Journal of Computers Vol. 28, No. 4, 2017, pp. 103-113
doi:10.3966/199115592017082804011

An Ant Colony Optimization Routing Strategy for Wireless Sensor Networks: A SDN-like Perspective

Tu-Liang Lin[1], Yu-Sheng Chen[1], and Hong-Yi Chang[1*]

[1] Department of Management Information System, National Chiayi University
Chiayi 600, Taiwan, ROC
{tuliang, s1021416, hychang}@mail.ncyu.edu.tw

Received 30 August 2014; Revised 30 June 2015; Accepted 28 December 2016

Abstract. Software defined networking (SDN) provides the advantages of decoupling the control and the forwarding planes and the packets are classified into flows and the routing decision are based on the flow. Since the SDN conducts the flow-based routing strategy and ant colony optimization (ACO) routing protocols have the potential to work in a flow-based manner. Previous research of ant-based routing algorithms mainly focused on energy efficiency. However, how ant-based routing algorithms perform in other metrics has been neglected. In this research, we compare the packet delivery ratio, the end-to-end delay and the first packet arrival time of an ant colony optimization (ACO) routing algorithm with three routing protocols, namely, AODV, DSDV, and DSR, in the wireless sensor networks with different density traffic. We found that the ACO is not only energy efficient but also has the best performance in both the packet delivery ratio and the first packet arrival time compared to other routing algorithms with low density traffic. In some low density traffic cases, it might not perform well in the end-to-end delay. The ACO does not perform well in both the packet delivery ratio and the end-to-end delay with high density traffic, so ACO is more suitable to be used in low density traffic.

Keywords: ant colony optimization, mobile ad hoc network, routing algorithm, wireless sensor network

1 Introduction

In 2008, McKeown et al. introduced the famous software defined networking (SDN) implementation, OpenFlow [1]. Software defined networking (SDN) provides the advantages of decoupling the control and the forwarding planes and depends on the underlying connecting devices, such as switches or routers to classify incoming packets into flows and make the routing decision based on the flow. This means a set of rules can be defined by a remote centralized controller or a group of controllers. The centralized controllers can define the forwarding policies and update the policies of the underlying switches. Instead of routing the packets in a distributed manner, controllers can define rules for individual connecting devices. The SDN conducts the flow-based routing strategy and we found that ant colony optimization (ACO) routing protocols have the potential to work in a flow-based manner.

Since the basic concept of SDN is that the controls and data are separated, the overall performance can be optimized [2]. ACO routing has similar characteristics. The movement of an ant is controlled by the pheromone. Therefore, in ACO routing, the pheromone possesses the SDN-like characteristics and we renamed it as "programmable pheromone. In other words, through the programmable pheromone, we are able to plan routes to meet different software requirement or different QoS. Fig. 1 shows the contrast between SDN and programmable ACO routing algorithm.

