Manuscript submitted to
AIMS' Journals
Volume **X**, Number **0X**, XX **200X**

Website: http://AIMsciences.org

pp. **X–XX**

OPTIMAL VISITING ORDER OF ISOLATED CLUSTERS IN DTNS TO MINIMIZE THE TOTAL MEAN DELIVERY DELAY OF BUNDLES

K. Habibul Kabir, Masahiro Sasabe and Tetsuya Takine

Department of Information and Communications Technology
Graduate School of Engineering, Osaka University
Suita 565-0871, Japan

Abstract. In delay tolerant networks (DTNs), the opportunity of communication among isolated networks (clusters) can be provided by a message ferry which moves around the network to proactively collect bundles and deliver them to a sink node. When there are lots of distant static clusters, the message ferry should visit them efficiently to minimize the mean delivery delay of bundles. In this paper, we propose an algorithm for determining the optimal visiting order of isolated static clusters in DTNs. We show that the minimization problem of the overall mean delivery delay in our system is reduced to that of the weighted mean waiting time in the conventional polling model. We then solve the problem with the help of an existing approach to the polling model and obtain a quasi-optimal balanced sequence representing the visiting order. Through numerical examples, we show that the proposed visiting order is effective when arrival rates at clusters and/or distances between clusters and the sink are heterogeneous.

1. **Introduction.** Challenged networks in delay tolerant networks (DTNs) [7,9] do not well match with the current end-to-end TCP/IP model. A store-carry-forward message delivery scheme [7] and custody transfer mechanism [10] are used in such kind of networks to assure reliable bundle transfers among nodes, where a bundle is the protocol data unit in DTNs. They perform a hop-by-hop reliable bundle transfer from a source node to the corresponding destination. To provide the opportunity of communication among isolated networks called *clusters*, Zhao et al. proposed message ferry schemes [25, 26], where a special mobile node proactively visits each cluster.

This kind of networks can be applied to sensor networking among physically distant regions and communications among rural areas without infrastructure. In such situations, the system periodically collects information from multiple isolated clusters. Note that each node in a cluster can directly/indirectly communicate with other cluster members through multi-hop communication but cannot communicate with nodes in other clusters due to long distances among them. It is usually assumed that there exists a fixed base station called *sink node*, which serves as a connector to the Internet or to other sink nodes. In such a scenario, a message ferry helps the inter-cluster communication by acting as a mediator between each cluster and the

2000 *Mathematics Subject Classification.* Primary: 90B18; Secondary: 68M20.

Key words and phrases. delay tolerant networks (DTNs), message ferry, optimal visiting order, polling model.

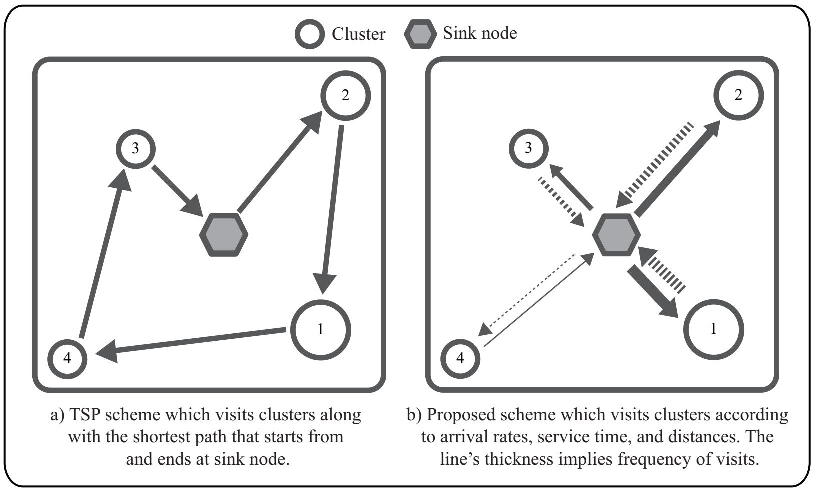

FIGURE 1. Example of message ferry's visiting sequence in TSP-based routing and the proposed scheme. Each arrow indicates the movement of message ferry and the size of each cluster is proportional to the arrival rate of bundles.

outer world through the sink node as shown in Figure 1. Thus the problem is to find an efficient route along which the message ferry visits isolated clusters and the sink node.

Suppose service times (i.e., times needed for collecting bundles from clusters and unloading them to the sink node) are negligible. In such a case, the shortest cyclic route seems to be a natural solution, which can be obtained by solving the traveling salesman problem (TSP) [18]. The shortest cyclic route starts from the sink node, passes through each cluster at once, and finally returns the sink node as shown in Figure 1(a). Therefore in terms of the mean waiting time, all clusters are treated fairly in this strategy. In practice, however, arrival rates of bundles are different among clusters and service times are not negligible. In such situations, the TSP-based shortest cyclic route strategy potentially has two drawbacks: 1) The time spent for one cycle increases with the number of clusters, and 2) if the arrival rates of bundles at clusters are different from each other and service times are not negligible, bundles in clusters with high arrival rate have to wait for long time to be delivered to the sink node while less important visits to clusters with a few bundles also take place.

For the first issue, the whole system can be divided into multiple groups such that each group consists of a sink node, clusters, and at least one message ferry. In what follows, we briefly introduce the guidelines for making groups. Suppose that the number and positions of clusters are fixed and the number of sink nodes and message ferries are limited in order to lower the expenses of the system. To reduce the traveling distance, a group should consist of physically-close clusters. The adequate place of the sink node depends on the route-setup strategy and the sink node can be shared by multiple groups. Note here that in any system, the overall traffic intensity should be less than one, i.e., the total arrival rate of bundles in the system should be less than the overall collection rate of bundles. Therefore the number of clusters in a group and the number of message ferries to visit them are determined depending on the system scale and the traffic intensity: the number of message ferries required for a group should be at least larger than the total offered load in that group.

