

Crowd Associated Network: Exploiting over Smart Garbage Management System

Saiful Azad, Arafatur Rahman, A. Taufiq Asyhari, Al-Sakib Khan Pathan

Abstract

Most of the existing *non-real-time* applications utilize *infrastructure* based or *semi-infrastructure* based network architectures. Such a network architecture demands a considerably high installment and maintenance cost. To alleviate the cost, in this paper, we propose an efficient *infrastructure-less* network architecture, named *Crowd Associated Network (CrAN)*. In *CrAN*, a set of crowds plays significant roles by completing the communication gaps among various associates in the network, and hence the name. We show the usability of this proposed architecture to support *non-real-time* data transmission over a *Smart Garbage Management System (SGMS)*, where optimum solutions need to be discovered to minimize the management cost. Due to the complexity of the optimization problem, we approximate these optimum solutions using a *Genetic Algorithm (GA)*. In the implementation of the GA, we apply new fitness functions to discover a feasible tradeoff between distance and waste volume. We then compare the performance of the proposed fitness functions with that of an existing fitness function. The results favorably suggest the necessity of employing the proposed fitness functions to obtain near-optimum solutions.

S. Azad, and A. Rahman are with the Faculty of Computer Systems and Software Engineering, University Malaysia Pahang, Malaysia (email: {saifulazad, arafatur}@ump.edu.my), and IBM Center of Excellence, UMP, Gamabang, Malaysia.

A. T. Asyhari is with Cranfield University, Shrivenham Campus, Shrivenham SN6 8LA, UK (from February 2017). He was previously with the School of EECS, University of Bradford, Bradford BD7 1DP, UK (email: taufig-a@ieee.org).

A. S. K. Pathan is with the Faculty of Computer and Information Systems, Islamic University in Madinah, Madinah al-Munawwarah, Kingdom of Saudi Arabia; and Department of Computer Science and Engineering, Southeast University, Bangladesh (email: sakib.pathan@gmail.com)

Published in IEEE Communications magazine (2017), vol 55/7 pp 186-192
(DOI: 10.1109/MCOM.2017.1600379)

I. INTRODUCTION

In general, all computer applications can be broadly classified into *real-time* and *non-real-time* applications. In *real-time* applications, responses to certain events are constrained within a fixed time interval, i.e., timeliness is a primary measure of performance. On the other hand, although every *non-real-time* application has its own performance indicator, its required response time is subjective. Consequently, unlike their counterparts, the demand of fixed network architectures by *non-real-time* applications is not imperative. However, à la *real-time* applications, most of the existing *non-real-time* applications utilize *infrastructure* based or *semi-infrastructure* based network architectures. Hence, they charge for high expenditures for installation and maintenance. To alleviate the cost of such applications, an inexpensive but efficient *infrastructure-less* network architecture is considered in this paper. The usability of the proposed network architecture is specifically demonstrated for an important *non-real-time* application, namely *Smart Garbage Management System (SGMS)*.

A number of frameworks have been proposed to satisfactorily manage the garbage problems within the vision of smart cities. For instance, in [1], the authors designed an intelligent solid waste bin to aid the existing waste management system. Their work focused only on the bin design, but designing an appropriate network architecture and optimizing the cost for garbage collection remain out of their scope. In [2], an Internet-of-Things (*IoT*)-based *SGMS* is proposed to reduce the amount of food waste by imposing certain constraints. In that proposed *SGMS*, the smart garbage bins (*SGBs*) communicate among themselves using a wireless mesh network, and transmit data to a router, which then forwards them to a server. All the acquired data are analyzed by the router, which decides service provisioning. Except *SGBs*, all other devices are connected via the Internet. Moreover, a direct path between an *SGB* and the router is assumed. Due to that assumption, the distances between the *SGBs* are kept notably short, which is again impractical. Other solutions for *SGMS* have also been addressed in [3]– [5]. However, most of

these solutions utilize *infrastructure* based or *semi-infrastructure* based network architectures. Consequently, the installment and maintenance cost of these networks are considerably high.

On the other hand, our proposed *infrastructure-less* network architecture aims to support the entire operation of the *SGMS*, and charges a minimum cost to deploy and maintain. The proposed architecture is compatible for any *non-real-time* application, where frequent data acquisition is not necessary for proper functioning of that application. In the proposed network, a set of crowds is utilized with other network components to acquire data from a considerably large area, e.g., a town or a city. The details of the proposed network architecture are discussed in Section II with a compatible application in Section III. Data acquired from the network nodes (*SGBs*) are further processed to assist in discovering optimum solutions (in terms of reducing management cost) for the *SGMS* [6], [7]. In this paper, we employ a *Genetic Algorithm (GA)* to discover such feasible solutions from the acquired data, alike [8], [9]. Within the GA, two new fitness functions are applied and compared with a trivial fitness function. The results hint the necessity of employing new fitness functions to find feasible solutions.

II. CROWD ASSOCIATED NETWORK

The key concept of the proposed *infrastructure-less* network architecture is the utilization of the crowd to complete the communication gaps among the associates. Hence, it becomes an inseparable part of the network; and thereby, it is named as *Crowd Associated Network (CrAN)*. In *CrAN*, two types of components are involved: *i)* dedicated agents, and *ii)* non-dedicated agents.