* Corresponding Author

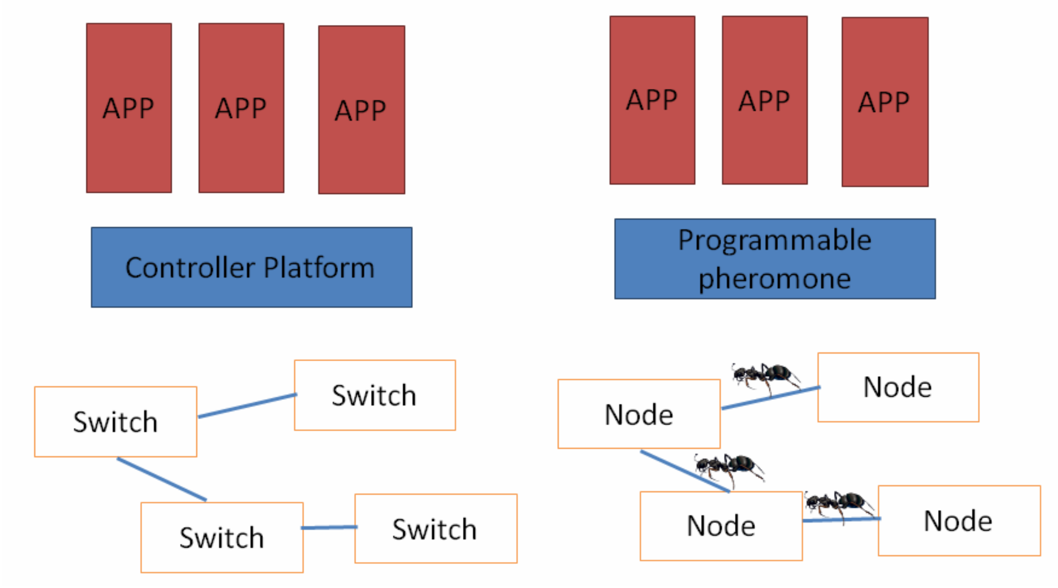

Fig. 1. Contrast between SDN and programmable ACO routing algorithm

In wireless sensor networks (WSNs), each sensor can monitor various characteristics of a region, e.g., humidity, gas, temperature, and so on. Wireless sensor networks also have been applied to many applications, such as pollution monitoring [3], automobile communication [4], biomedical health monitoring [5], natural disaster relief [6], and so on. After collecting the data, sensors will transfer the data to data collection center.

Each sensor in WSNs has a built-in battery which normally won't recharge or replace when the battery is drained. Especially in hazardous environmental exploration [7], sensing areas are difficult to reach by humans, so the sensors are placed by robots or sprayed from the air. In such a circumstance, if the built-in battery is exhausted, it is difficult to recharge, and thus eventually can cause the appearance of holes formed by the failure nodes, which could hinder the data transition in the wireless sensor network. Therefore, how to design an energy efficient protocol to limit the energy consumption is an import issue and ant colony optimization routing algorithms have been proved to be energy efficient.

Previous studies have shown that the ant-based routing algorithms are energy efficient compared to other protocols. However, there is a lack of study on other performance metrics such as the packet loss rate. Packet loss might lead to the disappearance of some sensing data, and thus cause the inaccuracy of the information. Therefore, in this research, we focus on the comparison of an ant colony optimization (ACO) algorithm with other routing algorithms commonly used in mobile ad hoc networks (MANET). Mobile nodes in MANET and sensor nodes in WSN share some common characteristics such as wireless communications in the absence of an infrastructure. Therefore, both MANET and WSN belong to the class of infrastructure-less networks and the routing protocols designed for MANETs and the routing protocols designed for WSNs sometimes can be easily applied to each other. In this research, we attempt to study the applicability of an ACO routing algorithm to the wireless sensor networks and compare the performance of different routing protocols on different network environments and traffic settings.

We attempt to investigate whether the ant colony optimization routing algorithm sacrifice other performances for energy saving purpose. Three common network performance metrics, the packet delivery ratio, the end-to-end delay and the first packet arrival time, are explored in this research.

With the arrival of the Internet of things (IOT), the number of things interconnected is huge. How to make these things to communicate efficiently and also to meet different application requirements is important. In SDN, open APIs (application programming interfaces) are used to support the services or applications running over the network. Therefore, a programmable ACO routing algorithm which can make ants work in a fashion that can meet the requirements of the upper layer services or applications not only can improve the performance of WSNs but also has the potential to be applied to the IOT.

2 Related Work

The wireless communication links between two nodes establish dynamically when two nodes are within the communication range and break when two nodes are out of the communication range. Since the communication range of each node is limited, the communications between two nodes out of the communication range are through multi-hops among the intermediate relay nodes. Discovering the routes between two nodes are challenge tasks and several routing algorithms were proposed. A routing algorithm is proactive if the routing table is maintained and the routing paths for all nodes are calculated even when they don't need the paths. The routing paths are maintained according to the up-to-date topology. On the other hand, reactive algorithms or on-demand algorithms create routes when only when the routes are needed for some packets to be delivered. The reactive or on-demand routing algorithms don't need to maintain the up-to-date routing information, so it is possible that the packets are not transmitted on the optimal routes.