The second issue is the main topic of this paper. In general, the visiting order of clusters by the message ferry should be determined based on arrival rates, service times of bundles, and one-way traveling times between clusters and the sink node. We assume that all the isolated clusters are significantly apart from each other, e.g., in kilometer range. As a result, the *inter-visit time* of a cluster (i.e., the interval time between a departure of the message ferry from the cluster and the next return of the message ferry to that cluster) naturally becomes long. In such a situation, when the message ferry visits each cluster, it would find bundles that wait for a long time with high probability. Therefore, in order to reduce the delivery delay, it might be reasonable to deliver those bundles to the sink node directly, as shown in Figure 1(b), rather than to visit other clusters while carrying them.

This inter-cluster communication of the message ferry can be best studied using a polling model [22,23], where the message ferry, clusters, and bundles are regarded as the server, stations, and customers, respectively, and "service" means that the message ferry collects (unloads) bundles from (to) the cluster (sink node). The optimization of the polling order are studied in [3,5,19], which is equivalent to find an optimal visiting order of stations, which minimizes the expected waiting time of all customers.

In the study of polling models, the waiting time (i.e., the length of an interval from the generation of a bundle to the instant at which its service starts) is a primary performance measure of interest. On the other hand, in our system, we are interested in the *delivery delay*, which is defined as the time interval from the generation of a bundle to the completion of its delivery to the sink node. In this paper, we show that the mean delivery delay of bundles is given in terms of the weighted sum of the mean waiting times of bundles at respective clusters. We then apply the optimization technique in [3,4,6] to our system and obtain a quasi-optimal visiting order that minimizes the total mean delivery delay of the system. Roughly speaking, clusters with high arrival rate and/or close to the sink node are visited more frequently than others in the optimal visiting order.

Besides, the intra-cluster communication can also be minimized by accumulating bundles to a limited number of aggregators, with the help of the self-organized data aggregation technique, which is our previous work in [12–14]. The number of aggregators can be controlled by the amount of energy supplied by the message ferry. As a result, the message ferry needs to collect the bundles only from the aggregators.

The rest of this paper is organized as follows. In Section 2. we review the related work. We describe the mathematical model in Section 3. Sections 4 provides the optimization problem formulation and its solution method. Section 5 shows the result of simulation experiments and demonstrates the effectiveness of our scheme. Finally we conclude the paper in Section 6.

2. Related Work.

Zhao et al. first applied the TSP-based routing to highly-partitioned ad hoc wireless networks [24, 25] by introducing a message ferry as the traveling salesman. A single ferry is used to communications among fixed nodes in partitioned networks [24,25] and a heuristic method for finding the visiting order is shown. In [25], they also extended their message ferry scheme to that for systems with mobile nodes.

In [2], Ammar et al. focused on the buffer size required for each node when the message ferry travels along the shortest cyclic path. They presented an algorithm for finding the visiting order that minimizes the maximum required buffer size among nodes. This problem can be regarded as a variant of the TSP problem under the assumptions of identical arrival rate and negligible service time, and minimizing the buffer size is equivalent to minimizing the mean waiting time for the ferry visiting. The objective is similar to ours but this approach is not suitable for scenarios with heterogeneous arrival rate and non-negligible service time.

Some works tried to improve the scalability and robustness of the system with the help of multiple message ferries, e.g., multiple ferries for a single route [2] and multiple ferries for multiple routes [26]. They considered the message ferry assignment to nodes and route making in such a way that the number of message ferries is minimized when the number of nodes and the upper bound of the waiting time are given. Miura et al. considered clustering of highly-partitioned wireless networks [20]. They assume that there are several partitioned clusters in which physically-close nodes exist; which is similar to our scenario in Figure 1. They applied the TSP-based routing by setting the visiting point of the message ferry to the center of each cluster.

All of the above mentioned studies assume that arrival rates are identical among nodes and service times are negligible. In practical situations, however, these assumptions do not necessarily hold. In such situations, finding the shortest cyclic path is insufficient to achieve minimizing the overall mean delivery delay of bundles. Kavitha et al. first tackled this problem by applying the polling model. In [15–17], they assumed message ferry-based wireless LANs, where nodes are well scattered over the area and designed an optimal route (among some given class of trajectories, e.g., circle and line) that minimizes the overall expected waiting times. The message ferry can serve nodes within its transmission range at any point on the path. Their approach can also support both uplink and downlink services.

Although our objective is similar to [15–17], the target scenario is totally different. It is assumed in [15–17] that nodes can exist at any point in an area according to a known probability distribution, while we assume that there are partitioned clusters, each of which consists of physically-close nodes. If the approach in [15–17] is applied to our scenario, it requires many paths to cover the whole area, each of which is a circle/line trajectory supported by a single message ferry. In addition, in [15–17] a cyclic policy is used: The server visits the stations in a predetermined cyclic order. Hence, if clusters with high arrival rates and those with low arrival rates coexist in the area, it will not be effective. On the other hand, our proposed scheme applies a non-cyclic policy, taking account of the arrival rate and location of each cluster.