The dedicated agents are those agents that are solely installed in the network to perform some specific tasks. In general, these agents are static and exchange information with non-dedicated agents to achieve the networking goal. On the other hand, the crowd is the latter type of agent who is equipped with necessary devices and acts like an intermediate relay in the proposed network architecture. The crowd completes the communication gaps among the dedicated agents

and thereby enables them to function properly. It acquires data from one or multiple dedicated agent(s), and delivers multiple copies to other dedicated agent(s). The members of the crowd may also exchange data among themselves with a hope that the cooperating members will deliver the data to one or multiple dedicated agent(s). This technique improves the performance of the network in terms of data delivery and end-to-end delay. A notable point is that everyone in the crowd is qualified to be a part of the network if he/she complies with the network requirements. However, in reality, not everyone would be interested to contribute. Therefore, from now onward, for the sake of distinguishing the non-contributors from the contributors, we will refer to the latter as *volunteers*. They provide services without any expectation of compensation and without any coercion. A volunteer will be given a network component, which he/she has to install in his/her own vehicles (e.g., motorcycle, car, bus, etc.). This network component will be called as a *Volunteer Agent (VA)* throughout the rest of the paper.

As mentioned earlier, this sort of network architecture is suitable for applications with *non-real-time* data, such as *SGMS*, where acquisition can be satisfactorily fulfilled by one or few successful transmission activities per day. There are manifold advantages of using this architecture, such as: *a)* no required fixed infrastructure, i.e., infrastructure-less, *b)* no fixed boundary in terms of deployment, *c)* a smaller number of dedicated agents, *d)* a lower expected deployment and maintenance cost than any *infrastructure* based or *semi-infrastructure* based network. Volunteers are the key actor in the proposed *CrAN* and their recruitment can be facilitated, e.g., through the following provisions.

- Local community may provide incentive to the volunteers through revising or reducing tax and/or other service charges.
- A social awareness campaign may also play a significant role in convincing people to become volunteers.
- Other sources, which can be explored, are the employees from municipal corporation, government offices or social organizations who live around the coverage area.

- Public buses that travel around the cities can contribute to this task or even garbage collecting containers can also be equipped with *VAs* to acquire information.

III. CRAN FOR SGMS

In the following, Subsection III-A briefly introduces the components that are utilized to install the *CrAN*. The proposed *infrastructure-less* network architecture is detailed in Subsection III-B. Subsection III-C discusses the communication protocols that are suitable for the proposed network architecture. Subsection III-D presents the techniques related to data processing and discovering optimum solutions.

A. Network Components

The *CrAN* consists of five distinct components, i.e., *Smart Garbage Bin (SGB)*, *Volunteer Agent (VA)*, *Sink*, *Control Center (CC)* and *Garbage Collecting Agent (GCA)*. All of them have their unique identification numbers. Among them, except *VA*, all others are dedicated agents. A collaborative effort of these components envisions to delivery necessary data and to discover optimum or near-optimum solutions for the *SGMS*, which contribute significantly in reducing waste management cost. The details of the components are briefly discussed below:

Smart Garbage Bin (SGB): Unlike other conventional garbage bins, an *SGB* is embedded with a sensor that can measure the volume of garbage. The *SGB* periodically acquires this information and transfers it along with other necessary information to the encountering associates. The *SGBs* are battery powered, and have low computational abilities and storage capacities. Hence, the following two initiatives are undertaken to enhance the lifetime of an *SGB*: *i*) it can only transfer data whenever necessary without any relaying capability and *ii*) instead of continuously delivering packets to all associates (within the range), a priority based technique is employed to reduce energy dissipation.

Volunteer Agent (VA): The objective of this component is to acquire data from the *SGBs*, and exchange them with compatible associates when encountered. This is the only non-dedicated component in the system and hence, its behavior is unpredictable. Therefore, it is prescribed to assign multiple *VAs* in an area with the idea that at least one of them is able to deliver the data to the appropriate associate(s). In order to obtain a reasonable performance, a *VA* is necessary to be supplied with an affluent energy source. In our case, each *VA* is attached to a vehicle and draws energy from the battery of the vehicle.

Sink: The objective of this component is to exchange information with the *VAs* when both of them are within the communication range. The sinks are dedicated agents, which connect to affluent energy sources through electrical wiring. The *destination sink* is a special type of sink with direct connection to the control center. Unlike other sinks, it only forwards data to the control center and never re-transmits any copies to other associates.

Control Center (CC): The primary objective of the *CC* is to acquire data from the *destination sink* and subsequently, utilize them to obtain optimum solutions for garbage collecting agents with respect to one or multiple parameters, e.g., distance, number of containers, etc. All the computed solutions are stored in a buffer and delivered on a demand basis.

Garbage Collecting Agent (GCA): This component is involved in unloading the *SGBs* by following the optimum solutions provided by the *CC*. The *GCA* can also be utilized to replace batteries when necessary. When a *GCA* completes unloading all assigned *SGBs*, it moves to the dumping zone for releasing the garbage and then returns back to the depot.

B. Network Architecture

The network architecture of the *CrAN* is depicted in Figure 1. It is a two-tier architecture where the first tier is mostly involved in data acquisition, and the second tier is involved in data

processing and discovering optimum solutions for the *GCA*s.

