Proposal and Evaluation of a Rendezvous-based Adaptive Communication Protocol for Large-scale Wireless Sensor Networks

Mirai Wakabayashi[1] and Harumasa Tada[2] and Naoki Wakamiya[1] and Masayuki Murata[1] and Makoto Imase[1]

[1] Graduate School of Information Science and Technology, Osaka University, Osaka 565-0871 Japan
m-wakaba,wakamiya,murata,imase@ist.osaka-u.ac.jp
[2] Faculty of Education, Kyoto University of Education, Kyoto 612-8522 Japan
htada@kyokyo-u.ac.jp

Abstract. Recently, wireless sensor networks (WSNs) have attracted attentions of many researchers since they can be used for wide range of applications such as environmental monitoring, security, disaster prevention, environmental control in office buildings, and precision agriculture. Control mechanisms for WSNs should adapt to a variety of communication patterns which reflect application requirements and the situation. In this paper, we propose ARCP (Ant-based rendezvous communication protocol), a novel communication protocol for WSNs. ARCP is designed to be adaptive to a variety of communication patterns by taking the rendezvous-based approach, where sensor data are collected and delivered through nodes marked as rendezvous points. At the same time, ARCP acquires robustness to failures and scalability with respect to network size by adopting AntHocNet, which is an ad-hoc routing protocol inspired by foraging behavior of ants. Through simulation experiments, we show that ARCP outperforms existing communication protocols in adaptability, robustness, and scalability.

Keywords: Wireless Sensor Network, Data-centric Communication, Rendezvous-based Approach, Ant Routing

1 Introduction

Recently, wireless sensor networks (WSNs) have attracted attentions of many researchers [2, 1]. WSNs consist of a number of small sensing devices (*nodes*) with wireless communicating component and a base station as a sink of sensor data. WSNs can be used for a wide range of applications including environmental monitoring, environmental control in office buildings, precision agriculture, and so on [3].

A variety of communication patterns emerges in WSNs reflecting application requirements and the situation. For example, consider a WSN for habitat monitoring. A number of nodes are distributed over the monitored area. They collect environmental data, such as temperature, humidity, and wind direction. Then, they send the collected sensor data to a base station periodically. Once an animal is detected, some of nodes

begin to collect and send more detailed sensor data more frequently to track the behavior of the animal. Following the movement of the animal, nodes for detailed sensing change. In this way, the location and number of nodes involved in communication and their frequency dynamically change in accordance with the situation. Therefore, control mechanisms for WSNs must be adaptive to a variety of communication patterns and changes of the situation. Failures of nodes and links also occur for fragility of low-cost device and unstable and unreliable radio communication environment. For example, a node halts due to energy exhaustion or physical damages. Some obstacles cause radio interference, which prevents a node from exchanging messages with a physically neighboring node. Therefore, control mechanisms for WSNs must be robust to failures of nodes and links.

To accomplish robust, scalable, and energy-efficient communication in WSNs, several protocols has been proposed [15, 17, 16]. Directed diffusion is a data-centric communication paradigm [15, 10, 12], where messages are sent with the description of interested data, e.g. "send wind direction and speed when the temperature is higher than 25°C" or "notify if a fire has been detected", rather than the address of the destination node, e.g. "send sensor data to the node with address *d*". Directed diffusion has three variations of communication protocols: two-phase pull (TPP), one-phase pull (OPP), and push. In TPP or OPP, nodes which are intended to gather sensor data (called *data gathering nodes*) send the description of interested sensor data to the entire network using flooding and then nodes that can provide requested sensor data (called *data provision nodes*) respond. In TPP, data gathering nodes receive responses via multiple routes and choose the minimum delay route among them. In OPP, on the other hand, data gathering nodes receive all responses via the minimum delay route. In push, data provision nodes send samples of sensor data to the entire network using flooding and then data gathering nodes respond. Because of their mechanisms, TPP and OPP are appropriate when data provision nodes are more than data gathering nodes and push is appropriate when data gathering nodes are more than data provision nodes [9]. Therefore, in directed diffusion, an appropriate protocol must be selected a priori taking into account expected communication patterns or must be selected dynamically reflecting the situation by introducing some switching mechanism. In [17], a tree-based multicasting scheme for communication between a data provision node and multiple mobile data gathering nodes is proposed. In [16], a Steiner tree is constructed between data provision nodes and a data gathering node by using an ant colony algorithm for data-centric routing. Although they also enable robust, scalable, and energy-efficient communication, they can be applied only to one-to-many or many-to-one type of communication.