3. **Model.** Suppose the system consists of N clusters labeled 1 to N, the sink node, and a message ferry, all of which have buffers of infinite capacity. The message ferry periodically visits clusters according to a predefined visiting order (i.e., a polling table). When the message ferry arrives at a cluster, it serves bundles under the exhaustive service discipline, i.e., bundles are transmitted successively to the message ferry, and when there are no waiting bundles, the message ferry leaves the cluster. It is known that the exhaustive service discipline has the best performance in terms of the overall mean waiting time [22]. After collecting all bundles at the

cluster, the message ferry immediately returns to the sink node, unloads all bundles it carries to the sink node, and goes to the next cluster.

We define S_i $(i \in \mathcal{N})$ as the one-way traveling time between cluster i and the sink node, where $\mathcal{N} = \{1, 2, \ldots, N\}$. We assume that S_i $(i \in \mathcal{N})$ is constant because of the fixed physical route and the constant speed of the message ferry. Bundles at cluster i $(i \in \mathcal{N})$ are generated according to a Poisson process with rate λ_i and all of them are stored at cluster i. Service times X_i $(i \in \mathcal{N})$ of bundles at cluster i follow a general distribution with finite mean x_i and second moment $x_i^{(2)}$. Note that X_i corresponds to the transmission time of a randomly chosen bundle at cluster i. We assume that high speed channels are available at the sink node, and therefore the unloading time of bundles at the sink node is assumed to be negligible.

Let $\rho_i = \lambda_i x_i$ $(i \in \mathcal{N})$ denote the traffic intensity at cluster i. The overall generation rate of bundles and the overall traffic intensity are denoted by $\lambda = \sum_{i \in \mathcal{N}} \lambda_i$ and $\rho = \sum_{i \in \mathcal{N}} \rho_i$, respectively. We assume that $\rho < 1$, which ensures the stability of the system [22]. In what follows, the system is assumed to be in steady state.

4. **Optimization problem formulation and its solution method.** We define the delivery delay of bundles as the time interval from the generation of the bundle to the instant at which it is delivered to the sink node. Let $W_{\text{deliver},i}$ $(i \in \mathcal{N})$ denote the delivery time of a randomly chosen bundle generated at cluster i. The goal of this section is to formulate and solve a mathematical program to find the optimal visiting order of clusters, which minimizes the overall mean delivery delay $\mathrm{E}[W_{\text{total}}]$:

$$\mathrm{E}[W_{\text{total}}] = \sum_{i \in \mathcal{N}} \frac{\lambda_i}{\lambda} \mathrm{E}[W_{\text{deliver},i}]. \tag{1}$$

As we will see, our problem is reduced to the minimization problem of a weighted sum of mean waiting times of a polling model. Without loss of generality, we assume that bundles at each cluster are served on an FCFS basis, because $\mathrm{E}[W_{\text{deliver},i}]$ $(i \in \mathcal{N})$ is irrelevant to the service order of waiting bundles at cluster i in the exhaustive service discipline.

We first divide $W_{\text{deliver},i}$ $(i \in \mathcal{N})$ into two disjoint parts T_i^* and S_i, where T_i^* denotes the sojourn time of a randomly chosen bundle at cluster i. See Figure 2. It then follows that

$$\mathrm{E}[W_{\text{deliver},i}] = \mathrm{E}[T_i^*] + S_i, \quad i \in \mathcal{N}. \tag{2}$$

In the exhaustive service policy, the server has to stay at each cluster until it finishes collecting all bundles. Therefore $\mathrm{E}[T_i^*]$ is considered as the mean delay cycle with an initial delay $W_{\text{wait},i} + X_i$, where $W_{\text{wait},i}$ denotes the waiting time of a randomly chosen bundle at cluster i (see Figure 2). We then have [8]

$$\mathrm{E}[T_i^*] = \frac{\mathrm{E}[W_{\text{wait},i}] + x_i}{1 - \rho_i}. \tag{3}$$

Note that $W_{\text{wait},i}$ is identical to the waiting time in the ordinary polling model.

It then follows from 1, 2, and 3 that

$$\begin{aligned} \mathrm{E}[W_{\text{total}}] &= \frac{1}{\lambda} \sum_{i \in \mathcal{N}} \lambda_i \left(\frac{\mathrm{E}[W_{\text{wait},i}]}{1 - \rho_i} + \frac{x_i}{1 - \rho_i} + S_i \right) \\ &= \sum_{i \in \mathcal{N}} c_i \mathrm{E}[W_{\text{wait},i}] + \alpha, \end{aligned} \tag{4}$$

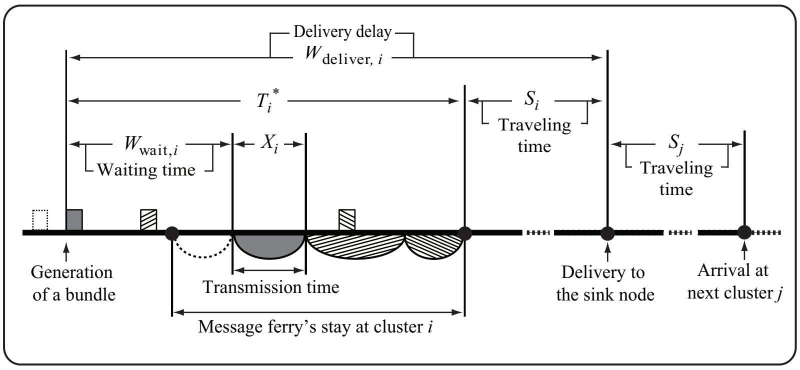

FIGURE 2. Timing chart (exhaustive service policy). When the message ferry arrives at cluster i, there are already three bundles waiting for the service. During the service for them, one bundle is further generated. When there is no bundle to be served, the message ferry leaves cluster i and visits the next cluster via the sink node.

where

$$c_i = \frac{\lambda_i}{(1-\rho_i)\lambda}, \quad i \in \mathcal{N}, \tag{5}$$

$$\alpha = \frac{1}{\lambda}\sum_{i\in\mathcal{N}}\left(\frac{\rho_i}{1-\rho_i} + \lambda_i S_i\right).$$

Because α is constant regardless of the visiting order of clusters, the minimization of $\mathrm{E}[W_{\text{total}}]$ is equivalent to that of the weighted sum of the mean waiting times $\mathrm{E}[W_{\text{wait},i}]$ in the exhaustive-service polling model.

$$\text{minimize} \quad \sum_{i\in\mathcal{N}} c_i \mathrm{E}[W_{\text{wait},i}]. \tag{6}$$

In the rest of this section, we follow the lower bound approach in [3, 6], and obtain an approximate solution of 6.

