Improving Bus Schedules and Waste Collection Routes in Practical Smart City Implementations

Jan Dünnweber
Ostbayerische Technische Hochschule Regensburg
jan.duennweber@othr.de

Amitrajit Sarkar
Ara Institute of Canterbury
amit.sarkar@ara.ac.nz

Vimal Kumar Puthiyadath
KPIT Technologies Ltd
Vimal.Puthiyadath@kpit.com

Omkar Barde
KPIT Technologies Ltd
Omkar.Barde@kpit.com

Abstract

Computer-based Improvements of waste collection and public transport procedures are often a part of smart city initiatives. When we envision an ideal bus, it will primarily connect the most crowded bus stops. Similarly, an ideal waste collection vehicle will arrive at every container exactly at the time when it is fully loaded. Beyond doubt, this will reduce traffic and support environmentally friendly intentions like an expansion of waste separation, as it will make more containers manageable. An obvious difficulty of putting that vision into practice is that vehicles cannot always be where they are needed. Knowing the best time for arriving at a certain position is insufficient for finding the optimal route. Therefore, we compare four different approaches to optimized routing: Regensburg, Christchurch, Malaysia and Bangalore. Our analysis shows that the most efficient schedules result from adapting field-tested routes frequently on the basis of sensor measurements and shortcuts resulting from route optimizing computations.

1. Introduction

Even with the latest IoT technology like networked sensors and simulations forecasting the behaviour of the population of megacities, practical implementations of on-demand public transport and smart waste collection still have difficulties in keeping up with the prognosticated improvements. A vehicle that drives obstinately from the most crowded bus stop (respectively, the most heavily filled container) to one with the fill-grade closest to that will obviously need more time in the majority of cases than a vehicle following a fixed plan, since the crowded bus stops and the heavily filled containers are probably positioned far apart from each other.

Finding a smarter route leads us to the classic *vehicle routing problem* (VRP [1]), an instance of the *Travelling Salesman Problem (TSP)* with the added constraint that we need to return to the starting point after visiting a fixed number of points, since the collection vehicle has a limited capacity. Thus, we don't need to find the minimum Hamiltonian circle through all the points but multiple circles forming some kind of clover leaf. However, finding the best route does not only require to consider the distances between the single stops. Busses should leave out stops where no passengers are waiting and waste collectors should skip containers which are filled only to a certain level, i. e. we are dealing with an instance of a dynamic route planning problem, which is also the subject of more recent research [2].

There are $\frac{n!}{2}$ different routes connecting n positions. For comparing all possible routes between only 10 stops, this means 3628800 routes must be analyzed. While a typical bus line might comprise 10 stops, a modern waste collection vehicle can be loaded with $\approx$ 400 containers of 120 liters [3]. 400! is a 882-digit number. Skipping containers with little load means, the vehicle has to pickup an other one, where it usually does not drive to. Thus, solving the dynamic VRP requires to solve a new problem of that size, every time when the fill level measurements are updated. Nowadays, supercomputers can deal with such problem sizes [4]. However, this work presents four projects based on approximate solvers, which can run a standard PC or an on-board computer. Therefore, the presented work is relevant for automating the navigation of autonomous busses and robotic garbage trucks, like the one Volvo started testing in Brussels recently [5].

The rest of this paper is structured as follows: Section 2 explains the *ant colony optimization* (ACO) and shows how Regensburg and Christchurch benefit from ACO and fill level sensing for collecting their waste containers. Section 3 introduces a different approach to minimizing routes be means of a greedy algorithm and shows two more practical implementations: The waste collection in Malaysia and the employee transport service of Bangalore. Section 4 discusses the lessons learned and points out some future perspectives.

2. Intelligent Routing by means of the Ant Colony Optimization (ACO)

The *ant colony optimization* is an approximate solution for TSP instances (ACO [6]). This approach has also been proven suitable for dynamic VRP instances in a simulation, where the road network and the related traffic were taken into account [7]. ACO is a *swarm intelligence* procedure, i. e. not an individual (a simulated ant in the case of ACO) solves the problem but a group. For finding optimal routes, the simulation starts with letting the ants take random paths until they reach their destination. This random walk is optimized iteratively: each ant leaves a pheromone trail behind it which evaporates after a certain number of iterations. In every iteration the pheromone intensity of the shorter paths increases because whenever a simulated ant can choose among multiple paths, it takes the one with the highest pheromone intensity. This means, in higher iterations, the paths are no more randomly chosen but influenced by the most successful ants from preceding iterations. These are the ants whose pheromone trails did not evaporate until their followers reached them, since they were on the shortest paths.