The rendezvous-based approach [13], which is a hybrid of pull and push, delivers sensor data indirectly via nodes marked as *rendezvous points (RPs)*. Both data provision nodes and data gathering nodes notify RPs of the description of sensor data they can provide or they are interested in (See Fig. 1). Data provision nodes deliver sensor data to RPs. Data gathering nodes retrieve sensor data from RPs. In the rendezvous-based approach, numbers, locations, and communication frequency of data provision nodes and data gathering nodes do not much affect the performance of communication. Locations of RPs, on the other hand, do affect the performance and thus RPs must be located appropriately according to the communication pattern and the situation. For example,

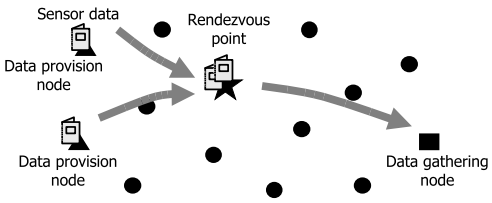


Fig. 1. Rendezvous-based approach

when there are more data provision nodes than data gathering nodes, an RP should be located closer to data provision nodes in order not to introduce much overhead in transmitting sensor data from data provision nodes to the RP. However, a mechanism to locate RPs has not been studied yet.

In this paper, we propose ARCP (Ant-based rendezvous communication protocol), a novel communication protocol for WSNs. Taking the rendezvous-based approach, ARCP is designed to be adaptive to a variety of communication patterns. ARCP appoints a node where delivery and retrieval of sensor data are expected to be frequent as an RP, promotes the usage of good RPs, and removes unused RPs. At the same time, ARCP acquires robustness to failures and scalability with respect to network size, by adopting AntHocNet [7], an ad-hoc routing protocol inspired by foraging behavior of ants. It is known that ants establish the shortest paths between the nest and food sources without centralized control by depositing volatile chemical substance called pheromone [8, 6, 5, 4].

This paper is organized as follows: Section 2 describes details of ARCP. Section 3 evaluates the adaptability, robustness, and scalability of ARCP through simulation experiments. Finally, Sect. 4 summarizes this paper and describes future works.

2 A Rendezvous-based Adaptive Communication Protocol for Large-scale Wireless Sensor Networks

In ARCP, nodes maintain neighborhood relation by periodically exchanging *hello messages*. Nodes send control messages called *routing ants* when they are ready to send or intend to receive sensor data. When routing ants from data provision nodes and those from data gathering nodes frequently encounter at a node, it is marked as an RP. Sensor data are transmitted by another kind of control messages called *carrier ants*. Carrier ants play similar role to data packets in AntHocNet. A *data provision carrier ant* generated by a data provision node carries sensor data from the data provision node to an RP. A *data gathering carrier ant* generated by a data gathering node carries sensor data from an RP to the data gathering node. Sensor data expires when a certain duration specified by the application passed from its generation. RPs discard expired sensor data. To improve delivery ratio of sensor data, carrier ants are guided to RPs with many arrivals of carrier ants, while unmarking RPs with few arrivals. All ants have the same TTL. When an ant travels more than $n_{TTL} > 0$ hops from its generation, it is discarded at the node.