Under the exhaustive service discipline, the mean waiting time $\mathrm{E}[W_{\text{wait},i}]$ ($i \in \mathcal{N}$) at cluster i takes a form: [3]

$$\mathrm{E}[W_{\text{wait},i}] = \frac{\lambda_i x_i^{(2)}}{2(1-\rho_i)} + \frac{v_i^{(2)}}{2v_i}, \quad i \in \mathcal{N}, \tag{7}$$

where v_i and $v_i^{(2)}$ ($i \in \mathcal{N}$) denote the first and second moments of interval lengths from departures of the message ferry from cluster i to the next arrival instants. Because $v_i^{(2)} \geq v_i^2$, the weighted sum of $\mathrm{E}[W_{\text{wait},i}]$ is bounded from below:

$$\sum_{i\in\mathcal{N}} c_i \mathrm{E}[W_{\text{wait},i}] \geq \frac{1}{2}\sum_{i\in\mathcal{N}} c_i \left(\frac{\lambda_i x_i^{(2)}}{1-\rho_i} + v_i\right). \tag{8}$$

We adopt the approach of [3] minimizing the lower bound given by the right hand side of 8, instead of $\sum_{i\in\mathcal{N}} c_i \mathrm{E}[W_{\text{wait},i}]$.

Let q_i ($i \in \mathcal{N}$) denote the mean number of visits at cluster i per unit time. Because q_i^{-1} ($i \in \mathcal{N}$) is equal to the mean cycle time $\mathrm{E}[C_i] = v_i/(1-\rho_i)$ [22], we have

$$v_i = \frac{1-\rho_i}{q_i}. \quad i \in \mathcal{N}. \tag{9}$$

Substituting 9 into the right hand side of 8, rearranging terms with 5, and ignoring constant factors and terms, we obtain the objective function $f(\boldsymbol{q})$ of the minimization problem.

$$f(\boldsymbol{q}) = \sum_{i\in\mathcal{N}} \frac{\lambda_i}{q_i}, \tag{10}$$

where $\boldsymbol{q} = (q_1, q_2, \ldots, q_N)$.

The constraints on $\boldsymbol{q}$ are obtained as follows. First of all, $q_i > 0$ for all i ($i \in \mathcal{N}$). Furthermore

$$\rho + 2\sum_{j\in\mathcal{N}} S_j q_j = 1,$$

should hold. Note that $2S_jq_j$ ($j \in \mathcal{N}$) denote the time-average probability that the message ferry is traveling between the sink node and cluster i. Because ρ represents the probability of one of the clusters being served. Therefore the sum of them should be equal to one. In summary, we have the following Problem P.

$$\begin{aligned} P: \ &\text{minimize} \ \ f(\boldsymbol{q}), \\ &\text{subject to} \ \rho + 2\sum_{j\in\mathcal{N}} S_j q_j = 1, \\ &\qquad\qquad q_i > 0, \quad i \in \mathcal{N}, \end{aligned} \tag{11}$$

Problem P is easy to solve with the Lagrange multipliers method. We define $L(\boldsymbol{q}, \theta)$ as

$$L(\boldsymbol{q}, \theta) = f(\boldsymbol{q}) + \theta(\rho + 2\sum_{j\in\mathcal{N}} S_j q_j - 1),$$

where $\theta > 0$ denotes the Lagrange multiplier. We then have

$$\frac{\partial L}{\partial q_i} = -\frac{\lambda_i}{q_i^2} + 2\theta S_i = 0, \quad i \in \mathcal{N},$$

from which, it follows that

$$q_i = \sqrt{\frac{\lambda_i}{2\theta S_i}} > 0, \quad i \in \mathcal{N}. \tag{12}$$

q_i in 12 should satisfy 11, so that

$$\rho + \sqrt{\frac{1}{\theta}} \cdot \sum_{j\in\mathcal{N}} \sqrt{2\lambda_j S_j} = 1,$$

from which, it follows that

$$\sqrt{\frac{1}{\theta}} = \frac{1-\rho}{\sum_{j\in\mathcal{N}} \sqrt{2\lambda_j S_j}},$$

and therefore we obtain from 12

$$q_i = \frac{1-\rho}{\sum_{j\in\mathcal{N}} \sqrt{2\lambda_j S_j}} \cdot \sqrt{\frac{\lambda_i}{2S_i}}, \quad i \in \mathcal{N}. \tag{13}$$

Let p_i ($i \in \mathcal{N}$) denote the ratio of the message ferry's visit to cluster i. It then follows from 13 that

$$p_i = \frac{q_i}{\sum_{j\in\mathcal{N}} q_j} = \frac{\sqrt{\lambda_i/S_i}}{\sum_{j\in\mathcal{N}} \sqrt{\lambda_j/S_j}} \tag{14}$$

14 indicates that the optimal frequency of visits to clusters is determined only by the ratio of the arrival rate λ_i ($i \in \mathcal{N}$) to the one-way travel times S_i ($i \in \mathcal{N}$), and it is independent of service times x_i ($i \in \mathcal{N}$). Thus, in our proposed scheme, the message ferry frequently visits clusters with high arrival rates and/or small distances to the sink node.