2.1. Application 1: The Collection of Biological Waste in Regensburg

Our work in Regensburg focuses on improving the collection of biological waste [8]. While other approaches toward the computer-aided routing of waste collection vehicles rely only on fill level sensing [9] or only on ACO-based path computations [10], we combined both ideas.

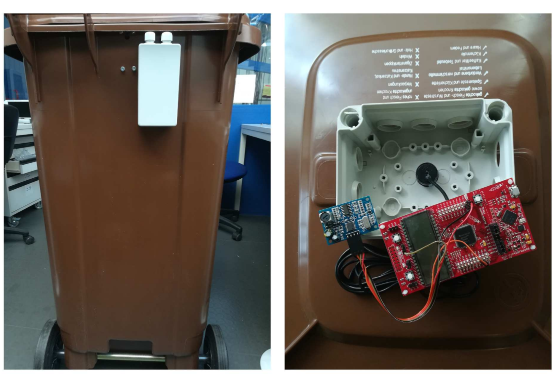

Figure 1. Container (left) and electronics (right)

Simulations have shown, that the combination of these techniques can significantly reduce the time needs for the collection of waste [11]. A comparison of related projects shows that most route optimization projects for the collection of waste lead to the development of a computer simulation [12]. The reasons which prevented the incorporation of this work in practical smart city implementations were, amongst others, present contracts with service providers and logistical obstacles conflicting with changes to the waste collection routes. Possible time savings could only be computed but not really be exploited in many places.

2.1.1. Practical implementation In Regensburg, we recently started applying route optimization and fill level sensing to the collection of the containers for biological waste. Figure 1 (left) shows one such container and the protection casing on the back which houses our electronics. Actually, the right part of the picture with the detail view of the electronics shows an earlier prototype where the electrical parts were placed in the lid. The new box is smaller (making it more difficult to recognize single parts like our LoRa transmitter) and delivers correct data, even if the lid of the container is not properly closed.

A1 | Source

	A	B	C	D
1	Source	Destination	Distancein Km	
2	Cranachweg 18, 93051 Regensburg, Germany	Dr.-Gessler-Straße 47, 93051 Regensburg, Germany	2.4 km	
3	Cranachweg 18, 93051 Regensburg, Germany	Dr.-Gessler-Straße 22, 93051 Regensburg, Germany	0.4 km	
4	Cranachweg 18, 93051 Regensburg, Germany	Dr.-Gessler-Straße 22, 93051 Regensburg, Germany	0.4 km	
5	Cranachweg 18, 93051 Regensburg, Germany	Dr.-Gessler-Straße 47, 93051 Regensburg, Germany	2.4 km	
6	Cranachweg 18, 93051 Regensburg, Germany	Defreggerweg 4, 93051 Regensburg, Germany	0.7 km	
7	Cranachweg 18, 93051 Regensburg, Germany	Kirchmeierstraße 20, 93051 Regensburg, Germany	1.0 km	
8	Cranachweg 18, 93051 Regensburg, Germany	Kirchmeierstraße 22, 93051 Regensburg, Germany	0.7 km	
9	Cranachweg 18, 93051 Regensburg, Germany	Simon-Sorg-Straße, 93051 Regensburg, Germany	0.8 km	
10	Cranachweg 18, 93051 Regensburg, Germany	Simon-Sorg-Straße, 93051 Regensburg, Germany	0.8 km	
11	Cranachweg 18, 93051 Regensburg, Germany	A93, 93049 Regensburg, Germany	3.2 km	
12	Cranachweg 18, 93051 Regensburg, Germany	Georg-Herbst-Straße 35, 93049 Regensburg, Germany	1.6 km	
13	Cranachweg 18, 93051 Regensburg, Germany	Georg-Herbst-Straße 18, 93049 Regensburg, Germany	1.7 km	
14	Cranachweg 18, 93051 Regensburg, Germany	Georg-Herbst-Straße 18, 93049 Regensburg, Germany	1.7 km	
15	Cranachweg 18, 93051 Regensburg, Germany	Clausewitzstraße 14, 93049 Regensburg, Germany	1.9 km	
16	Cranachweg 18, 93051 Regensburg, Germany	Scharnhorststraße 24, 93049 Regensburg, Germany	1.8 km	
17	Cranachweg 18, 93051 Regensburg, Germany	A93, Regensburg, Germany	3.6 km	
18	Cranachweg 18, 93051 Regensburg, Germany	Lilienthalstraße 1, 93049 Regensburg, Germany	2.6 km	
19	Cranachweg 18, 93051 Regensburg, Germany	Prüfeninger Str. 141, 93049 Regensburg, Germany	2.6 km	