Generally, the *SGB*s are placed beside the roads to ease the garbage collection process. In other words, the *SGB*s are spatially distributed components installed inside the network area. They periodically acquire waste volume status and generate waste DATA packets and other relevant information. The *VAs* are the mobile agents in the network that provide voluntary services and gather necessary DATA packets from the *SGB*s when they encounter the latter. Thereafter, the *VAs* exchange these packets opportunistically when they come into contact with nearby *sinks* or other *VAs* with a hope that these relaying nodes will deliver the packets to the destination (more specifically, to the *destination sink*). The *sinks* also apply a similar opportunistic forwarding technique to route the packets to the destination. Consequently, these components have a routing capability to decide which packets to transmit, how many duplicate copies to spread, etc. Finally, the *destination sink* receives packets from various sources and delivers them to the *CC* for further processing.

As noted before, the second tier in the network architecture is involved in data processing, discovering optimum solutions for the *GCA*s and acquiring feedback from the *GCA*s. After receiving packets from the *destination sink*, the *CC* processes the required data and stores them in a buffer after performing a simple freshness treatment, i.e., old data are overwritten with the fresh data. Then, periodically it computes the optimum solutions with respect to one or multiple parameters as mentioned before. At a later time, these solutions are delivered to the *GCA*s in order to collect the garbage in the most efficient manner. A *GCA* unloads an *SGB*, updates the system if required, and also changes energy source whenever necessary. It may also acquire data from other *SGB*s, which are not unloaded, but encountered during the trip. At the end, when it returns back to the depot, it delivers the feedback and the acquired packets to the *CC*.

C. Data Transmission

In *CrAN*, only the *SGBs* can generate DATA packets, and all other associates act as intermediate relays to deliver them to the *destination sink*. In a DATA packet, an *SGB* encloses the waste volume status as well as the status of the energy source. The prior one indicates when to unload it and the latter indicates when to change its energy source. A single copy of a DATA packet may result in failure to reach at the destination since the nodes experience intermittent connectivity due to a large network area. Therefore, duplication of packets may result in a high probability of reliable delivery to the *destination sink* within a given time frame. Hence, an *SGB* injects L copies of a DATA packet in the network through various associates (mainly *VAs*).

As mentioned earlier in Subsection III-B, the network components in the *CrAN* have heterogeneous capabilities in terms of data transmission. An *SGB* and the *destination sink* do not require routing capabilities. A simple *MAC* protocol can enable these components to transmit or receive DATA packets from other associates. By contrast, the rest have to relay packets as they are intermediate nodes, and hence, they need routing capabilities. In the following two subsubsections, we discuss some direction in selecting the most relevant routing and MAC protocols for the *CrAN* architecture.

1) *Routing*: Since the *CrAN* is an *infrastructure-less* network, and the operation area can be considerably large, there is a small possibility that a complete end-to-end route can be discovered for delivering a packet to the *CC*. Thereby, all the components in the network may experience intermittent connectivity or lack of connectivity and time-varying hop-to-hop propagation delays. Hence, the routing protocols that assume direct end-to-end routes before data transmission are not applicable in the *CrAN*. Conversely, there exist opportunistic routing protocols [10]–[12], which store the packets until an opportunity arises to forward them to another node(s) with a hope that the receiving node is the destination or will at least forward the packets to the destination directly or via other intermediate nodes. These protocols are known as *store-and-forward* based

routing protocols.

Most *store-and-forward* based routing protocols can be broadly classified into *replication-based* and *forwarding-based* routing protocols. As the name suggests for the preceding class of protocols, they replicate the packets whenever necessary. A generalized practice, which is observed among these protocols, is that they allow a considerable amount of replication to increase the delivery probability of a DATA packet. By contrast, *forwarding-based* routing protocols forward a packet until it reaches the destination without any duplication. Although, this approach achieves a higher efficiency in terms of resource preservation and overhead reduction, it experiences a lower packet delivery ratio and higher end-to-end delay. Hence, they are not preferable for adoption in the *CrAN*.

There are various *replication-based* routing protocols proposed in the literature. In [10], an epidemic routing protocol is proposed, which replicates every packet when it encounters a new contact. Hence, its packet delivery ratio is considerably higher than other similar protocols. However, since it is very similar to *flooding* technique, it experiences a considerably high overhead in the network. Consequently, the epidemic routing protocol is neither preferable in this network. On the other hand, there exist protocols, which limit the replication overhead through specific techniques, e.g., in [11], [12]. These protocols are considerably easy to implement and demand a relatively lower computing power. Therefore, these protocols are good candidates for the proposed network architecture.

2) *Media Access Control (MAC)*: Unlike routing protocols, a *MAC* protocol is obligatory for all the network components in the *CrAN*. Among the existing *MAC* protocols, *handshake-based* MAC protocols, such as IEEE802.11 [13] and IEEE802.15.4 [14], are suitable for those networks where channel contentions are frequent phenomena and packet drop probability is high due to collisions. In *handshake-based* MAC protocols, a node has to reserve a channel before initiating any transmission attempt through the handshaking procedure. Conversely, a network architecture, like *CrAN*, where contention and collision are seldom phenomena, these

protocols are not applicable due to a considerable amount of overhead they impose before any data transmission. On the other hand, most of the *contention-based* protocols, such as *ALOHA* and *CSMA*, transmit a packet with an assumption that the next node exists within its vicinity. Hence, this type of protocols is neither suitable for the *CrAN*. For this network architecture, only those *MAC* protocols that trigger packets when the nodes come within a communication range are preferred. A node must store the packets and transmit them opportunistically. Such a mechanism is embedded within *Store-and-Delivery based MAC (SD-MAC)* protocol as proposed in [15]. It is a light-weight MAC protocol, which is suitable for most sensor nodes.