Details of the behavior of ants in establishment and management of RPs and routes to them will be described in the following sections.

2.1 Behavior of routing ants

In ARCP, routes from data gathering nodes and data provision nodes to RPs are established and maintained by AntHocNet. In AntHocNet, routing information called *route pheromone* are maintained using three kinds of routing ants: reactive forward ants, proactive forward ants, and backward ants. Since ARCP is a data-centric communication protocol, a destination node is specified by the description of sensor data, rather than the address of the node.

Upon an application's request for delivery or retrieval of sensor data, a data provision node or a data gathering node s checks whether it has the pheromone for the description d. If the pheromone exists, the node s generates carrier ant and sends it at the regular intervals specified by the application. The carrier ant travels toward an RP choosing the next hop node according to pheromone values on each intermediate node. If no pheromone exists, the node s generates a routing ant. If the node s is a data provision node, the routing ant is a *data provision routing ant*. Otherwise, the routing ant is a *data gathering routing ant*. A routing ant generated at the node s behaves in the same way as a reactive forward ant in AntHocNet, which is sent to the next hop node chosen according to the pheromone for the description d or broadcast when there is no such pheromone. A node where data provision routing ants and data gathering routing ants frequently visit becomes an RP. When a routing ant marks a node as an RP or arrives at an RP, the routing ant becomes a backward ant in AntHocNet to return to the node s by traversing the same route it took to the RP. On the way to the node s, the routing ant updates pheromone values at the nodes along the route. Once the routing ant arrives at the node s, i.e. a route to the RP has been established, the node s begins to generate and send carrier ants.

In addition, as in AntHocNet, a data provision node and a data gathering node generate and send a data provision routing ant and a data gathering routing ant per n_r carrier ants, respectively, in order to maintain and improve routes. These routing ants behave as proactive forward ants of AntHocNet and they are sent to the next hop node chosen according to the pheromone.

2.2 Marking and Unmarking of RPs

Appropriate location of RPs depends on the number and position of data provision nodes and data gathering nodes. As mentioned in Sect. 2.1, nodes where data provision routing ants and data gathering routing ants frequently encounter are marked as RPs. It implies that a node is marked as an RP if it has many data provision nodes and data gathering nodes around it and has relayed many sensor data. By locating RPs closer to many data provision nodes and data gathering nodes, overhead in transmitting sensor data from data provision nodes to RPs and from RPs to data gathering nodes can be reduced. In this context, the term *encounter* means that a data gathering routing ant and a data provision routing ant pass the same node within a certain period of time.

Upon the arrival of a data provision routing ant at the node i, the timer X_p^i is set to the initial value $X_p > 0$. Likewise, upon the arrival of a data gathering routing ant at the node i, the timer X_g^i is set to the initial value $X_g > 0$. When the other timer is greater than zero on setting a timer, i.e. a certain period has not passed from the arrival of a routing ant from a node of the other type, the *encounter counter* C^i is increased by one. If the other timer is zero, the encounter counter C^i is decreased by one. If the encounter counter C^i of the node i reaches the value $n_{RP} > 0$, then the node i is marked as an RP. Since this marking is performed autonomously at each node, there is the possibility that multiple nodes are marked as an RP. Multiple RPs enable load distribution, where carrier ants are distributed among RPs. Consequently, the energy consumption at an RP and its neighbor will be suppressed and network congestion will be avoided. Robustness against RP failures will also be improved. However, because it is necessary to visit multiple RPs, not necessarily all though, to collect all sensor data, the gathering or delivery ratio of sensor data could decrease. The number of RPs can be reduced by, for example, using larger threshold n_{RP}. With larger n_{RP}, however, it takes long time for the first RP to appear. If the encounter counter C^i of the node i decreased to less than n_{RP}, i.e. few ants arrive at the node i, then the node i is unmarked. The unmarked node broadcasts a *link failure notification message* of AntHocNet in order to remove routes to itself.