2.1 ACO routing

ACO is an algorithm that mimics the real ant colony behavior. When searching for foods, ants will leave pheromones on the path and other ants will choose the path according to the pheromone concentrations. Pheromones evaporate over time, so pheromones are rapidly accumulated in the shorter paths. After some time, a shortest path is found. Fig. 2 shows the real ant colony behavior. When there is an obstacle on the path, the ants spread into two directions at an equal chance in the beginning. After some time, due to the accumulation of pheromones, ants tend to choose the shorter path.

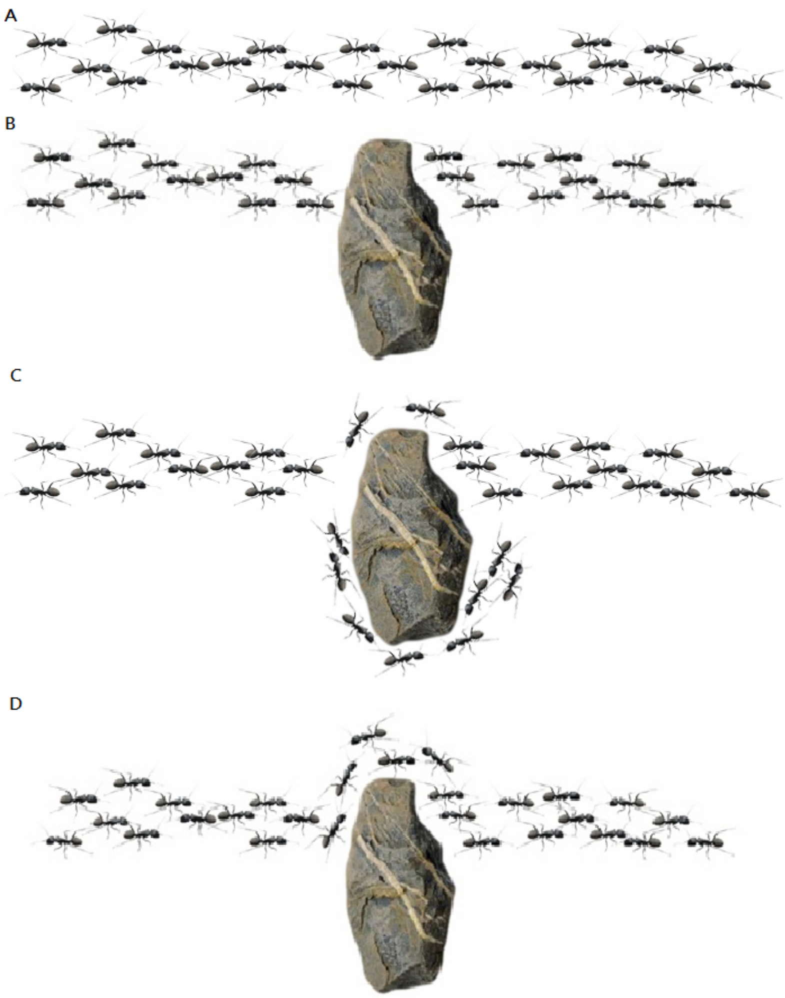

Fig. 2. The real ant colony behavior

The ACO algorithm was first applied to the travel salesman problem (TSP) [8-9]. A certain number of artificial ants are randomly put on the nodes in the TSP. Each ant chooses the next visiting node according to the intensity of the pheromone. The formula of the transition probability is as follows:

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{\left[\tau_{ij}\right]^{\alpha}\left[E_{j}\right]^{\beta}}{\sum\limits_{u\in J_s}\left[\tau_{iu}\right]^{\alpha}\left[E_{u}\right]^{\beta}} & if \quad j\in J_s(i)\\ 0 & others\end{cases} \tag{1}$$

where τ_{ij} is the intensity of the pheromone of the node i and node j at time t. $J_s(i)$ is the node set which ant s can sense on node i. E is the visibility function calculated from $1/(C-e_j)$ where C is the initial energy level of the nodes and e_j is the actual energy level of node j. The α and β are parameters to control the relative influence of the pheromone and the distance. After an ant completes its path, the pheromone level of the path will be updated. Local updating is as follows:

$$\tau_{ij}=(1-\rho)\tau_{ij}+\left[\Delta\tau_{k}\big/\varphi\cdot Bd_{k}\right] \tag{2}$$

$$\Delta\tau_{k}=\frac{1}{C-\left[EMin_{k}-Fd_{k}\big/EAvg_{k}-Fd_{k}\right]} \tag{3}$$

The pheromones evaporate over time and $\rho\in(1,0)$ is the pheromone decay parameter and φ is a coefficient. Bd_k is the number of visited nodes by the backward ant k until node i. fd_k is the number of nodes stored in its memory by the forward ant k.