Figure 2. Distances in an Excel sheet

Regensburg established its novel program for the collection of biological waste in 2018. The city and its surrounds were equipped with 700 new containers. The novelty of the program allows us, computer scientists (headed by author Jan Dünnweber) and electrical engineers (headed by his colleague Martin Schubert) at the *Technical University of Applied Sciences* in Regensburg (OTH), to contribute to the implementation. Noteworthy as well, is that 700 containers are not too many. While the number of possible thorough paths, connecting all containers is tremendous ($\frac{n!}{2}$ is a 1719-digit number), there are only $\binom{700}{2}$ ways to choose an interconnecting route between an unordered pair of containers. Numerically, there are $\frac{700!}{2!(700-2)!} = \frac{700!}{2\times(698)!} = \frac{699\times700}{2} = 244650$ possible interconnections. The Combinations may vary, but the single interconnections are static and can be stored in a file (like shown in Figure 2) or database.

2.1.2. Preliminary work We set up a database, which holds for half of the 700 containers (*one-way*) a file with the distances connecting it to the remaining 699 candidates.

To ascertain the profitability of our undertakings, we also started with a simulation. However, our simulation did not forecast the profit from using ACO on arbitrary routes or the benefits from observing arbitrary containers. We computed a viable forecast for Regensburg (see section 2.1.4). With the help of a group of students (David Burger, Vadim Dechand, Haris Shehzad and Markus Wildgruber), we could fill the mentioned 244650-entry database with concrete distances and average driving times. Figure 2 shows the first 19 entries of this database (in an Excel-sheet). Actually, we maintain this data in Redis to benefit from caching when the same interconnection distances are requested repeatedly [13].

After accompanying the waste collectors and recording the GPS positions of the containers and the time needs for emptying them with a fitness tracking app on the smart phone, we requested the distances for all possible pairs from the *Google Maps* Web service by means of a Java program, which the students have developed to export the distance data into the popular TSPLIB-format [14]. With this representation, our data can be processed using Open-Source ACO-code and other TSP-solvers. Our routing software makes use of the Thomas Stützle Implementation [6] and we used the exact TSP-solver *Concorde* [15] as a reference. The recorded data and container-specific properties, such as the distance of a container to the city center and the number of neighboring containers, helped us to determine realistic probabilities for the containers to be empty at collection time in our simulation. The resulting forecasts were used to select the first 10 containers which we equipped with real electronics. Figure 3 shows the route we computed for connecting 10 containers in red. This route is 11 kilometers long and could be shortened to 7 kilometers (green route) after leaving out two containers, which we identified as empty containers using our fill level sensors.

2.1.3. Overfull and Underfull Containers To integrate fill-level sensing with the ACO-based route computations, we started to cope with underfilled containers. While overfilled containers seem to be a more urgent problem, it is difficult (or almost impossible) to send a vehicle instantaneously, when an overfilled container is detected. Planning collection routes such that underfilled containers are skipped is

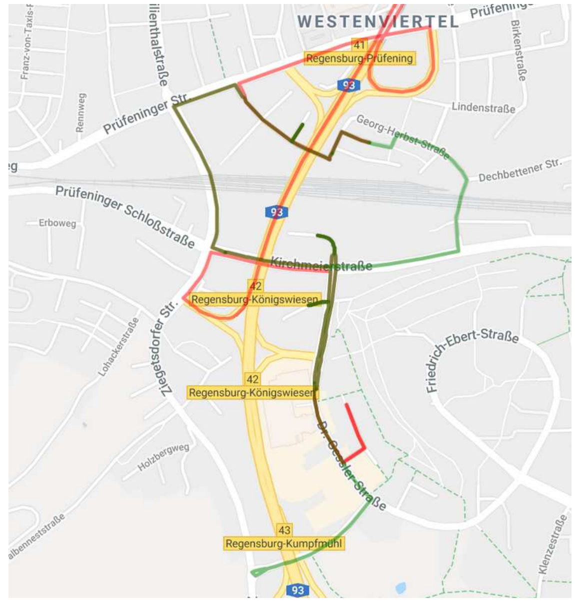

Figure 3. Original (red) and optimized (green) container collection route

much easier and saves time and the saved time is used to collect new containers. The recorded data about overfilled containers helps the Regensburg city council with positioning the new containers where they are needed. Thus, our waste management software, tackles both problems, underfilled containers directly (by leaving them our during the collection) and overfilled containers as well, by finding the best positions for new containers in the long run. Instead of resetting the route computations when a container is added, removed or re-positioned, we keep the ACO software continuously running, since it is known that the swarm algorithm adapts to its input [16].

