directly with the citizens and city infrastructure to enable a better quality of life.

Our aim is to use ICT to enhance quality, performance and interactivity of urban services, and hence provide an efficient and proper waste management system that will overcome the shortcomings of current management system as well as reduce cost and make better utilization of resources. We propose a cloud based system that will find the best route for collecting waste using Ant Colony Optimization(ACO), monitor the waste bin using low-cost sensors, and find the usage pattern of the waste bins so that collection will be maximized. This smart and proficient system is able to take dynamic decisions and handle special situations such as a collection route is blocked and waste volume and number of required trucks exceed capacity etc.

The rest of the paper is organized as follows. In Section 2 we briefly describe some previous works related to waste management. Section 3 illustrates our approach followed by few experimental data in Section 4. Finally we conclude with few directions for future work in Section 5.

2. RELATED WORK

Different waste management systems have been proposed in different countries and regions. For example, in [5], NFC and cloud computing technology is used to develop a web-based system and a mobile app is designed for supporting waste management. They incorporated Radio Frequency Identification (RFID) and mobile app technology for waste management. A pilot test system was launched in an university and the results showed that the RFID-based waste management system improved the recycling rate. However, the waste collection process of this system was very simple, it doesn't handle any special situation like route blocked or availability of vehicle.

In [2], they proposed a system for that equipped with information system and advanced technologies like RFID, Global Positioning System (GPS), General Packet Radio Service (GPRS) and Geographic Information System (GIS) along with camera technology in order to develop the bin and truck intelligent monitoring system. Another system described in [3] which would be able to monitor and management the overall collection process. It provides in time solid waste collection, tracking the vehicle position through

ACM DEV '16 November 17-22, 2016, Nairobi, Kenya

DOI: http://dx.doi.org/10.1145/3001913.3006629

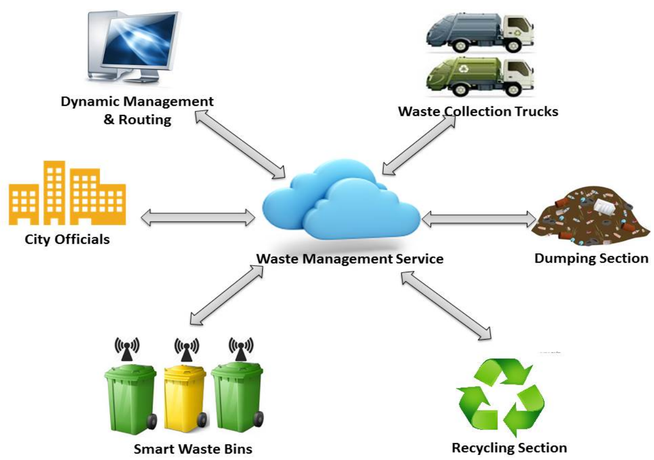

Figure 1: Workflow of the proposed model.

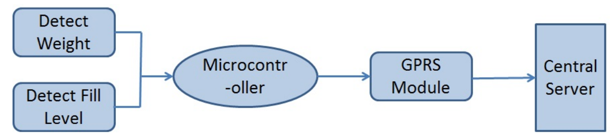

Figure 2: Data collection from sensor and sending it to server.

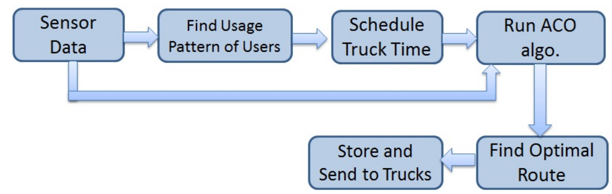

Figure 3: Find optimal route for each truck from sensor data

the GIS database, real time bin status. But one drawback of this work is that, this requires the collecting truck to come near to the waste bin to send data to server.

In [6], they stated that, 85% of the solid waste system cost takes the collection process. Hence, they find the best route for collecting solid waste in cities taking Irbid City in Jordan as an example problem. This work has developed a methodology based on real genetic algorithm. However, their solution was limited to bidirectional paths as they considered distance as a symmetric process.