D. Data processing and discovering optimum solutions

After acquiring necessary DATA packets through the *CrAN* architecture, the *CC* extracts all the required data and then at a later time, processes these data to find optimum solutions. An *SGMS* is incomplete if the acquired data are not processed, and optimum results (in terms of minimizing the management cost of the system) are not calculated.

For simplicity of our forthcoming discussion, let us assume that the *CC* has adequate recent data of the network. It then has to compute feasible solutions and deliver them to the *GCAs* on a demand basis. In terms of cost optimization, let us assume that we aim to minimize the requirements of the *GCAs*. Note that if a single *GCA* can unload all the bins, then this problem can be casted into a simple and well-studied *Traveling Salesman Problem (TSP)*. However, in reality, this latter assumption is less realistic since all the *GCAs* have limits in terms of capacity. It is therefore necessary to consider this constraint, and we shall refer to the more realistic context as a *Garbage Collection Problem (GCP)*.

The *GCP* resembles to the known *Capacitated Vehicle Routing Problem (CVRP)*. In *CVRP*, a fixed fleet of delivery vehicles with identical capacity must be utilized to provide service to known customer demands for a single commodity and from a single depot at minimum cost. The objectives of the *CVRP* include: *i)* minimizing the vehicle fleet, and *ii)* minimizing the

travel time while keeping the total demand of commodities for each route within the capacity of the serving vehicle. However, in *GCP*, instead of minimizing the travel time, maximizing the garbage collection is envisioned with an assumption that it will reduce the requirement of the *GCA*s.

All the trivial and new ideas discussed before can be hypothesized as follows:

hypothesis I: minimizing the travel time will minimize the requirements of the number of *GCA*s.

hypothesis II: maximizing the waste volume collection by a *GCA* will minimize the requirements of the number of *GCA*s.

hypothesis III: minimizing the coverage distance for collection per waste volume will minimize the requirements of the number of *GCA*s.

Among the aforementioned hypotheses, we consider *hypothesis I* as a trivial (benchmark) objective since it has been widely used in the evaluation of existing algorithms with a similar objective (e.g., *CVRP* [9]), whereas *hypotheses II* and *III* represent the proposed new objectives. Similar to its predecessor, the *GCP* is an *NP*-hard problem for a large number of *SGB*s, i.e., $N \geq 100$. It is infeasible to solve this type of problems in polynomial times. Several *metaheuristic* methods that can produce near-optimum solutions have therefore been proposed since the last decade. Among them, *Genetic Algorithms* [4] are widely applied due to their reduced solving time and quality of solutions (if relevant parameters are selected properly). In this paper, this technique is employed to obtain viable solutions.

A *GA* utilizes a set of *populations* and creates several *generations* to solve a particular optimization problem. A *population* consists of a set of solutions, *a.k.a.*, chromosomes with each containing the solution in the form of genes. A *crossover* operation is performed for the reproduction of new chromosomes whereas a *mutation* operation makes random changes in the solutions or chromosomes. A selection procedure is invoked to select only the fittest solutions as parents, which are then utilized by the *crossover* operation to create the other fit

solutions, namely *offsprings*. At the end of each iteration, a new generation is produced from the combination of the old generation and the new *offsprings*. Generally, the size of the new generation is larger than the previous one. To keep the size fixed, the fitness values of all the solutions are calculated. At the end, a filtering procedure is applied so that only the fittest nodes survive and get themselves placed in the population. In our case, we need three corresponding fitness functions for three hypotheses, such as:

$$\mathcal{F}(i) = \frac{1}{1 + (\delta_{0,j} + \sum_{j=1}^m \delta_{j,j+1} + \delta_{m,0})} \quad (1)$$

$$\mathcal{F}(i) = \sum_{j=1}^m \vartheta(j) \quad (2)$$

$$\mathcal{F}(i) = \frac{1}{1 + \frac{\delta_{0,j} + \sum_{j=1}^m \delta_{j,j+1} + \delta_{m,0}}{\sum_{j=1}^m \vartheta(j)}} \quad (3)$$

where $\mathcal{F}(i)$ measures the fitness of a particular solution/chromosome i of a certain population, which has m number of genes (i.e., m *SGBs*), $\delta_{\ell,k}$ denotes the Euclidean distance between *SGBs* ℓ and k , $\vartheta(j)$ denotes the waste volume of a particular *SGB* j , $j \in \{1, \dots, m\}$. Equations (1), (2) and (3) are used to select solutions according to *hypotheses I*, *II* and *III*, respectively. Since the volume of each *SGB* is considered random against a fixed capacity container, the size of the chromosomes/solutions may vary, which makes the implementation of the GA more challenging.

IV. EVALUATION

The proposed hypotheses are evaluated by conducting a comprehensive simulation campaign. The details of this simulation campaign along with parameter optimization and results analysis are discussed in the following.