Fig. 3 shows the flow chart of the ant colony optimization. When the number of repetitions reaches the upper limit or the path is found, terminates the optimization.

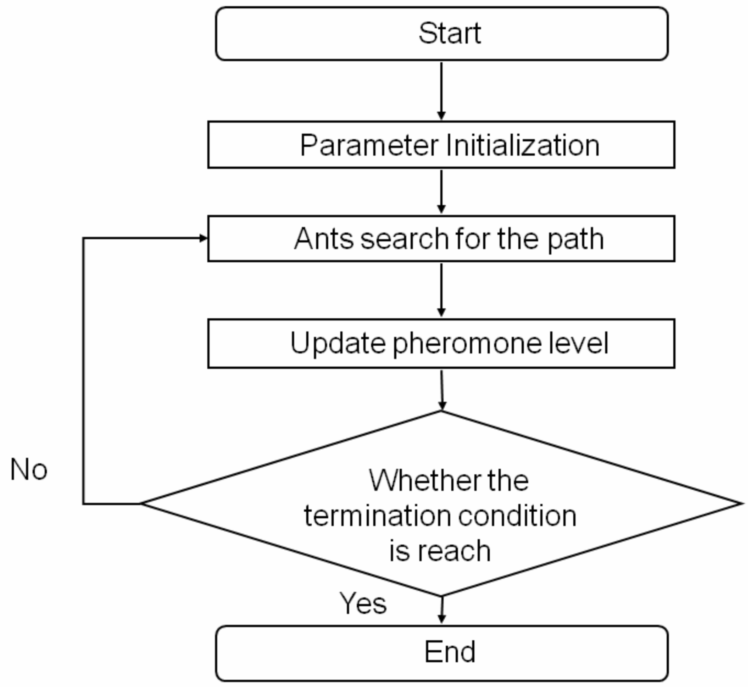

Fig. 3. The flow chart of the ACO algorithm

2.2 Ad-hoc On-demand Distance Vector (AODV) Routing

A particularly popular algorithm for wireless sensor networks is ad-hoc on-demand distance vector (AODV) algorithm [10]. The sensor nodes broadcast rout request (RREQ) messages to the nearby sensor nodes. After receiving a RREQ message, the sensor node will check whether the sensor itself is the target destination. If the sensor node is the destination, then the sensor node replies routing response (RREP) to the source sensor node. If the sensor node is not the destination, then the sensor node searches its routing table, and if the destination can be found in the routing table, then the sensor node sends routing response (RREP) message to the source sensor node, otherwise the sensor node continue to broadcast the RREQ.

The route will be removed from the routing table if not used for a period of time. Therefore, a new route discovery is required when the sensors need to send the data again and thus cause additional energy consumptions.

2.3 Destination-Sequenced Distance-Vector (DSDV) Routing

Destination-Sequenced Distance-Vector (DSDV) routing protocol is a table-driven routing algorithm for mobile ad hoc networks based on the Bellman–Ford algorithm [11]. In the DSDV protocol, each node has to build and maintain a routing table. The routing table contains the destination field, next hop field, hop count field, sequence number field, and install time field. In order to solve the routing loop problem, the destination sequence numbers are used to determine whether the paths are outdated and need updates.

When the network topology is changed, the nodes periodically broadcast rout updates to the neighboring nodes. After receiving the route updates, nodes will update the routing tables accordingly. The table updates can be time-driven or event-driven. In order to ensure using the best path for packet transmissions, the nodes will retain previous path information. The use of destination sequence numbers in DSDV can prevent the occurrences of loops.