3. OUR SMART AND DYNAMIC APPROACH

In this work we propose our cloud service based dynamic waste management system, which use real-time data from sensors installed in waste bins and employs Ant Colony Optimization to suggest a near optimal route for waste collection trucks. The model of the proposed system is depicted in Figure 1. The process starts with the data collection from the waste bins. Every waste bin has a unique id. The data is processed and then used to find the optimal routes and sent to trucks. After collection, wastage are divided into dumping and recycling section. The whole process is managed dynamically and can be updated at any time. City officials can also monitor the whole process to make the city a better place for the inhabitants of a smart city. Our system mainly consists of two phases. They are -

1. Collecting and sending data from the waste bins
2. Finding optimal route for waste collection

3.1 Collecting and Sending Data from Waste Bins

This phase mainly focuses on collecting data from waste bins using sensor and sending them to server. Both the weight and filling level of waste bins are sensed using two sensors that are attached with waste bin to collect data. The process can be shown in Figure 2

Load sensors are used to detect the weight of the waste disposed by the user. Load sensor model SEN-10245 is used to detect the weight which can measure upto 50kg. To detect the filling level ultrasonic sensors are used. It operates with the propagation of sound waves. Distance to object is determined via a sound transit time. And it is not affected by color, transparency, or glossiness of a surface which is very much advantageous for sensing waste filling level. Besides, the LED display shows if the container is full to alert the users.

All these data collected are sent for processing which is done using a Microcontroller and GPRS (General Packet Radio Service). The data collected from sensors requires filtering and sorting. These raw data will be formatted into structured data. A central server receives the data and stores all the necessary information for current and future use.

A microcontroller is a small computer on a single integrated circuit, containing a processor, memory, and input/output functions. Arduino Uno microcontroller can be used for testing of collecting and processing sensor data. After formatting the data into a specific format, the controller sends it to central server through GPRS module. GPRS is a data service offered by cell phone carriers and operates in conjunction with voice cell phone services. A GPRS module is connected with a remote server that receives the data. This technology may prove very cost effective and practical as cell phone data providers have most areas covered. This technology eliminates the need for complex radio based telemetry networks.

3.2 Finding Optimal Route for Waste Collection

In this stage, we find the optimal route for waste collection. If the truck can collect all the waste from desired location with shortest possible path, it can save both fuel money and time. Moreover, we analyze usage pattern of different localities to collect maximum waste and allow waste bins more time to fill up. The steps are shown in Figure 3. Hence, this process is further divided into two steps-

i Mining Usage Pattern

ii Finding Optimal Route

Mining Usage Pattern: We analyze data to determine when the waste bins are full or almost full. To find the usage pattern, first we collect the data sent from the waste bins. We sense data from 9.30 am to 4.30 pm with one hour

Sl. No	Waste Bin	Date	Status	09.30am(KG)	10.30am(KG)	11.30am(KG)	12.30pm(KG)	....
1	WB-1	1/7/16	OK	4.30	5.63	6.01	6.99	
2	WB-2	1/7/16	OK	10.56	15.12	16.23	17.25	
3	WB-3	1/7/16	OK	2.03	3.69	5.65	6.01	
4	WB-4	1/7/16	OK	8.20	8.57	9.26	9.99	
...	...							

Figure 4: Data from weight sensor at different times

Data: Data from sensors and the graph $G(V, E)$
Result: Optimal route for each truck
initialization;
for *each truck V* **do**
  Starting node, v_i=Garrade location
  Compute total weight(W) and volume(A) of the nodes
  if $W > Capacity \; || \; A > Capacity$ **then**
    Discard node with minimum W or A
  end
  for *nodes* $v_i, \; v_{i+1}, ...v_n$ **do**
    $ACO()$;
  end
  $F = Total \; KM \; * \; Fuel \; per \; KM$
  if $F \; > \; FuelCapacity$ **then**
    Discard node with minimum W
    $ACO()$;
  end
  Store and send optimal route to trucks.
end

Algorithm 1: Finding optimal routes for each truck

interval and a total of 15 day's data are collected. A sample of collected data from weight sensor is shown in Figure 4. Then we create a matrix where row is represented as the day number and each column is the weight collected(W_i) from the sensor every hour. For each day the total amount of weight(W) is calculated and then for each hour (H_i) we determine the percentage amount using following equation -

$$H_i = \frac{W_i - W_{i-1}}{Total \; Amount(W)} * 100 \tag{1}$$

Then we find the maximum value for each row and represent them as 1 and other values 0. Then the column with maximum number of 1's has the highest usage value, means waste bins are filled most at that time. We can schedule the sensing process between those hours each day and trucks can start collection at that time. Sensing only one time in a day will cost much less than sensing several times in a day. To update or check if the time is changed, we can repeat the finding usage pattern process in every 3 or 4 months.