A. Simulation Scenario

To evaluate our hypotheses, we consider three Euclidean 2D areas of $500\text{ m} \times 500\text{ m}$, $2000\text{ m} \times 2000\text{ m}$, and $5000\text{ m} \times 5000\text{ m}$, where the *SGBs* are installed in a random fashion. We consider a variable number of nodes N (ranging from 10 to 100) that are deployed within the area following a uniform probability distribution. Every *SGB* has a unique identification number and in this process, 0 is considered as the identification number of the depot. We assign a random waste volume to every *SGB*, which is assumed to be less than the bin capacity β_c , and the capacity, ζ_c , of each container is assumed to be symmetric. To stress the simulation, all the nodes are considered to have a waste volume, which is larger than the minimum considerable volume μ , i.e., $\vartheta_i > \mu$. The distance of the two nodes is found using a *Euclidean distance*, δ . We assume that the node that travels within the shortest distance would require minimal time to travel the area. For simplicity, we also assume that the *CC* has adequate recent data to discover appropriate solutions.

In order to discover feasible solutions using the *GA*, 1-opt *crossover* and 1-opt *mutation* are utilized. The following parameters are considered throughout the simulation campaign: $\zeta_c = 1000\text{ kg}$, $\beta_c = 200$, $\mu = 0.5 \times \beta_c$, *generation* = 50, *sizeof(population)* = $2 \times N$. The length of the chromosomes/solutions varies from the minimum $\lfloor \zeta_c / \beta_c \rfloor$ to the maximum $\lfloor \zeta_c / \mu \rfloor$. Every scenario runs with 100 different seeds which are then averaged before plotting on a graph. Finally, the simulation program has been implemented in C++ and all the results are tabulated in a plain text file.

B. Parameter Optimization

For finding appropriate solutions from a *GA*, it is obligatory to utilize optimum parameter values, which are volatile and can change from one scenario to another. Generally, *mutation rate* and *crossover rate* play important roles in discovering appropriate solutions in any evolutionary algorithm like *GA*. Hence, a simulation campaign is carried out to discover optimum *mutation*

rates and *crossover rates* for the three preferred scenarios. These values are later utilized in subsequent simulations. In Figure 2, the impacts of various *mutation rates* and *crossover rates* on utilization of containers—where Equation (3) is specifically selected for the fitness function—are shown using contour graph for $N = 30$. In this figure, mild colors represent lower utilization whereas intense colors represent higher utilization. It can be observed from the figure that multiple *mutation-crossover-rate* pairs may offer similar types of solutions. Hence, for subsequent simulations, the optimum parameter values in Table I are adopted.

C. Results and Discussion

For evaluating the performance of the three hypotheses and to discover their effectiveness in finding optimal or near optimal solutions, we consider three metrics, namely the required number of containers, utilization of the *GCA*s and travel distance per waste volume or in short, distance per volume. The results are depicted in Figures 3a, 3b, and 3c. All the results are normalized before plotting on graphs using a *max-min normalization* technique. Consequently, for each metric, the performance resulting from each given hypothesis does not vary significantly with the size of the area.

From Figure 3a, it can be observed that since *hypothesis II* endeavors to maximize the garbage collection for a fixed capacity container, its utilization is considerably higher than the other two hypotheses for any preferred area. It achieves the highest utilization of vehicle capacity, i.e., 1 at $N = 100$. Since *hypothesis III* attempts to minimize the distance per volume collection, it achieves considerably higher utilization than *hypothesis I*, i.e., 0.74. These results of utilization reflect on the requirement of the *GCA*s and are further illustrated in Figure 3b. Since *hypothesis II* utilizes the *GCA*s in the most efficient manner, it requires a lower number of containers than the other two hypotheses. Between *hypotheses I* and *III*, the latter one outperforms its counterpart. For *hypotheses II* and *III*, the required number of containers increase linearly with N . On the other hand, for *hypothesis I*, the required number of containers have a linear

trend with N initially, but seem to have an exponential increase when N is sufficiently large. Moreover, *hypotheses II* and *III* appear to have nearly the same performance in terms of the required number of containers.

Although, from the aforementioned discussion, it may seem that *hypothesis II* yields a superior performance, Figure 3c shows other important insights. Since *hypothesis II* attempts to maximize the volume, a *GCA* has to travel a long distance, which is the longest among the three hypotheses, i.e., 0.82 or more. By contrast, although *hypothesis III* requires a slightly higher number of containers, its average travel distance is considerably lower than the preceding one. Again, another interesting observation is that initially *hypothesis I* and *III* yield almost equal travel distances, but as we increase N , *hypothesis III* will have a lower average travel distance per volume than its counterpart. From the investigation, it is found that since *hypothesis I* tries to minimize the distance, the distance for various containers increases chronologically. For instance, the first container has to travel the shortest distance and the final one has to travel the longest distance, which is even longer than the longest distance of *hypothesis III*. Consequently, longer distances dominate when the average is calculated. Therefore, if fuel consumptions are taken into account when calculating optimum solutions, *hypothesis III* might offer a better performance (in terms of cost) than the other two hypotheses.