Master-student Josef Weiß (from Martin Schubert's group) set up the electronics shown in Figure 1: The larger board (on the right) holds an ultrasonic transceiver module and a low power C. The smaller board holds and a LoRa (Low Range) transceiver. We communicate the fill grade measured by the ultrasonic transceiver via a LoRa gateway. The total costs for our first 10 *DIY*-devices were below 1000 € and sponsored by our project partner `kpit.com`.

2.1.4. Fill Level Prediction Using our simulation, we could predict that 10 sensors are enough to start benefiting from our software. To compute a trustworthy prediction, our simulation does not simply roll the dice to decide whether a sensor-equipped container can be skipped. Instead the probability for each container i to be empty was computed individually using the formula: $P(x_i) = q_i * \frac{c_i}{d_i}$. The value for q_i ranges between 0.1 (overfull), 0.2 (full), 0.3 (half-full) and 0.5 (empty) and was set accordingly to the average fill level of that container which we observed on the collection tours. Parameter d_i takes account for the fact that waste containers located near the city center are more likely found full and is 1.2 for a container withing a 2 kilometer circle around the center; 1.1 withing 4 kilometers and 0.9 for containers located more than 6 kilometers away from the center. Parameter c_i is weighted accordingly to the number of waste containers next to it and 0.9 when there is none; 1.1 for up to 5 neighboring containers and 1.3 for 6 or more. With these estimations, we predicted time savings of approximately one hour per day. Regensburg has just started to adapt the routes accordingly. Thus, we will soon know how many extra containers can be emptied within the saved time.

2.2. Application 2: LevelSense™ and PiP-IoT

Smart garbage collection is only a part of a larger smart city initiative in New Zealand which comprises efficient street and traffic lighting, water and wastewater management and energy-saving as well. New Zealand may be considered a small country, but its smart city ambitions are anything but small. For the last two consecutive years New Zealand earned a remarkable amount of honors for its smart cities innovations in the annual Smart City Asia Pacific Awards (SCAPA), a competitive benchmarking program conducted by Council Associate Partner IDC. In reply to the disastrous earthquakes of 2010/11, Christchurch City Council (CCC) put aside funds for Sensing Cities initiatives. In November 2015, the Smart Cities programme was initiated to carry out rapid proof of concept projects.

With 3 more years background, the Christchurch projects are a bit more advanced than the work in Regensburg and the results matter beyond the borders of New Zealand: More than 50% of the worlds population lives in cities by 2050, it will be more the 75%. While the worlds cities only cover two per cent of the global land area, they are responsible for 70% of greenhouse gas emissions and the global climate change. As a consequence, we urgently need sustainable solutions to combat these city-focused issues. The reduction of CO_2 emissions from motor vehicles are all essential elements of these solutions.

2.2.1. A Versatile Level Sensor Christchurch is using smart sensors to ensure that rubbish bins are emptied at the right time.

Figure 4. The CCC waste bin sensor

Figure 4 shows one of the LevelSense™ devices. The LevelSense™ product was developed by Christchurch company PiP IoT to check rubbish levels. The sensors allow city councils contractors to recognise optimal waste collection times. Its key features are

- Tilt sensing
- GPS position and movement monitoring
- Tampering and orientation detection
- Shock and vibration sensing
- Temperature measurement
- An expected battery life of 5 years
- Low power network / LPWAN connectivity
- Real time analytics
- Mobile / Web reporting
- SIGFOX class 0 certification

2.2.2. Project Outcomes Christchurch's rubbish collection contractors work constantly to empty the city's bins but when they arrived at a bin, before the introduction of LevelSense™, they did not know whether it was empty, full one or an overflowing mess. PiP IoT developed LevelSense™ primarily to avoid overflowing public rubbish bins, which where a frequent source of complaints from residents. Though, with the LevelSense™ devices, multiple other enhancements are anticipated.