The next is to find the visiting order of clusters, whose frequency is given by 14. When non-periodic orders are allowed, this problem is called balanced sequence/words and examined in [1, 21], where each cluster is spaced as evenly as possible in the sequence. In our system, however, the target frequency p_i is an approximate one and the frequency of visits to each cluster is not exactly identical to the target frequency. Taking account of it, we use the following procedure for determining the visiting order of clusters, which is a combination of proposals in [4, 6].

Step 1: *Determination of the cycle length and the frequency of visits.* We borrow an idea in [6]. Let M denote an positive integer representing the cycle length in terms of the number of visited clusters. Also, let m_i ($i \in \mathcal{N}$) denote the number of visits to cluster i in a cycle. We define $\mathrm{int}(x)$ ($x > 0$) as

$$\mathrm{int}(x) = \begin{cases} \lfloor x \rfloor, & x - \lfloor x \rfloor < 0.5, \\ \lceil x \rceil, & \text{otherwise.} \end{cases}$$

For $m = N, N+1, \ldots$, we seek minimum $m = m^*$ such that

$$\mathrm{int}(m^* p_i) \geq 1, \quad i \in \mathcal{N},$$

$$|m^* p_i - \mathrm{int}(m^* p_i)| \leq \epsilon, \quad i \in \mathcal{N},$$

and

$$\sum_{i\in\mathcal{N}} \mathrm{int}(m^* p_i) = m^*,$$

where ϵ is a predetermined parameter. We then set

$$M = m^*, \qquad m_i = \mathrm{int}(m^* p_i) \quad (i \in \mathcal{N}).$$

Step 2: *Determination of the visiting order.* We use the procedure given in Appendix C of [4], which is summarized as follows. Let $\mathcal{M} = \{m_i;\ i \in \mathcal{N}\}$ denote the set of the numbers of visits to respective nodes in a cycle. For $r \in \mathcal{M}$, let $\mathcal{I}^{(r)} = \{i \in \mathcal{N};\ m_i = r\}$ denote the set of indices of clusters visited r times in a cycle. Furthermore, let $Q^{(r)}$ ($r \in \mathcal{M}$) denote a repeated string of symbols in $\mathcal{I}^{(r)}$, where each symbol appears r times with equal distance. For example, if $\mathcal{I}^{(3)} = \{2, 4\}$, $Q^{(3)}$ is given by 242424. For any string A, let $|A|$ denote the length of string A.

1. Prepare $Q^{(r)}$ for all $r \in \mathcal{M}$.
2. Choose $r \in \mathcal{M}$ and $\mathcal{D} = \{r\}$. Let $P = Q^{(r)}$.
3. If $\mathcal{M} \setminus \mathcal{D} = \emptyset$, stop the procedure, where P gives the visiting order of clusters.
4. Choose $r \in \mathcal{M} \setminus \mathcal{D}$ and $\mathcal{D} := \mathcal{D} \cup \{r\}$. We then merge $Q^{(r)}$ into P, and the resulting string is denoted by $P^{(r)}$, where $|P^{(r)}| = |P| + |Q^{(r)}|$. The rule of

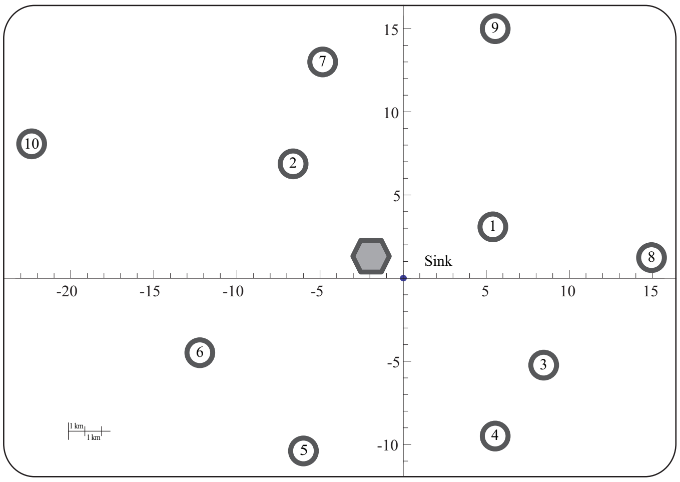

FIGURE 3. Random layout model ($N = 10$, S_1=600, S_2=900, S_3=1,000, S_4=1,100, S_5=1,200, S_6=1,300, S_7=1,400, S_8=1,500, S_9=1,600, S_{10}=2,400, $\overline{S} = 1,300$, $C = 13,304.94$). The cluster IDs are assigned in an ascending order of the distance from the sink node.

this merging operation is as follows. The kth symbol in $Q^{(r)}$ is identical to the $(k + d(k))$th symbol in $P^{(r)}$, where

$$d(k) = \text{int}((k-1)|P|/|Q^{(r)}|), \quad k = 1, 2, \ldots, |Q^{(r)}|.$$

The rest of symbols in $P^{(r)}$ is identical to those in P, and the order of those symbols are identical in P and $P^{(r)}$.

5. Let $P = P^{(r)}$ and go to step 3.

5. **Simulation results.** In this section, we evaluate the performance of our proposed scheme through simulation experiments.