V. CONCLUSION

In this paper, we have proposed a low cost but efficient *infrastructure-less* network architecture, which is exploited in the *Smart Garbage Management System (SGMS)*. Since the crowd is associated inseparably within the architecture, it is named as *Crowd Associated Network (CrAN)*. A set of crowds works like mobile agents (called *Volunteer Agents (VA)* in this paper) who acquire data from various dedicated agents of the network. At a later time, it delivers the acquired data to the other dedicated agent(s) or similar agent(s) with a hope that the other parties can deliver the data to one or multiple dedicated agent(s). This combined effort is envisioned to deliver data

to the *destination sink*, which is further connected to the control center. Thereby, these agents complete the communication gaps among the dedicated agents of the network. After receiving all the packets from various sources, at a later time, the control center computes optimum solutions with respect to garbage collection in order to minimize the management cost of the *SGMS*. We have employed the *Genetic Algorithm* to discover feasible solutions from the acquired data utilizing three objectives, namely minimizing the travel distance, maximizing garbage collection and minimizing the travel distance per volume. We have performed an extensive simulation campaign with these objectives and discovered that the third objective seems to offer more feasible solutions than its counterparts.

VI. ACKNOWLEDGEMENT

This work is partially supported by the RDU grant, (no. RDU160360), funded by University Malaysia Pahang (UMP), Malaysia.

REFERENCES

- [1] M. Abdulla Al Mamun, M. A. Hannan, A. Hussain and H. Basri, "Integrated sensing systems and algorithms for solid waste bin state management automation," *IEEE Sens. J.*, vol. 15, no. 1, pp. 561–567, Jan. 2015.
- [2] I. Hong, S. Park, B. Lee, J. Lee, D. Jeong, and S. Park, "IoT-based smart garbage system for efficient food waste management," *The Scientific World J.*, vol. 2014, Aug. 2014.
- [3] B. Chowdhury and M. U. Chowdhury, "RFID-based real-time smart waste management system," in *Proc. Australasian Telecommunication Networks and Applications Conference*, Christchurch, NZ, Dec. 2007.
- [4] S. Longhi, D. Marzioni, E. Alidori, G. D. Buo, M. Prist, M. Grisostomi, and M. Pirro, "Solid waste management architecture using wireless sensor network technology," in *Proc. The 5th International Conference on New Technologies, Mobility and Security (NTMS)*, May 2012.
- [5] V. Catania and D. Ventura, "An approach for monitoring and smart planning of urban solid waste management using Smart-M3 platform," in *Proc. The 15th Conference of Fruct Association*, 2014.
- [6] Y. Kryftis, G. Mastorakis, C. X. Mavromoustakis, J. M. Batalla, J. J. P. C. Rodrigues and C. Dobre, "Resource usage prediction models for optimal multimedia content provision," in *IEEE Syst. J.*, vol. PP, no. 99, pp. 1–12, doi: 10.1109/JSYST.2016.2548423, 2016.

- [7] Y. Kryptis, C. X. Mavromoustakis, G. Mastorakis, J. M. Batalla and P. Chatzimisios, "Epidemic models using resource prediction mechanism for optimal provision of multimedia services," in *Proc. The 20th IEEE International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD)*, Guildford, UK, Sep. 2015.
- [8] K. Buhrkala, A. Larsena, and S. Ropkea, "The waste collection vehicle routing problem with time windows in a city logistics context," in *Proc. Social and Behavioral Sciences*, vol. 39, pp. 241–254, 2012.
- [9] N. V. Karadimas, K. Papatzelou, and V. G. Loumos, "Genetic algorithms for municipal solid waste collection and routing optimization," *Artificial Intelligence and Innovations 2007: from Theory to Applications*, vol. 247, pp. 223–231, 2007.
- [10] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Department of Computer Science, Duke University, Tech. Rep. CS-2000-06, Apr. 2000.
- [11] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and Wait: An efficient routing scheme for intermittently connected mobile networks," in *Proc. ACM WDTN*, Philadelphia, USA, pp. 252-259, Aug. 2005.
- [12] A. Balasubramanian, B. N. Levine, and A. Venkataramani, "DTN routing as a resource allocation problem," in *Proc. ACM SIGCOMM*, Kyoto, Japan, pp. 373-384, Aug. 2007.
- [13] IEEE-SA, "IEEE 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," Apr. 2012.
- [14] IEEE-SA, "IEEE 802.15: Wireless Personal Area Networks (PANs)," 2014.
- [15] S. Y. Liew, S. Azad, H. G. Goh, B. Y. Ooi, and A. Rahman, "A store-and-delivery based MAC protocol for air-ground collaborative wireless networks for precision agriculture," in *Proc. The 3rd International Workshop on Internet of Things Technologies (IoTT 2015)*, Melbourne, Australia, Dec. 14-17, 2015.



Saiful Azad [M'16] received his PhD in Information Engineering from the University of Padova, Italy, in 2013. Currently, he is serving as a faculty member at the faculty of Computer Systems & Software Engineering, University Malaysia Pahang, and a fellow of IBM center of excellence, Malaysia. His research interests includes: design and implementation of communication protocols for various network architectures, QoS issues, network security, and simulation software design.



conference publications.

Arafatur Rahman [M'15] received his Ph.D. degree in Electronic and Telecommunications Engineering from the University of Naples Federico II, Naples, Italy, in 2013. He worked as a Postdoc in the same university in 2014. Currently, he is an Assistant Professor with Faculty of Computer Systems & Software Engineering, University Malaysia Pahang. He has become the IEEE member from 2014. His research interests include Cognitive Radio Networks, IoT, and 5G, and he has co-authored around 40 journals and



A. Taufiq Asyhari [M'13] received his Ph.D. degree from the University of Cambridge-UK. Since February 2017, he has been a Lecturer in Networks and Communications at Cranfield University-UK. Prior to this position, he held a position at the University of Bradford-UK. He has won several academic awards, including a Grant from NSC-Taiwan (2013) and the ISWCS Best Paper Award (2014). His research interest is information theory with applications to wireless and nano-molecular networks.