- A reduction of CO2 emissions and pollution: Fewer rubbish collection trucks on the road for less time means lower fuel consumption and lower greenhouse gas emissions. Fewer collection trucks on the road will also mean less noise pollution, air pollution, and less wear and tear on the road network.

- A reduction in operational costs: Managing waste takes a large portion of our rates. Bin level sensors and monitoring solutions have been known to reduce waste collection costs by up to 50 % (fewer collections mean less money spent on driver hours, fuel and truck maintenance).
- Improved hygiene: overflowing rubbish is a breeding ground for bacteria, insects and pests because of accumulated rubbish. Its a public nuisance and unpleasant for residents and tourists.
- Identification of misuse of public rubbish bins: for example, sudden spikes in rubbish levels at night can indicate that household rubbish is being dumped illegally from residents or campers.

2.2.3. The PiP IoT mobile waste monitoring App

Christchurch selected the local company PiP IoT to develop and test sensor and software technology with the objective of creating operational efficiencies and an enhanced streetscape for citizens.

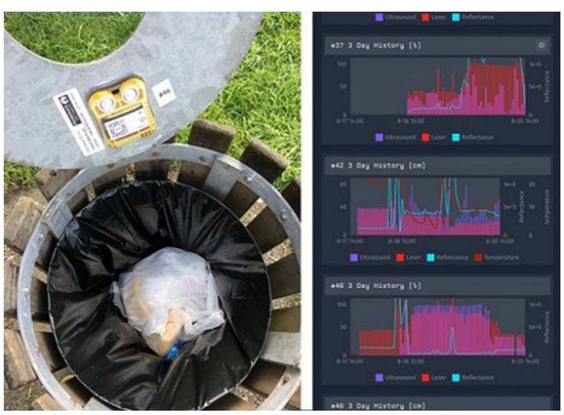

Figure 5. LevelSense™ in container lid (left) and the Christchurch online monitoring app (right)

PiP IoT has rolled out LevelSense™ devices to bins around rubbish trouble spots, including a city park and a retail area. Notifications are sent to contractors phones and bin status across the city can be viewed in an online dashboard telling them how full each bin is (see figure 5, right). Rubbish levels are also being tracked, providing a graphic illustration of when the bins are used most.

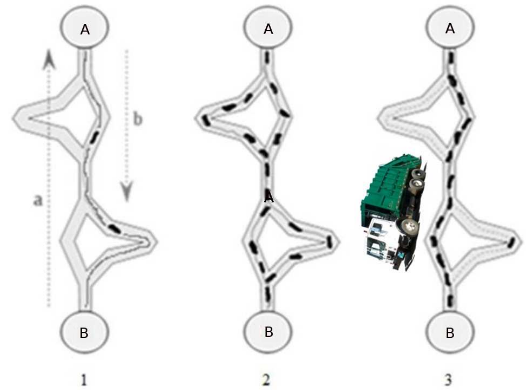

Figure 6. Three iterations of ACO

2.2.4. Updating and Distributing Information

For finding the optimal route connecting the full (or almost full) containers, Christchurch pioneered in the use of ACO (Actually, the Christchurch implementation served as an exemplar for the Regensburg project described in section 2.1).

In Christchurch, the original algorithm was tuned to run on embedded devices using two types of locally accessible information:

1. problem-specific information, i.e., distance among containers, and
2. information added by ants during previous iterations of the algorithm, such as pheromone values.

While building a solution, each simulated ant collects information on the problem characteristics and on its own performance, and uses this information to modify the representation of the problem. The modified representation is the new environment shared by all the ants and seen locally by other ants. This form of indirect communication mediated by the environment is called *stigmergy*, and is typical of social insects [6]). Figure 6 shows how the information updates are used for routing the garbage truck: it will take path b from $A \rightarrow B$ already after the third iteration, since in the second iteration more simulated ants have returned from this path, leading to a stronger pheromone concentration than on the longer return path a from $B \rightarrow A$.

The waste collection problem in Christchurch poses a dynamic instance of the TSP, where cities are represented by containers and containers which are (almost) empty are not collected. Containers which are full or almost full appear in the route dashboard and will be collected, once a vehicle is near it.