2.3 Goodness of RPs

RPs are not equal in the frequency of arrival of carrier ants and the number of stored sensor data. The RP r is considered good when both data provision carrier ants and data gathering carrier ants arrive at r frequently and many sensor data are relayed at r. Guiding many ants to good RPs improves the delivery ratio of sensor data. For this purpose, in addition to the route pheromone, we introduce another kind of pheromone called *rendezvous pheromone* which reflects the goodness of the RP.

Since the route pheromone value T_{nd}^i in AntHocNet reflects only the latency of the route, ants prefer shorter routes rather than longer ones. In the rendezvous-based approach, however, the distance to an RP does not necessarily indicate the goodness of the RP. For example, consider the following case (See Fig. 2). There are two RPs r_1 and r_2 between data provision nodes and a data gathering node. The RP r_1 is near the data provision nodes, but it is far from the data gathering node. On the other hand, the RP r_2 is near the data gathering node, but it is far from the data provision nodes. In this case, while data provision carrier ants frequently visit the RP r_1, no data gathering carrier ant comes to the RP r_1, for choosing closer RP r_2. As a result, no sensor data can be delivered to the data gathering node via the RPs.

Now, we introduce rendezvous pheromone to tackle this problem. Ants choose the next hop node according to both the rendezvous pheromone value U_{nd}^i and the route pheromone value T_{nd}^i. On the way to the originating node, an ant at the node i updates the rendezvous pheromone value U_{nd}^i as well as the route pheromone value T_{nd}^i by the following equations:

$$T_{nd}^i = \gamma_T \, T_{nd}^i + (1 - \gamma_T) \tau_d^i \tag{1}$$

$$U_{nd}^i = \gamma_U \, U_{nd}^i + (1 - \gamma_U) \, n_{data} \tag{2}$$

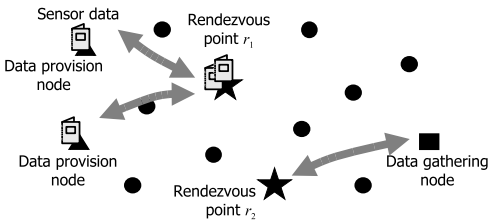


Fig. 2. RPs unshared by data provision node and data gathering node

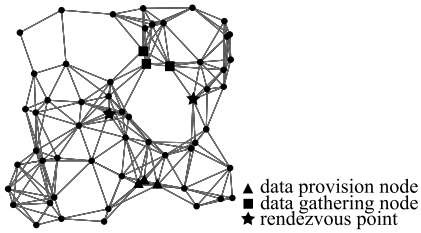


Fig. 3. An example of network topology

where n is the node from which the node i received the ant, $\gamma_T \in [0,1]$ and $\gamma_U \in [0,1]$ are smoothing parameters, τ_d^i is the estimated time for a carrier ant to travel towards the RP (See [7] for details), and n_{data} is the number of sensor data stored at the RP when the ant left. The rendezvous pheromone value U_{nd}^i to an RP which has many sensor data becomes large by (2). An ant at the node i chooses the next hop node n for the description d with the probability P_{nd} given by the following equation:

$$P_{nd} = \frac{(T_{nd}^i)^{\beta_T}(U_{nd}^i)^{\beta_U}}{\sum_{j \in N_d^i}(T_{jd}^i)^{\beta_T}(U_{jd}^i)^{\beta_U}} \tag{3}$$

where N_d^i is the set of neighbors of the node i and $\beta_T, \beta_U > 0$ are parameters representing the weights of route pheromone and rendezvous pheromone respectively.