The DSDV protocol defines two different types of routing information broadcast packets, full dump and incremental. When large amounts of routing data need to be updated, the full dump update is used. On the other hand, when only small amounts of routing data need to be updated, the incremental update is used. In DSDV protocol, the damping fluctuation constant is set to avoid the constant changes of the routing table and thus can stabilize the network.

2.4 Dynamic Source Routing (DSR)

DSR is similar to AODV. The nodes will broadcast route requests (RREQ) to the nearby nodes. When receiving a RREQ, the node will check whether it is the destination. If the node is the destination, then it replies a routing response (RREP) to the source node, otherwise the node continue to broadcast the RREQ. After receiving all the RREPs, the source node chooses the best path from the RREPs.

Instead of maintaining the routing tables, the source node has all the routing information, so in DSR the routing information is kept in the packet header and the packets are transmitted according to the routing information in the packet header [12].

3 The Proposed Design of an ACO-based Controller

Theoretically, a single controller can control unlimited number of nodes. Suppose there are multiple controllers existing in a network. These controllers can form a NOS (Network operating system) [13]. NOX, a famous open source controller implementation [13], provide a framework to control the connecting devices through the OpenFlow API. Although SDN and OpenFlow were designed for infrastructure-based network, the adaptation to wireless sensor network which is not an infrastructure-based network is not straightforward. In wireless sensor network, one or several sink nodes exist, and normally the sink nodes will connect to a more powerful device to aggregate or to process the collecting data. The controller can be run on this powerful device, as Fig. 4. There are two major tasks the ACO-based controller will perform.

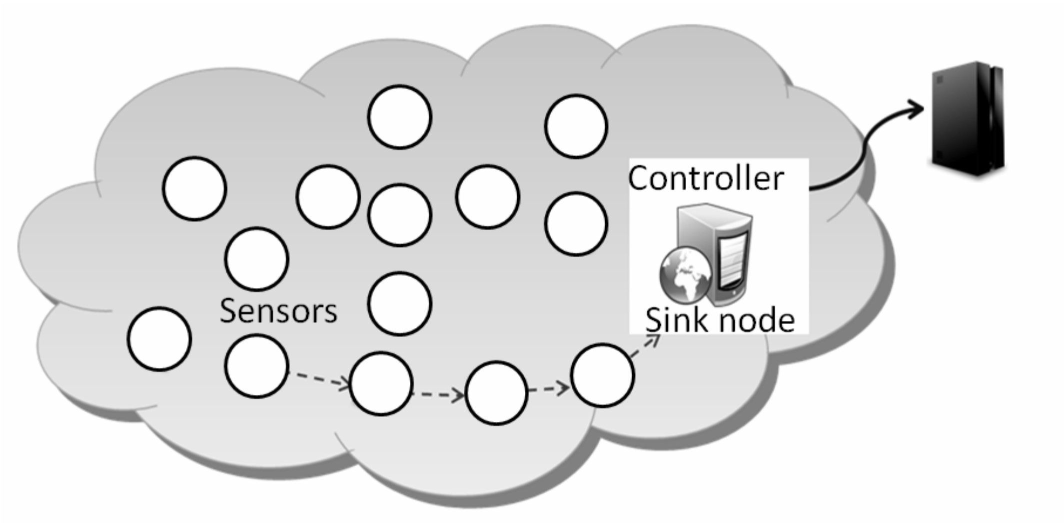

Fig. 4. Controllers run on the sink node

3.1 Collection of Pheromone Level

The pheromone level will be provided to the controller. When the sensor data reach the sink node, the ants that carry the sensor data also report the pheromone level of the passing by nodes to the controller. The controller will maintain the pheromone level of each node in the network. It's somewhat like collecting the topology information in the wired networks.