Finding Optimal Route: Optimal routes for each truck is determined in this stage. Data collected from the sensor is also used here to find the optimal routes. First we need to know the position of the waste bins. Hence, the city is represented as a directed graph $G(V, E)$, where vertex set V is the positions of the waste bins and E is the distance between two waste bins and edges exist if there is a road in between. Here we consider directed graph, because bidirectional distances between waste bins are not necessarily identical. There may exist only one way route from one waste bin to another or both way may not have the same distance due to road network restriction. This makes the routing process more practical and robust. Initially the starting and ending point for each truck will be same as they are parked in garage and dump the collected waste in same place. Each truck have different area for the collection process and therefore, optimal route finding algorithm is run within those locations. The algorithm will give the shortest route for each truck. Therefore, a huge amount of fuel money will be saved. It will also take less time than usual excluding factors like traffic jam or unexpected road blocks.

Ant Colony Optimization (ACO) [1] is a shortest path finding algorithm which was inspired by the behavior of ants. Each ant moves at random for food source from nest. Pheromone is deposited on path to follow. More pheromone on path increases the probability of path being followed and finally the ants arrive at the shortest path. For small number of nodes, shortest path can be found using exhaustive search but for large number of nodes, it will be very hard to find. ACO adapts to change, hence can be used in dynamic application. Moreover, they retain memory of entire ant colony, so less affected by poor initial solution. As finding shortest path for the collection truck is a vital issue and the system is dynamic, we choose ACO to find shortest path for each collection truck. ACO will run for the valid number of nodes(collection points). The ants(trucks) are placed randomly in every nodes, then the ants visits the next nodes with transition probability. Pseudo-random-proportional transition rule can be used as in [4]. After all the ants completed their tour, only the best solution is used to globally update the pheromone. These two processes are followed until convergence. The output will be the shortest path among the nodes.

The Algorithm-1 works as following - first each truck(v_i) calculates the total weight(W) and volume(A) of waste of the locations it will visit. If either one exceeds its capacity, it discards the location with minimum amount weight or volume. The assumption behind this idea is that the container with minimum amount of waste will take more time to fill up than others and we can skip it. We then run the ACO to find the shortest path for the truck. After optimal route is found, the algorithm will check if the fuel needed for total path(F) will exceeds its capacity. We can check this by multiplying fuel need per kilometer with total distance of the route. If this value exceeds fuel capacity, we discard the node with minimum weight and volume and then run the ACO again. The assumption behind this idea is same as before. Then the optimal route is both stored and send to the trucks.

3.2.1 Handling Unexpected Situation During Collection

In order to build a robust system, we have also considered the unexpected situation that may occur during collection of waste, such as, a road may be blocked due to accident, police barricade, construction, repair etc or the collection truck might be damaged or stopped. If a truck encounters such situations while collecting waste, it will try to search for an alternative route. The process can be shown in Algorithm-2.

Algorithm-2 works as follows - If the truck encounters a block or other unexpected situation, the driver sends its cur-

Data: The graph $G(V, E)$ and the set of nodes to be avoided
Result: New optimal route for unvisited nodes
if *route is blocked* **then**
  Driver sends its position to server via GPS
  Calculate length of the remaining path (L_R)
  Starting position, v_b = Current position
  for *nodes* $v_b,\ v_{b+1}, ...v_n$ **do**
    $ACO()$;
  end
  Calculate length of Alternate route(L_A)
  while $L_A < \alpha * L_R$ **do**
    Discard next blocked route
    $ACO()$;
  end
end
if *truck is damaged* **then**
  Driver sends its position to server via GPS
  Identify truck id(T_D)
  if *truck is available in garradge(T_A)* **then**
    Starting position of T_A, V_b= Next bin position of T_D,
    for *nodes* $V_b,\ V_{b+1}, ...V_n$ **do**
      $ACO()$;
    end
  end
  exit
end
Store and send optimal route to trucks.