2.4 Transmission of Sensor Data

Data provision nodes deliver sensor data to RPs using data provision carrier ants. Data gathering nodes retrieve sensor data from RPs using data gathering carrier ants. Sensor data are associated with lifetime specified by an application. A data provision node holds its sensor data during their lifetime. An RP holds received sensor data during their lifetime, but it discards duplicated sensor data immediately. In addition, an RP aggregates sensor data [14], if an application requires. A data provision carrier ant carries all the sensor data stored at the data provision node to an RP. A data gathering carrier ant carries all the sensor data stored at an RP to the data gathering node. The sending rate of carrier ants at a data provision node and a data gathering node is determined according to an application requirement. A carrier ant travels toward an RP by choosing the

Table 1. Control parameter setting in simulation

β_T, β_U	Weight of pheromone	1 (routing ants) or 3 (carrier ants)
γ_T, γ_U	Smoothing parameter for pheromone	0.7
n_r	Number of carrier ant per data provision routing ant	30
n_{RP}	Threshold for encounter counter	10
t_{hello}	Hello message interval	30 s
X_p, X_g	Initial timer value for routing ants	240 s

next hop node with the probability given by (3) at each node and then returns to the originating node by traversing the same route it took to the RP. Note that carrier ants tend to choose better route than routing ants, because parameters β_T and β_U in (3) are set larger for carrier ants.

3 Simulation and Evaluation

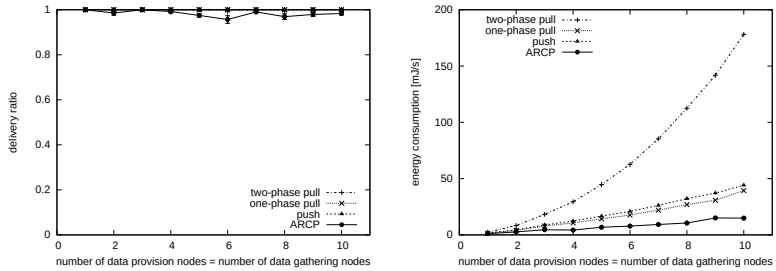
In this section, we evaluate adaptability, robustness, and scalability of ARCP through simulation experiments.

3.1 Simulation Environment

We consider a WSN consisting of randomly placed immobile sensor nodes. A bidirectional link is established between any two nodes whose distance is less than 12 m. The link propagation delay is assumed to be 3 ms. It is assumed that no message is lost in the MAC layer due to e.g. collisions. Data provision nodes and data gathering nodes are chosen to form clusters [13]. First, one node is randomly chosen as a data provision node. Then, $n-1$ nodes closest to the node are appointed as a data provision node. n data gathering nodes are chosen in the same way from the remaining nodes. Data provision nodes and data gathering nodes do not change during a simulation run. An example of network topology is shown in Fig. 3.

A data provision node generates sensor data every 2 s. For each of sensor data, the data provision node generates a data provision carrier ant. A data gathering node generates a data gathering carrier ant every 2 s. The size of single sensor data is set at 10 bytes. The size of an ant is set at 63 bytes except that a carrier ant amounts to $63+10k$ bytes where k is the number of sensor data it carries. The size of a hello message and a link failure notification message are set at 3 and 7 bytes respectively. The lifetime of sensor data is set at 40 s. Sensor data are not aggregated at RPs. Table 1 summarizes parameter setting in ARCP. Parameter setting for directed diffusion (TPP, OPP, and push) is based on [9].

As performance metrics, we use the delivery ratio of sensor data and the energy consumption. The delivery ratio for the whole network is defined as the average of the delivery ratio for all data gathering nodes. The delivery ratio for a data gathering node is defined as the ratio of the number of sensor data received by data gathering node to the number of sensor data generated by all data provision nodes. The energy consumption of the whole network is defined as the sum of the amount of energy consumed by



(a) Delivery ratio (b) Energy consumption

Fig. 4. Adaptability to the number of data provision/gathering nodes

all nodes in sending and receiving messages. The energy consumption in sending and receiving a message is calculated according to [11]. Each simulation runs for 10,000 s in simulation time. We show the average and 95% confidence interval of results of five runs.