5.1. **Simulation setting.** We consider a system composed of a sink node and ten isolated clusters ($N = 10$). We use two kinds of cluster layouts: Circle-based layout and random layout models. In the circle-based layout model, ten clusters are placed equally dividing a circle with a radius of 13km, and the sink node is located at the center of the circle. On the other hand, the random layout model is illustrated in Figure 3. The circle-based layout and random layout models correspond to the cases of identical and different one-way traveling times S_i ($i = 1, 2, \ldots, 10$), respectively. We assume that the message ferry travels at a fixed speed of 10m/s (i.e., 36km/h). We denote the mean one-way traveling time by $\overline{S} = N^{-1}\sum_{j\in\mathcal{N}} S_j$, which is fixed to $1,300$ [s] in any case. Transmission times of bundles at all clusters are independent and identically distributed according to an exponential distribution with mean $x_i = 1$ [s]. For the settings of λ_i ($i = 1, 2, \ldots, 10$), we consider four cases, one is the homogeneous case and other three cases are heterogeneous, as shown in Table 1. In the following results, we mainly examine how λ_i and S_i affect the mean delivery delay $\mathrm{E}[W_{\text{total}}]$ (sec).

We compute the visiting order of clusters according to the procedure in Section 4, where ϵ is set to be 0.4. Recall that in the proposed visiting order, the message ferry

TABLE 1. Scenarios of λ_i (N=10, λ=0.76 [1/s]).

Case		λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8	λ_9	λ_{10}
Hetero.	Descend.	.30	.10	.08	.07	.06	.05	.04	.03	.02	.01
	Random	.02	.05	.30	.10	.08	.07	.04	.01	.03	.06
	Ascend.	.01	.02	.03	.04	.05	.06	.07	.08	.10	.30
Homogeneous		.076									

TABLE 2. Mean delivery delay E[W_{total}] in the circle-based layout model with homogeneous arrival rates (N=10).

	Visiting order	E[W_{total}]
Proposal	1-2-3-4-5-6-7-8-9-10-	44,679.25±87.26
Cyclic	1-2-3-4-5-6-7-8-9-10-	44,679.25±87.26
TSP	sink-1-2-3-4-5-6-7-8-9-10-sink-	38,290.08±49.84

returns to the sink node before visiting the next cluster, as shown in Figures 1 (b). For the sake of comparison, we also consider a cyclic visiting order and a TSP-based routing (cf. Figure 1 (a)). In the cyclic visiting order, the message ferry visits clusters one by one through the sink node, i.e., 1–sink–2–sink–$\cdots$. On the other hand, in the TSP-based routing, the message ferry visits clusters according to the shortest cyclic path that starts from and ends with the sink node. Let C denote the traveling time of one cycle in the TSP-based routing. We then have $C = 9830.92$ and $C = 13,304.94$ in the circle-based and random layout models, respectively. For each simulation experiment, we discard the initial interval of 50,000 seconds as transient period and collect data in the subsequent interval from 50,000 to 6,000,000 (sec). All simulation results are presented with 95% confidence intervals, based on ten independent simulation runs.

5.2. **Performance evaluation.** We first evaluate the performance of three schemes in the circle-based layout, where S_i are homogeneous. Table 2 shows the visiting order and the mean delivery delay E[W_{total}] when λ_i's are homogeneous. Note that our proposed scheme is identical with the cyclic scheme in this case because λ_i/S_i's ($i \in \mathcal{N}$) are identical. We observe that the TSP-based routing has the smallest E[W_{total}] in this scenario.

Next, we examine the influence of the heterogeneity of λ_i. Table 3 shows the result in the circle-based layout, where λ_i's are set according to the descending/random/ascending arrival rate scenarios in Table 1. Note that the descending and ascending scenarios are both extremes and therefore in each scheme, the random arrival rate scenario yields the second best mean delivery delay. Even though the TSP-based routing has the smallest delivery delay, the difference between our proposed scheme and TSP-based routing becomes small, compared with the homogeneous case in Table 2.

The small difference of the results within our proposed scheme comes from a specific implementation of the procedure for generating the visiting order, where clusters are always arranged in an ascending order of their indices. If we arranged clusters in the descending order of their indices in the case of the ascending arrival rate scenario, we would have the visiting order of 10-9-8-7-10-6-5-9-4-10-3-2-1- and the result would be identical to that in the descending arrival rate scenario. The

TABLE 3. Mean delivery delay $\mathrm{E}[W_{\mathrm{total}}]$ in the circle-based layout model with heterogeneous arrival rates (N=10).

	λ_i	Visiting order	$\mathrm{E}[W_{\mathrm{total}}]$
Proposal	Descend.	1-2-3-4-1-5-6-2-7-1-8-9-10-	38,004.41±63.15
	Random	3-4-1-2-3-5-6-4-7-3-8-9-10-	38,142.16±38.95
	Ascend.	10-9-1-2-10-3-4-9-5-10-6-7-8-	38,246.43±55.56
Cyclic	Descend.	1-2-3-4-5-6-7-8-9-10-	45,353.64±61.46
	Random	1-2-3-4-5-6-7-8-9-10-	45,434.70±63.28
	Ascend.	1-2-3-4-5-6-7-8-9-10-	45,654.29±53.12
TSP	Descend.	sink-1-2-3-4-5-6-7-8-9-10-sink-	36,983.58±82.62
	Random	sink-1-2-3-4-5-6-7-8-9-10-sink-	36,384.11±33.85
	Ascend.	sink-1-2-3-4-5-6-7-8-9-10-sink-	33,744.17±68.49

TABLE 4. Mean delivery delay $\mathrm{E}[W_{\mathrm{total}}]$ in the circle-based layout model with one heavily loaded cluster ($N = 10$, $\lambda = 0.76$, $\lambda_1 = 0.9\lambda$, $\lambda_i = 0.1\lambda/9$ $(i = 2, 3, \ldots, 10)$).