3.2 Specification of Rules for Updating Pheromone

Different from the wired network, the nodes in wireless sensor networks have less computational capacity. Therefore, only simple rules and actions can be performed. We will check whether the sensor data are critical, and if the sensor data are critical, the pheromone level will increase by large quantity. Otherwise, if the sensor data are not so critical, the pheromone level will increase by small quantity. We will also check battery of the sensor nodes, if the battery of the sensor node is low, the pheromone level should increase by small quantity. On the other hand, if the battery of the sensor node is high, the pheromone level should increase by large quantity. Table 1 shows the exemplary flow table, "Critical" means the senor data is critical, "Low" means the battery is low and $\triangle$P is the amount of pheromone that can be increased in every iteration.

Table 1. Exemplary flow

Node Addr	Op	Value	Action
2	=	Critical	Increase $\triangle$P
4	$\neq$	Critical	Decrease $\triangle$P
1	=	Low	Increase $\triangle$P
3	$\neq$	Low	Decrease $\triangle$P

4 Network Simulations

In this research, network simulator 2 (ns2) was used to conduct the simulations. The performances of four routing algorithms, namely, ACO, AODV, DSDV, and DSR, were investigated in the wireless sensor network. The wireless sensor nodes were deployed onto a 50 x 50 m^2 grid as shown in Fig. 5. Two CMU tools, namely, setdest and cbrgen, were used to create the moving and data flow scenarios. Table 2, Table 3 and Table 4 are the parameter settings of movement, data flow and network setting respectively.

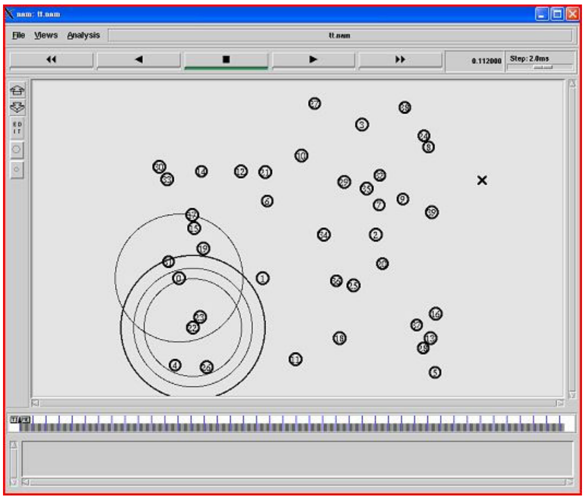

Fig. 5. The graphical representation of network topology

Table 2. The parameter set of the movement setting

Parameters	Values
Number of nodes	10/20/30/40/50/60/70/80/ 90/100
Node moving range	500 × 500 /m
Pause time	0.0 /sec
Max speed	10.0 m/sec
Simulation time	100 /sec

Table 3. The parameter set of the data flow setting

Parameters	Values
Type	Cbr
Random number generator	1
Max number of connections	10/100
Rate	10 /sec
Type	Cbr

Table 4. The parameter set of the network setting

Parameters	Values
Channel type	Channel/WirelessChannel
Propagation model	Propagation/TwoRayGround
Phy type	Phy/ WirelessPhy
Mac protocol type	Mac/802_11
Queue type	Queue/ DropTail/ PriQueue, and CMUPriQueue
Antenn type	Antenna/ OmniAntenna
Link layer type	LL
Routing	ACO, AODV,, DSDV, and DSR
Channel type	Channel/WirelessChannel

5 Performance Evaluations

In this research, the packet delivery ratio, the end-to-end delay and the first packet arrival time are compared in different traffic density. Two densities, high and low, are used in the network simulation. The low density traffic has the max number of connections equals to 10 and the low density traffic has the max number of connections equal to 100.

5.1 Low Density Traffic

Fig. 6 shows the packet delivery ratios of ACO, AODV, DSDV and DSR in different number of nodes in low density traffic. The ACO has the best performance among these four routing algorithms in the packet delivery ratio in almost all cases. No matter how many nodes are there, the ACO have highest packet delivery ratio, which is close to 100%. The packet delivery ratios of AODV, DSDR and DSR fluctuate when the number of nodes varies. From the perspective of packet delivery, the ACO is more stable than the other three routing algorithms in low density traffic.