3.2 Evaluation of Adaptability

In order to evaluate the adaptability to communication patterns, we performed simulation experiments while changing the number of data provision nodes and data gathering nodes. 60 nodes are randomly placed in the $50 \times 50\,\mathrm{m}^2$ monitored area and the TTL of ants, n_{TTL} is set at 10.

Results are shown in Fig. 4. The number of data provision nodes which is equal to the number of data gathering nodes is changed from 1 to 10.

As shown in Fig. 4(a), the delivery ratio is always 1 in directed diffusion and is almost 1 in ARCP. That is, sensor data are successfully delivered to data gathering nodes regardless of the numbers of data provision nodes and data gathering nodes in any protocol. The reason for the slightly low delivery ratio of ARCP is that an ant chooses a next hop node in a probabilistic way for robustness against node failure. As such, an ant would take a long way and spend its TTL before reaching an RP. The delivery ratio can be improved by using larger n_{TTL}. However, it leads to larger energy consumption for allowing a longer route.

As shown in Fig. 4(b), ARCP is the most energy efficient protocol among the four under the many-to-many communication scenario. In ARCP, the increase of the number of data provision nodes and data gathering nodes only results in the increase of traffic between RPs and data provision nodes and between RPs and data gathering nodes. On the other hand, in directed diffusion, the increase of the number of data provision nodes and data gathering nodes results in the increase of the number of message flooding, which consumes the considerable amount of energy for involving all sensor nodes. Note that TPP costs the largest amount of energy, because data provision nodes first flood messages and then data gathering nodes send responses using all possible routes. In conclusion, ARCP is adaptive and scalable to the number of data provision nodes and data gathering nodes.

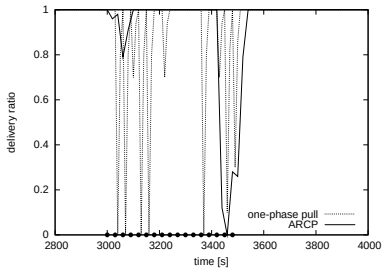


Fig. 5. Temporal variation of delivery ratio in a network with node failures

3.3 Evaluation of Robustness

In simulation experiments, 500 nodes are randomly arranged in the $140 \times 140\,\mathrm{m}^2$ monitored area and n_{TTL} is set at 20. The numbers of data provision nodes and data gathering nodes are both set at 10. In order to evaluate robustness against node failures, we randomly choose 50 nodes among nodes which are not either of data provision nodes or data gathering nodes and stop them at 3,000 s. Failed nodes cannot send or receive any messages. All messages sent to failed nodes are lost. Any routing information and sensor data in failed nodes are removed. After 30 s, the failed nodes go back to normal operation and next 50 nodes are randomly chosen to halt. The same procedure is repeated until 3,500 s. For the comparison, we also consider OPP for its low energy consumption shown in Fig. 4(b).

Although not shown in figure, the average delivery ratio during the 500 s period is 77.5% in OPP and 80.5% in ARCP, respectively. In general, the rendezvous-based approach is vulnerable to failure of RP. If an RP fails and no other RP remains, the delivery ratio considerably degrades until a new RP is established. On the contrary to this conjecture, the delivery ratio of ARCP is higher than that of OPP.

To understand this result, we show an instance of temporal variation of average delivery ratio in Fig. 5. Circles on the horizontal axis in the figure indicate instances when node failures occurred. As can be seen, in OPP, the delivery ratio often drops to zero when node failure occurs. Although the delivery ratio is recovered soon by frequent message flooding, a large amount of sensor data is lost in this period. On the contrary, node failure does not affect the delivery ratio of ARCP in most cases, since ant-based routing allows ants to detour failed nodes by the probabilistic next hop selection. However, the delivery ratio of ARCP decreases to zero at about 3,430 s. At this time, an RP halted and there was no other RP. Due to autonomous and self-organizing behavior in establishing RPs and routes, it takes time to recover the delivery ratio once the only RP fails. However, the typical number of RPs was 2 in the simulation experiments and thus the probability that all RPs fail is low in the random failure scenario. Therefore, it can be concluded that ARCP is similarly to or slightly more robust than flooding-based deterministic protocol, i.e. OPP.