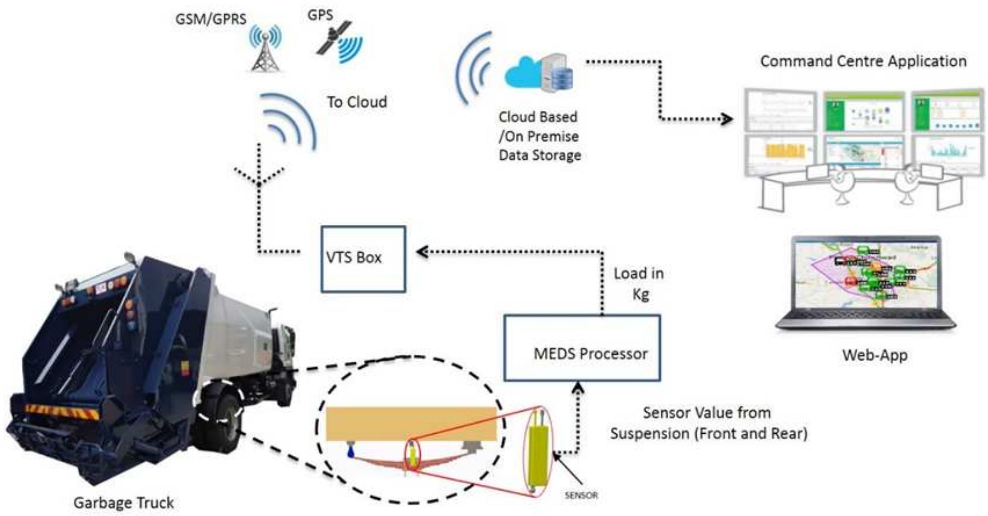

Figure 7. Suspension Monitoring for Routing Malaysia's Trucks Optimally using the Greedy Approach

3. The Greedy Approach to the Travelling Salesman Problem

Regensburg's project partner `kpit.com` has been working on the use of TSP solvers for vehicle routing problems in two former projects: One was focused on collecting waste and another one was focused on scheduling busses.

Both projects started from a brute-force method, i. e., a computer program that compares all possible routes. This method was improved using a greedy approach. Similarly to ACO, the main idea in the greedy algorithm is *local* optimization, i. e., the algorithm picks what seems to be the best thing to do at a particular time, instead of considering the global situation. In comparison with ACO, the greedy algorithm finds approximate solutions using a more coarsely granular procedure leading to an increase of inaccuracy but also to an increase in performance in many cases. In terms of swarm intelligence, the greedy approach is closer to the *Monarch Butterfly Optimization* (MBO) than to ACO [17] (although our implementation has no perturbation step). The greedy algorithm can be sketched as follows

1. Calculate transport times and distances between all stops, e.g. pickup and destination locations, as well as vehicle start locations.

2. Based on the information collected in step 1, generate routes and assign vehicles. The primary constraint is that the total travel time for the first person to board the vehicle should not exceed 1.5 times the time taken for him to reach the destination had he chose his direct transport option.

3. For each node (stop), the eligible (satisfying time & capacity constraints) neighbors are evaluated to form paths and the ones which meet the criteria (serve maximum demand with shortest path) are selected. The acceptance criteria considers aspects like remaining capacity, direction deviation from the destination and total travel time.

4. Since this is a greedy approach, this algorithm is susceptible to choice of first route. We run it for a set of feasible origins and best possible solution (i. e. the one which requires the minimum number of resources) is chosen.

The first route is chosen by picking the farthest location from the office. We draw a circle with the office as the center and the farthest location as radius, thus, capturing all the stops.

The above description is about routing busses collecting employees. However, it can be fine tuned for routing waste collection vehicles as well. Interestingly, `kpit.com` implemented this use case in Malaysia without container sensors. Instead of monitoring the containers, the collection vehicle load was monitored, allowing for a predictive estimation of the container fill grades.