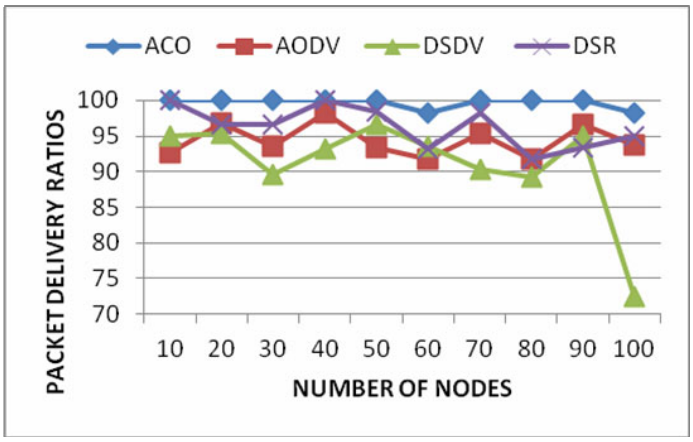

Fig. 6. The comparison of the packet delivery ratio in low density traffic

Fig. 7 shows the comparison of end-to-end delay of ACO, AODV, DSDV and DSR. The ACO has the maximum end-to-end delay in 10 nodes and 100 nodes. The result shows that there is some space for future improvements in the end-to-end delay. DSR as an on demand based protocol has the best performance in this comparison.

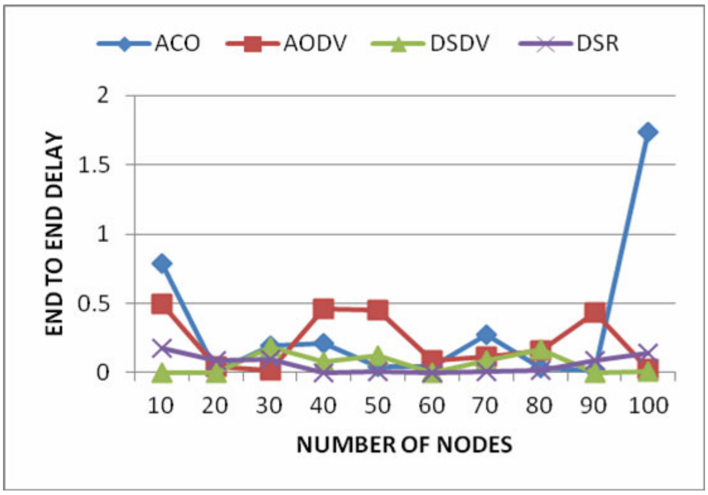

Fig. 7. The comparison of the end-to-end delay in low density traffic

Fig. 8 shows the comparison of the first packet arrival time of the four routing algorithms. ACO and DSR have the best performance in this comparison. It's surprisingly to know that the ACO perform the best in this comparison. Since the ACO discovers the best path based on the accumulation of pheromones, it seems it does not take a very long time for an ant to find the best path.

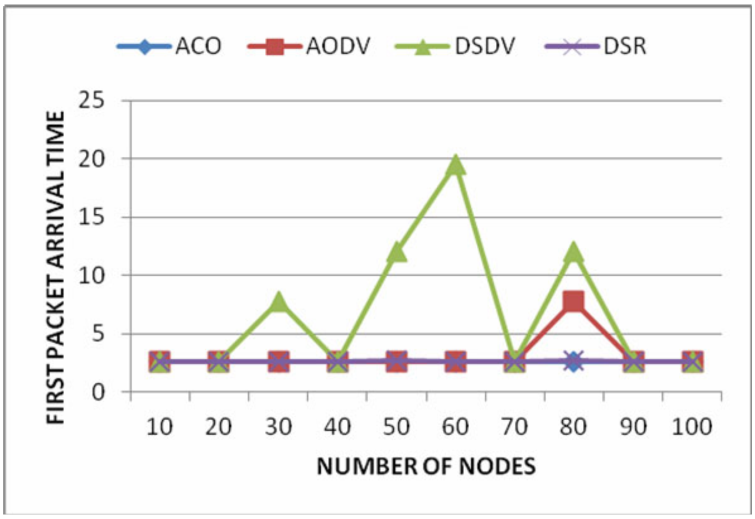

Fig. 8. The comparison of the first packet arrival time in low density traffic

5.2 High Density Traffic

Fig. 9 shows the packet delivery ratios of ACO, AODV, DSDV and DSR in different number of nodes in high density traffic. The DSR has the best performance among these four routing algorithms in the packet delivery ratio in almost all cases. No matter how many nodes are there, the DSR have highest packet delivery ratio, which is close to 100% when the number of nodes is below 70. The packet delivery ratios of AODV, DSDR, and ACO fluctuate when the number of nodes varies. From the perspective of packet delivery, the ACO is not stable than the other three routing algorithms in high density traffic.