3.4 Evaluation of Scalability

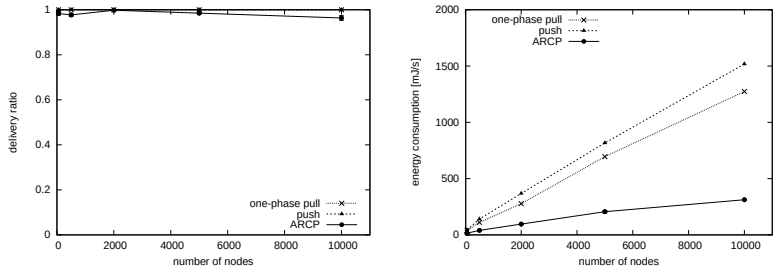
In order to evaluate scalability with respect to the number of nodes, we conducted simulation experiments for 60, 500, 5,000, and 10,000 nodes. In the case of 60 nodes, they are distributed in the $50 \times 50\,\mathrm{m}^2$ monitored area. We keep the density of nodes the same among the node population by changing the area. n_{TTL} is set in accordance with the diameter of a network as 10, 20, 70, and 100, respectively. The numbers of data provision nodes and data gathering nodes are both set as 10.

As shown in Fig. 6(a), the delivery ratio is kept 1 in OPP and push, since both use flooding to establish routes among data provision nodes and data gathering nodes. The delivery ratio of ARCP is also almost 1 for all of the node population, but we see the slight tendency of decrease as the number of nodes increases. As the population becomes large, ARCP takes more time to establish RPs and routes. Therefore, the amount of sensor data which expire at a data provision node before RP establishment increases. In addition, the length of routes also increases. Consequently, the probability that a carrier ant spends up its TTL becomes large, for taking a longer route by choosing a next hop node with small amount of pheromone. However, the delivery ratio is kept as high as 96%.

As shown in Fig. 6(b), the energy consumption increases in order of $O(N)$ in OPP and push and $O(\sqrt{N})$ in ARCP where N is the number of nodes, respectively. That is, ARCP is more scalable than others. Messages in OPP and push are classified into two categories: flooding and non-flooding. Since the number of flooding messages increases in proportional to the number of nodes, energy consumption increases in order of $O(N)$ for flooding messages. A non-flooding message is transmitted via the shortest route between a data provision node and a data gathering node. Since nodes are randomly located in the square monitored area, the length of the shortest route between two arbitrary nodes is in order of $O(\sqrt{N})$. Then, the amount of energy consumed by non-flooding messages is in order of $O(\sqrt{N})$. Since flooding is periodically performed, the energy consumption of OPP and push as a whole becomes in order of $O(N)$. On the other hand, in ARCP, it is difficult to estimate the energy consumption accurately, since ants sometimes broadcast themselves or choose a longer route. However, now, we approximate the energy consumption as follows. Most of ants are expected to choose the shortest route and their energy consumption is in order of $O(\sqrt{N})$. Some ants behave like a flooding message and their energy consumption is in order of $O(N)$. However, flooding behavior basically occurs to find an RP and establish a route, i.e. only at the initial stage. Therefore, we can approximate the energy consumption of ARCP as $O(\sqrt{N})$.

4 Conclusion

In this paper, we propose ARCP, a novel data-centric communication protocol for WSNs. ARCP combines the rendezvous-based approach and the ant-based routing protocol to be adaptive to communication patterns, robust to failures of nodes and links, and scalable with respect to network size. Through simulation experiments, we show that ARCP is more adaptive and scalable than existing communication protocols while keeping as high robustness as existing communication protocols.