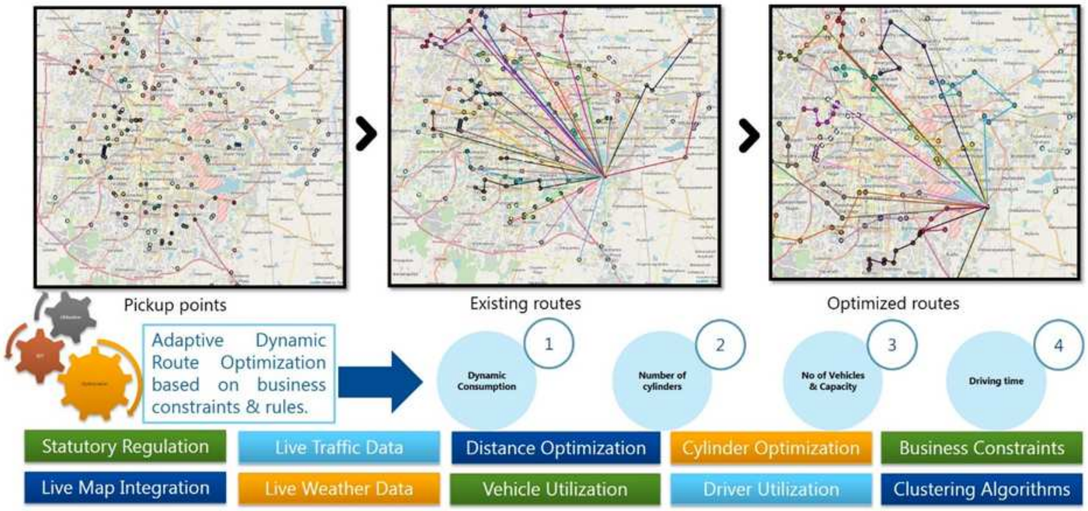

Figure 8. Employee collection

3.1. Application 1: The Malaysia Garbage Initiative

Like New Zealand, Malaysia is another trend-setter concerning smart city activities and their garbage collection initiative is unique in multiple aspects. Figure 7 shows how the components of this IoT infrastructure work together: the MEDS (*Mechanical Engineering Design Service*) is was developed by `kpit.com`'s engineering team to connect a microcontroller to the VTS (*Vehicle Tracking System*). This microcontroller converts the measured deflection on the leaf suspension to payload weight and uses the GSM network to transmit the data back to the backend server.

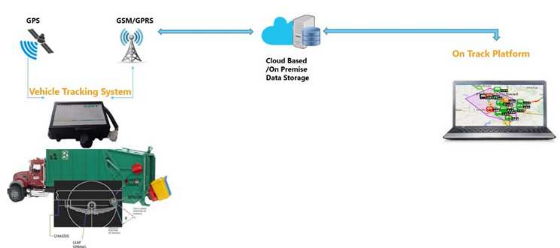

Figure 9. Real-time routing

The city does not only encompass the collection of garbage in a smart manner but it also takes into account the fleet of collection trucks and their maintenance to ensure availability and the necessary framework to manage the contingencies that may occur.

For monitoring the garbage collection, Malaysia's garbage trucks are equipped with leaf spring sensors measuring the vehicles' suspension levels. The measured data and the GPS positions of the respective trucks can be visualized in a Web App and they are also used as input of the command center application for coordinating the trucks dynamically. The other parameters processed by this application comprise static information, such as the truck parking locations, the bin locations and time estimates before a bin is full. Using these data, the Malaysian authorities can observe the amount of the loaded garbage at each location, which helps them to avoid vehicle overloading and truck breakdowns.

The vehicle tracking system makes use of geo fencing to track discrepancies between the planned and the actually driven routes (see figure 9). Comprehensive reports of every vehicle's trip history, map views and the cargo weight details are available in the Web App at the click of a button.

3.2. Application 2: Optimized Bus Routing for Collecting Employees in Bangalore

The costs of transporting employees is a concern for many companies. Especially in India, where the number of employees is high in many companies these costs are huge: employee transport costs are the third highest cost after employee salaries and rentals. Therefore, minimizing these costs is of vital importance.

The `kpit.com` initiative was undertaken to find minimal routes and also minimize the number of vehicles required to transport employees located at various location in the metro city of Bangalore. The practical implementation (sketched in Figure 8) makes use of the greedy approach with the follwing setup:

Input

- The demand at every stop (employees can order a bus using a smart phone App)
- The stop locations and their positions
- The available fleet and their corresponding capacities

Constraints

- The total demand may not exceed the maximum capacity of the 39 vehicles ranging between 6 and 30 passengers. There are 106 stops and 191 employees are using the bus transport.

4. Conclusion and Future Perspectives

In this work, we have analyzed and compared the vehicle routing procedures used in four practical smart city implementations. Our analysis has shown that the fundamental IoT approach, allowing people to order on-demand (using their phone or a signal button at the bus stop), can be improved significantly by route optimization algorithms. The analysis of the waste collection projects has also shown that the combination of processing fill level signals with route optimization leads to the best collection times.