IEEE.

Al-Sakib Khan Pathan [SM'14] received Ph.D. in Computer Engineering in 2009 from Kyung Hee University, South Korea and B.Sc. in Computer Science and Information Technology from Islamic University of Technology, Bangladesh in 2003. He is currently an Associate Professor in the CSE Department at Southeast University, Bangladesh. He has served as a Chair and Committee Member in numerous international conferences and on editorial roles of several renowned journals. He is a Senior Member of

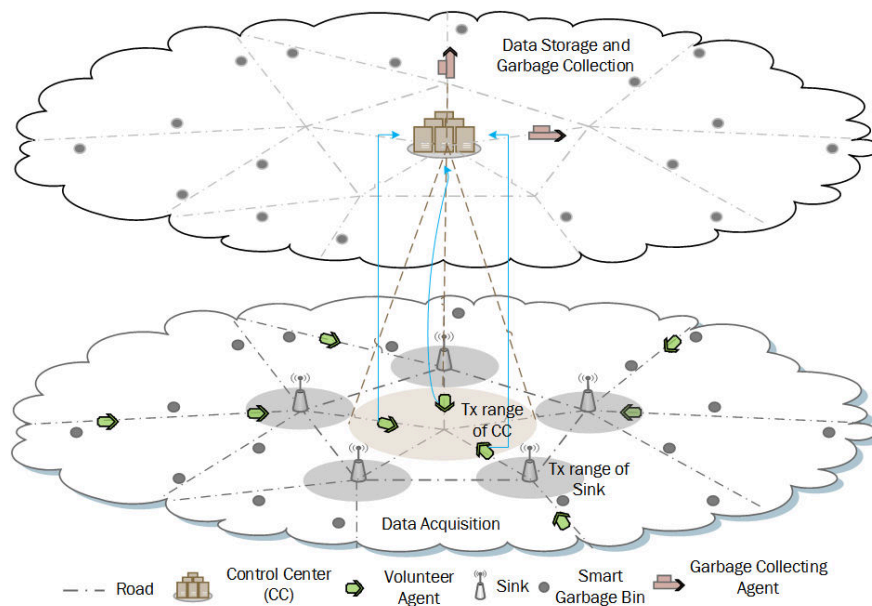
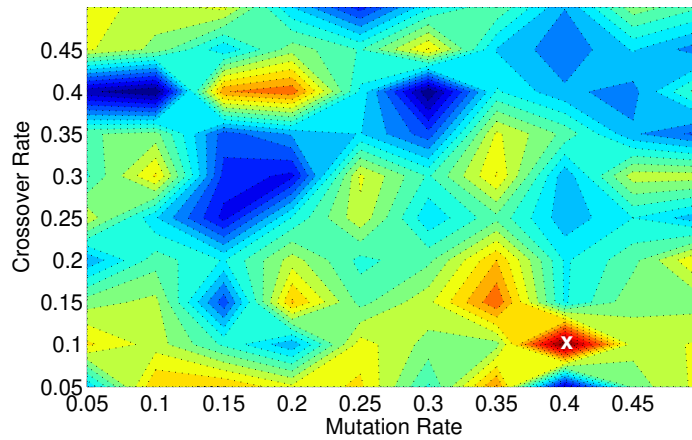
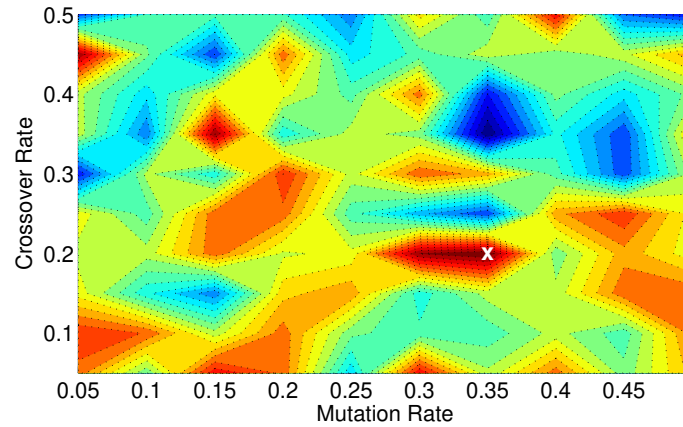


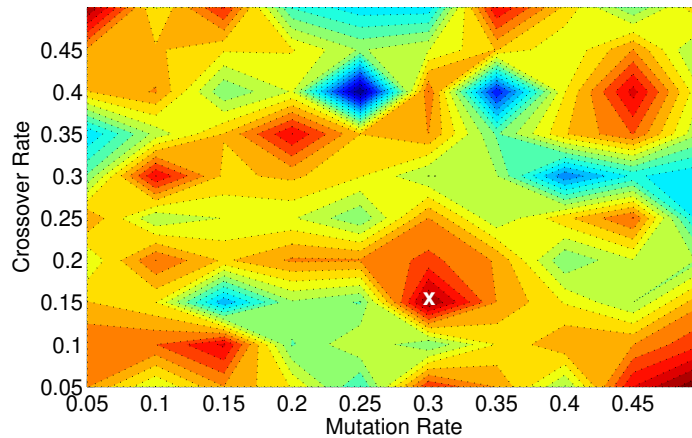
Figure 1: The two-tier architecture of the *CrAN*.