(a) Delivery ratio (b) Energy consumption

Fig. 6. Scalability with respect to the number of nodes

We are considering further improvement on the adaptability of ARCP to dynamic changes in communication patterns. For this purpose, we need to develop a mechanism to dynamically move RPs to locations more appropriate for new condition.

Acknowledgments. The authors would like to thank Associate Professor Hiroyuki Ohsaki of Osaka University for his fruitful suggestions. This work was partially supported by "The Global Center of Excellence Program" and a Grant-in-Aid for Scientific Research (A) (2) 16200003 from The Japanese Ministry of Education, Culture, Sports, Science and Technology.

References

1. Akkaya, K., Younis, M.: A survey on routing protocols for wireless sensor networks. Ad Hoc Networks **3**(3), 325–349 (2005)
2. Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: A survey. Computer Networks **38**(4), 393–422 (2002)
3. Arampatzis, T., Lygeros, J., Manesis, S.: A survey of applications of wireless sensors and wireless sensor networks. In: Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation (ISIC05/MED05), pp. 719–724 (2005)
4. Beckers, R., Deneubourg, J.L., Goss, S.: Modulation of trail laying in the ant *Lasius niger* (Hymenoptera: Formicidae) and its role in the collective selection of a food source. Journal of Insect Behavior **6**(6), 751–759 (1993)
5. Beckers, R., Deneubourg, J.L., Goss, S., Pasteels, J.M.: Collective decision making through food recruitment. Insectes Sociaux **37**(3), 258–267 (1990)
6. Deneubourg, J.L., Aron, S., Goss, S., Pasteels, J.M.: The Self-Organizing Exploratory Pattern of the Argentine Ant. Journal of Insect Behavior **3**(2), 159–168 (1990)
7. Di Caro, G., Ducatelle, F., Gambardella, L.M.: AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks. European Transactions on Telecommunications, Special Issue on Self-organization in Mobile Networking **16**(5), 443–455 (2005)
8. Goss, S., Aron, S., Deneubourg, J.L., Pasteels, J.M.: Self-organized shortcuts in the Argentine ant. Naturwissenschaften **76**, 579–581 (1989)
9. Heidemann, J., Silva, F., Estrin, D.: Matching data dissemination algorithms to application requirements. In: Proceedings of the 1st ACM conference on Embedded networked sensor systems (SenSys 2003), pp. 218–229 (2003)

10. Heidemann, J.S., Silva, F., Intanagonwiwat, C., Govindan, R., Estrin, D., Ganesan, D.: Building efficient wireless sensor networks with low-level naming. In: Proceedings of Symposium on Operating Systems Principles, pp. 146–159 (2001)
11. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings of the Hawaii International Conference on System Sciences (HICSS) (2000)
12. Krishnamachari, B., Estrin, D., Wicker, S.: Modelling data-centric routing in wireless sensor networks. Tech. Rep. CENG 02-14, USC Computer Engineering (2002)
13. Krishnamachari, B., Heidemann, J.: Application-specific modelling of information routing in wireless sensor networks. Tech. Rep. ISI-TR-576, USC-ISI (2003)
14. Rajagopalan, R., Varshney, P.K.: Data aggregation techniques in sensor networks: A survey. IEEE Communications Surveys and Tutorials **8**(4), 48–63 (2006)
15. Silva, F., Heidemann, J., Govindan, R., Estrin, D.: Directed diffusion. Tech. Rep. ISI-TR-2004-586 (2004)
16. Singh, G., Das, S., Gosavi, S.V., Pujar, S.: Ant colony algorithms for steiner trees: An application to routing in sensor networks. Recent Developments in Biologically Inspired Computing pp. 181–204 (2004)
17. Zhang, W., Cao, G., Porta, T.L.: Dynamic proxy tree-based data dissemination schemes for wireless sensor networks. Wireless Networks **13**(5), 583–595 (2007)