Algorithm 2: Finding alternate optimal route if current route is blocked

rent position to server via GPS. The server sets the current position as starting position and calculates the length of the remaining original path(L_R). It further determines the unvisited nodes from the stored route by mapping truck id. Then ACO algorithm is performed for the rest of the path and optimal alternate route is gained. Now the total length of the alternate route is calculated (L_A). If this length exceeds L_R by a significant amount(α), then the next blocked route is discarded. Here $1.0 < \alpha < 1.5$ is dependent on the system, which can be any value between 1.0 to 1.5 meaning that the alternate route could be as high as 1.5 times of the original route. The server administrator may vary the value of α considering the situation. After that, the alternate route is stored and send to the truck. If the truck is damaged, driver updates its position like before. Then if truck is available in garradge, its starting position is set as the next visiting position of the damaged truck. Then ACO algorithm is run for the rest of the nodes. The routes are stored and send to the trucks.

4. EXPERIMENTAL RESULTS

We have implemented the main algorithm and run it for a small artificial network. In ACO, parameters α and β determines relative influence of pheromone trail and heuristic information.If $\alpha = 0$, closest nodes are more likely to be selected and if $\beta = 0$, only pheromone amplification is at work means a situation where the route will be suboptimal. $0 < \rho \leq 1$ is pheromone trail evaporation. We tested our algorithm against the route from [6], where they implemented using genetic algorithm. We choose different parameters for α, β and ρ. Then the result was determined from the best simulation. We took different parameter settings for (α, β, ρ) where $\alpha\epsilon(0, 0.5, 1, 1.5, 2, 3)$, $\beta\epsilon(0, 0.5, 1, 2, 3, 4)$ and $\rho\epsilon(0.1, 0.15, 0.25, .35, .4)$. The ACO algorithm managed to find the shortest tour after 2000 cycles with tour length of 6585m for the following parameter settings $\alpha = 1$, $\beta = 2$ and $\rho = 0.1$. Although this route was for undirected graph, we made the graph bidirected and the test result was same. However, changing some bidirectional nodes to unidirectional changes the path length and also takes more cycles to converge.

5. CONCLUSION AND FUTURE WORK

Here we have presented an efficient and smart system for waste collection in smart cities. Using this approach from waste disposal by citizens to collection and dumping, the whole process can be monitored and handled smartly. Making the system dynamic helps to reduce complexities during the process. Complementary mobile applications and social platforms can be embedded with the system to bridge citizens with the cloud services. Moreover incorporating their feedback and recommendations will surely augment our smart city vision. Finding the usage pattern in different localities might also help to maximize waste gain and reduce chaos. This is an ongoing work and we plan to do exhaustive experiments using artificial and real-life scenarios to explore challenging alternatives.

6. REFERENCES

[1] L. Bianchi, L. M. Gambardella, and M. Dorigo. An ant colony optimization approach to the probabilistic traveling salesman problem. 2002.

[2] M. Hannan, M. Arebey, R. Begun, and H. Basri. Radio frequency identification (rfid) and communication technologies for solid waste bin and truck monitoring system. In *Waste Management*, pages 2406–2413, 2011.

[3] M. A. Hannan, M. Arebey, H. Basri, and R. A. Begum. Intelligent solid waste bin monitoring and management system. *Australian Journal of Basic and Applied Sciences*, pages 5314–5319, 2010.

[4] N. V. karadimas, K. Papatzelou, and V. Lomous. Optimal solid waste collection routes identified by the ant colony system algorithm. *Waste Management and Research*, pages 139–147, 2007.

[5] C. K. M. Lee and T. Wu. Design and development waste management system in hong kong. In *Industrial Engineering and Engineering Management (IEEM), 2014 IEEE International Conference on*. IEEE, 2014.

[6] I. von Poser and A. Awad. Optimal routing for solid waste collection in cities by using real genetic algorithm, 2006.

[7] Wikipedia. Smart city — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Smart_city, 2016. [Online; accessed on 27-August-2016].