(a) Area: 500 m $\times$ 500 m

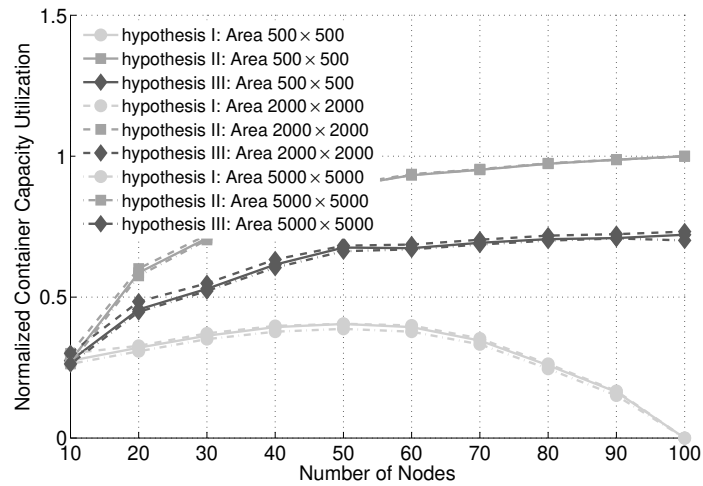


(b) Area: 2000 m $\times$ 2000 m

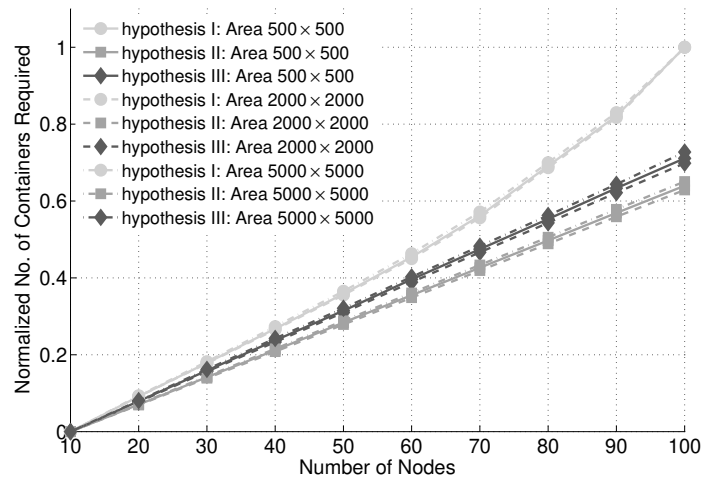


(c) Area: 5000 m $\times$ 5000 m

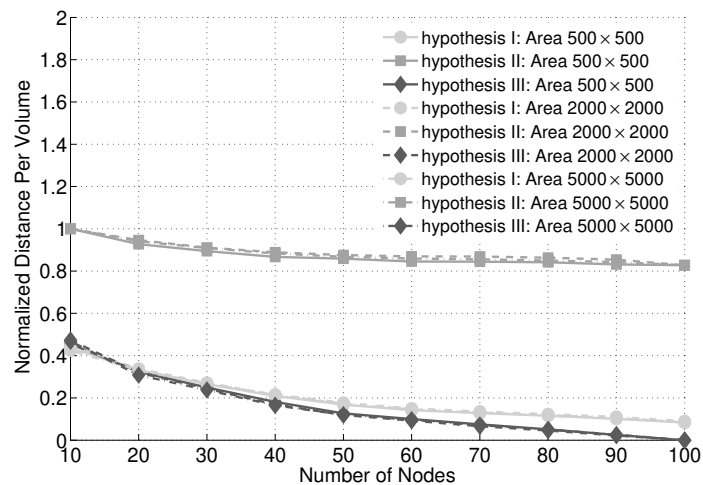
Figure 2: The impacts of various *mutation rates* and *crossover rates* on the utilization of the containers for diverse areas. The optimized *mutation rate* and *crossover rate* are pointed out using an **x** mark.



(a)



(b)



(c)

Figure 3: Results of three hypotheses for various metrics vs number of nodes.

Area	Mutation Rate	Crossover Rate
500 m $\times$ 500 m	0.4	0.1
2000 m $\times$ 2000 m	0.35	0.2
5000 m $\times$ 5000 m	0.3	0.15

Table I: Optimum *mutation rates* and *crossover rates* for the three preferred scenarios.



DSPACE

<https://dspace.org/>

Crowd Associated Network: Exploiting over Smart Garbage Management System

Azad, S.; Rahman, A.; Asyhari, A. Taufiq; Pathan, A. K.

2017-07-14

IEEE

[https://doi.org/ 10.1109/MCOM.2017.1600379](https://doi.org/10.1109/MCOM.2017.1600379)

Azad S, Rahman A, Asyhari AT, Pathan AK. (2017) Crowd Associated Network: Exploiting over Smart Garbage Management System. IEEE Communications Magazine, Volume 55, Issue 7, July 2017, pp. 186-192