However, there is also a trade-off between finding the best schedule or route and taking into account the specific requirements of a particular application. While ACO can lead to a shorter route than the greedy algorithm, applications like the employee collection must cope with time windows [18]. a better utilization of busses is not no more beneficial, when it comes for the price of employees arriving late at work. A specific characteristic of both, the Bangalore and the Malaysia use case was that arbitrary deviations from the collections plans cannot be tolerated. This additional constraint is always given, when a route planning algorithm is applied to a vehicle with a limited load capacity or operating distance [19]. This observation from project that were all (at least partially) implemented in practice distinguishes our work from simulation experiments that are often aimed at improving route plans at all costs. The discussed experiences with the discrepancies between theoretical and practical time savings can help estimating the limitations of futuristic projects targeted at fully automatized public transport or waste collection.

Our next plans contain an improvement of the reliability of the employed forecasts by the use of more sensors and more detailed data (including e. g. loop ways and one-way streets). Especially in the narrow lanes of the Regensburg city center, the shortest way is not always the fastest and the traveling times predicted by the Google Web service can be infeasible for a garbage truck. However, the time saving which we could prove for all projects let already rate all of them as a success.

References

[1] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Management Sciences*, vol. 6, no. 1, 1959.

[2] J. Chen, R. Bai, H. Dong, R. Qu, and G. Kendall, "A dynamic truck dispatching problem in marine container terminal," *Symposium Series on Computational Intelligence (SSCI)*, pp. 1–8, Dec 2016.

[3] Hyundai Motor Company, "Garabge Truck Press Pack." Online e-Brochure `http://www.hyundai.com/gd/en/find-a-car/garbage-truck/press-pack.html`, 2018.

[4] V. V. Burkhovetskiy and B. Y. Steinberg, "Parallelizing an exact algorithm for the traveling salesman problem," *Procedia Computational Science*, vol. 119, no. C, 2017.

[5] Volvo Group, "Volvo pioneers autonomous, self-driving refuse truck in the urban environment." *Press release* `http://www.volvogroup.com/en-en/news/2017/may/news-2561936.html`, 2017.

[6] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," 1997.

[7] N. V. Karadimas, N. Doukas, M. Kolokathi, and G. Defteraiou, "Routing optimization heuristics algorithms for urban solid waste transportation management," *Transactions on Computing*, vol. 7, no. 12, pp. 2022–2031, 2008.

[8] D. Burger, J. Weiß, A. Sarkar, K. Kirsch, and J. Dünnweber, "Combining fill-level sensing with route optimization for a more efficient waste collection," *18th European Conference on Digital Government (submitted)*, 2018.

[9] A. C. Lundin, A. G. Özkil, and J. Schuldt-Jensen, "Smart cities: A case study in waste monitoring and management," in *50th Hawaii International Conference on System Sciences*, 2017.

[10] Z. Ismail and S. L. Loh, "Ant colony optimization for solving solid waste collection scheduling problems," *Journal of Mathematics and Statistics*, vol. 5, pp. 199–205, 03 2009.

[11] S. Sharmin and S. T. Al-Amin, "A cloud-based dynamic waste management system for smart cities," in *Proceedings of the 7th Annual Symposium on Computing for Development*, pp. 20:1–20:4, ACM, 2016.

[12] E. D. Likotiko, D. Nyambo, and J. Mwangoka, "Multi-agent based iot smart waste monitoring and collection architecture," *Computing Research Repository*, 2017.

[13] J. L. Carlson, *Redis in Action*. Greenwich, CT, USA: Manning Publications Co., 2013.

[14] G. Reinelt, "TSPLIB - A t.s.p. library," Tech. Rep. 250, Universität Augsburg, Institut für Mathematik, Augsburg, 1990.

[15] D. Applegate, "Concorde - a code for solving traveling salesman problems." http://www.math.princeton.edu/tsp/concorde.html.

[16] D. Angus and T. Hendtlass, "Dynamic ant colony optimisation," *Applied Intelligence*, vol. 23, pp. 33–38, Jul 2005.

[17] S. Chen, R. Chen, and J. Gao, "A monarch butterfly optimization for the dynamic vehicle routing problem," *Algorithms*, vol. 10, no. 3, 2017.

[18] P. Kirci, "An optimization algorithm for a capacitated vehicle routing problem with time windows," *Sdhan*, vol. 41, pp. 519–529, 05 2016.

[19] L. Yu, Z. Wei, Z. Wang, Y. Hu, and H. Wang, "Path optimization of auv based on smooth-rrt algorithm," in *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 1498–1502, Aug 2017.