	Visiting order	$\mathrm{E}[W_{\mathrm{total}}]$
Proposal	1-2-1-3-1-4- $\cdots$ -1-9-1-10-	7072.23±39.45
TSP	sink-1-2-3-4-5-6-7-8-9-10-sink-	14901.43±67.89
	sink-10-9-8-7-6-5-4-3-2-1-sink-	8285.17±56.17

cyclic visiting order is essentially identical to the ordinary polling model, and the mean waiting time in asymmetric polling models is known to depend on the visiting order [11].

Recall that neither the TSP-based routing nor the cyclic visiting order take account of arrival rates at clusters. Compared with the cyclic visiting order, the difference between the mean delivery delay of the descending and ascending arrival rate scenarios in TSP-based routing is significantly large by the following reason. In the TSP-based routing, the message ferry visits clusters successively while carrying collected bundles with it, before returning to the sink node. In the descending arrival rate scenario, the message ferry tends to collect many bundles at clusters with small indices (i.e., in the former part of the cycle), and it carries them while visiting other lightly-loaded clusters with large indices. In this way, many bundles suffer from long delay, which leads to a significant increase of the mean delivery delay in the descending arrival rate scenario.

Note here that the TSP-based routing is not always superior to our proposed scheme. For example, suppose 90% of traffic is generated at cluster 1 and the rest is divided evenly among nine other clusters, while keeping the total traffic intensity fixed to λ=0.76. Table 4 shows the result. The mean delivery delay in the TSP-based routing is greater than that in our proposed scheme and in the TSP-based routing, the direction at which the message ferry moves affects the performance significantly.

We now turn our attention to the random layout model in Figure 3, where distances between clusters and between the sink node and respective clusters are different. Recall that cluster indices are set in the ascending order of S_i $(i \in \mathcal{N})$. Table 5 shows the result for the homogeneous arrival rate scenario. Our proposed

TABLE 5. Mean delivery delay $\mathrm{E}[W_{\mathrm{total}}]$ in the random layout model with homogeneous arrival rates.

	Visiting order	$\mathrm{E}[W_{\mathrm{total}}]$
Proposal	1-2-7-3-4-5-8-6-1-2-9-3-4-5-10-6-	44,325.52±56.87
Cyclic	1-2-3-4-5-6-7-8-9-10-	49,357.16±49.41
	10-9-8-7-6-5-4-3-2-1-	49,416.77±52.19
TSP	sink-9-7-2-10-6-5-4-3-8-1-sink-	45,842.34±55.98
	sink-1-8-3-4-5-6-10-2-7-9-sink-	45,896.17±78.30

TABLE 6. Mean delivery delay $\mathrm{E}[W_{\mathrm{total}}]$ in the random layout model with heterogeneous arrival rates.

	λ_i	Visiting order	$\mathrm{E}[W_{\mathrm{total}}]$
Proposal	Descend.	1-2-3-7-1-4-5-1-2-8-6-1-3-9-1-2-4-5-1-10-6-	31,749.81±39.45
	Random	3-2-1-4-3-5-7-6-8-3-2-4-9-3-5-6-10-	33,748.28±45.10
	Ascend.	10-1-2-3-4-5-10-6-7-8-9-	41,672.71±70.32
Cyclic	Descend.	1-2-3-4-5-6-7-8-9-10-	45,954.46±92.14
		10-9-8-7-6-5-4-3-2-1-	45,922.72±53.58
	Random	1-2-3-4-5-6-7-8-9-10-	46,237.65±48.98
		10-9-8-7-6-5-4-3-2-1-	46,478.53±87.18
	Ascend.	1-2-3-4-5-6-7-8-9-10-	46,961.42±37.85
		10-9-8-7-6-5-4-3-2-1-	46,982.49±34.47
TSP	Descend.	sink-9-7-2-10-6-5-4-3-8-1-sink-	45,053.78±69.97
		sink-1-8-3-4-5-6-10-2-7-9-sink-	47,838.46±77.27
	Random	sink-9-7-2-10-6-5-4-3-8-1-sink-	45,176.45±78.48
		sink-1-8-3-4-5-6-10-2-7-9-sink-	47,615.67±32.09
	Ascend.	sink-9-7-2-10-6-5-4-3-8-1-sink-	46,776.59±83.44
		sink-1-8-3-4-5-6-10-2-7-9-sink-	45,298.15±32.95

scheme is superior to the TSP-based routing, which indicates that serving clusters close to the sink node more frequently is beneficial to the reduction of the overall mean delivery delay.

Finally, Tables 6 shows the results when both λ_i and S_i are heterogeneous. In all arrival rate scenarios, our proposed scheme shows the better performance than the TSP-based routing, and the difference between the mean delivery delays in our proposed scheme and the TSP-based routing depends on the scenarios. In general, a large variation in $\sqrt{\lambda_i/S_i}$ $(i \in \mathcal{N})$ yields the large variance of p_i $(i \in \mathcal{N})$, and it leads to a long visiting order sequence. Performance of our scheme has a strong correlation to the length of the visiting order sequence and scenarios yielding long sequences are more preferable for our proposed scheme.

6. **Conclusion.** In this paper, we focused on a system where a message ferry collects bundles from isolated clusters and delivers those to the sink node, where transmission times of bundles are not negligible. To minimize the total mean delivery delay of bundles, we proposed an algorithm for obtaining a quasi-optimal visiting order of clusters, with the help of the optimization technique of the conventional

polling model. Through simulation experiments, we showed that the proposed visiting order can perform well, especially when the arrival rate and/or distances are heterogeneous.

Acknowledgments. This research was supported by "Global COE (Centers of Excellence) Program" of the Ministry of Education, Culture, Sports, Science and Technology, Japan, as a part of "Ambient Networking" research group, Grant-in-Aid for Young Scientists (B) of JSPS under Grant No. 22700070, and Grant-in-Aid for Scientific Research (C) of JSPS under Grant No. 22500056.

REFERENCES

[1] E. Altman, B. Gaujal, and A. Hordijk, *Balanced sequences and optimal routing*, Journal of ACM, **47** (2000), 752–775.
[2] M. Ammar, D. Chakrabarty, A. Sarma, S. Kalyanasundaram, and R. Lipton, *Algorithms for message ferrying on mobile ad hoc networks,*, in "Proc. IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2009)," (2009), 13–24.
[3] D.J. Bertsimas and H. Xu, *Optimization of polling systems and dynamic vehicle routing problems on networks*, in "Technical Report, OR 283-93, MIT," (1993).
[4] S. C. Borst, O. J. Boxma, J. H. A. Harink, and G. B. Huitema, *Optimization of fixed time polling schemes*, Telecommunication Systems, **3** (1994), 31–59.
[5] O. J. Boxma, W. P. Groenendijk, and J. A. Weststrate, *A pseudoconservation law for service systems with a polling table*, IEEE Transactions on Communications, **38** (1990), 1865–1870.
[6] O.J. Boxma, H. Levy, and J.A. Westrate, *Efficient visit orders for polling systems*, Performance Evaluation, **18** (1993), 103–123.
[7] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall and H. Weiss, *Delay tolerant network architecture*, work in progress as an IETF RFC 4838 draft. Available from: http://www.ietf.org/rfc/rfc4838.txt.
[8] R. W. Conway, W. L. Maxwell, and L. W. Miller, "Theory of Scheduling," Addison Wesley, Reading, MA, 1967.
[9] K. Fall, *A delay-tolerant network architecture for challenged internets*, in "Proc. ACM SIGCOMM," (2003), 27–34.
[10] K. Fall and W. Hong, *Custody transfer for reliable delivery in delay tolerant networks*, in "Technical Report, IRB-TR-03–030, Intel Research Berkeley," (2003).
[11] M. J. Ferguson and Y. J. Aminetzah, *Exact results for nonsymmetric token ring systems*, IEEE Transactions on Communications, **COM-33** (1985), 223–231.
[12] K.H. Kabir, M. Sasabe, and T. Takine, *Design and analysis of self-organized data aggregation using evolutionary game theory in delay tolerant networks*, in "Proc. 3rd IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications," (2009).
[13] K.H. Kabir, M. Sasabe, and T. Takine, *Evolutionary game theoretic approach to self-organized data aggregation in delay tolerant networks*, IEICE Transactions on Communications, **E93-B** (2010), 490–500.
[14] K.H. Kabir, M. Sasabe, and T. Takine, *Self-organized data aggregation among selfish nodes in an isolated cluster*, in "Proc. 5th International ICST Conference on Bio-Inspired Models of Network, Information and Computing Systems," (2010).
[15] V. Kavitha and E. Altman, *Queuing in space: Design of message ferry routes in static ad hoc networks*, in "Proc. 21st International Teletraffic Congress," (2009), 1–8.
[16] V. Kavitha and E. Altman, *Analysis and design of message ferry routes in sensor networks using polling models*, in "Proc. 8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)," (2010), 247–255.
[17] V. Kavitha and E. Altman, *Opportunistic scheduling of a message ferry in sensor networks*, in "Proc. 2nd International Workshop on Mobile Opportunistic Networking," (2010), 163–166.
[18] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys (eds.), "Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization," John Wiley & Sons, 1985.
[19] H. Levy and M. Sidi, *Polling systems: Applications, modeling, and optimization*, IEEE Transactions on communications, **38** (1990), 1750–1760.

[20] H. Miura, D. Nishi, N. Matsuda, and H. Taki, *Message ferry route design based on clustering for sparse ad hoc networks*, in "Proc. 14th international conference on Knowledge-based and intelligent information and engineering systems," (2010), 637–644.
[21] S. Sano, N. Miyoshi, and R. Kataoka, *m-balanced words: A generalization of balanced words*, Theoretical Computer Science, **314** (2004), 97–120.
[22] H. Takagi, *Queuing analysis of polling models*, ACM Computing Surveys, **20** (1988), 5–28.
[23] H. Takagi, *Queueing analysis of polling models: An update*, in "Stochastic Analysis of Computer and Communication Systems", Elsevier, 1990.
[24] W. Zhao and M. Ammar, *Message ferrying: proactive routing in highly-partitioned wireless ad hoc networks*, in "Proc. IEEE Future Trends of Distributed Computing Systems," (2003), 308–314.
[25] W. Zhao, M. Ammar, and E. Zegura, *A message ferrying approach for data delivery in sparse mobile ad hoc networks*, in "Proc. ACM MobiHoc," 2004, 187–198.
[26] W. Zhao, M. Ammar, and E. Zegura, *Controlling the mobility of multiple data transport ferries in a delay-tolerant network*, in "Proc. IEEE INFOCOM," (2005), 1407–1418.

Received xxxx 20xx; revised xxxx 20xx.

E-mail address: `habib@post.comm.eng.osaka-u.ac.jp`
E-mail address: `sasabe@comm.eng.osaka-u.ac.jp`
E-mail address: `takine@comm.eng.osaka-u.ac.jp`