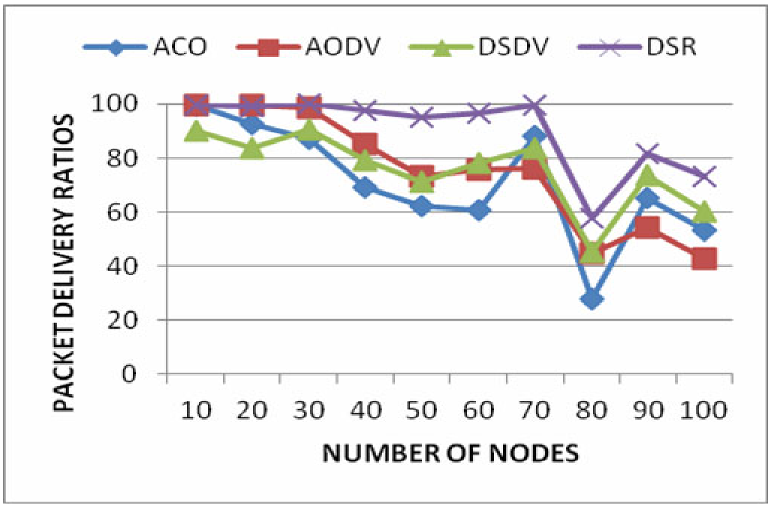

Fig. 9. The comparison of the packet delivery ratio in high density traffic

Fig. 10 shows the comparison of end-to-end delay of ACO, AODV, DSDV and DSR in high density traffic. The ACO and DSR have the maximum end-to-end delay in 80 nodes and 100 nodes. The result shows that there is some space for future improvements in the end-to-end delay. DSR as an on demand based protocol has the best performance in this comparison.

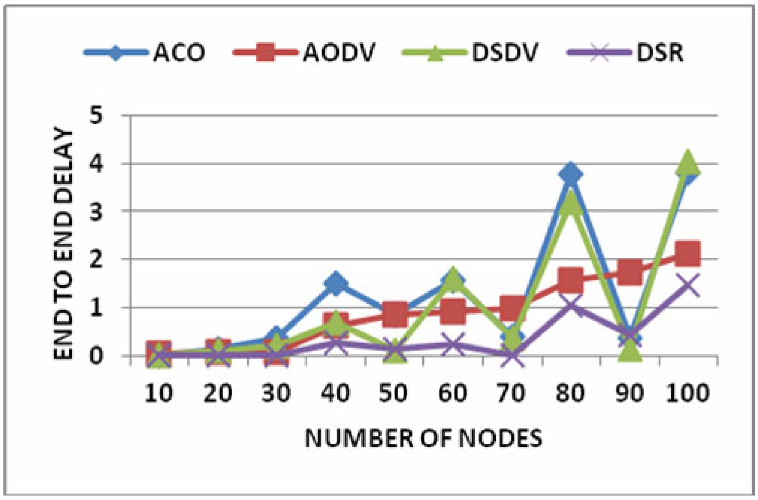

Fig. 10. The comparison of the end to end delay in high density traffic

Fig. 11 shows the comparison of the first packet arrival time of the four routing algorithms in high density traffic. ACO and DSR have the best performance in this comparison. It's surprisingly to know that the ACO perform the best in this comparison. Since the ACO discovers the best path based on the accumulation of pheromones, it seems it does not take a very long time for an ant to find the best path in high density traffic. Table 5 summarizes the performance comparisons among ACO, AODV, DSDV, and DSR.

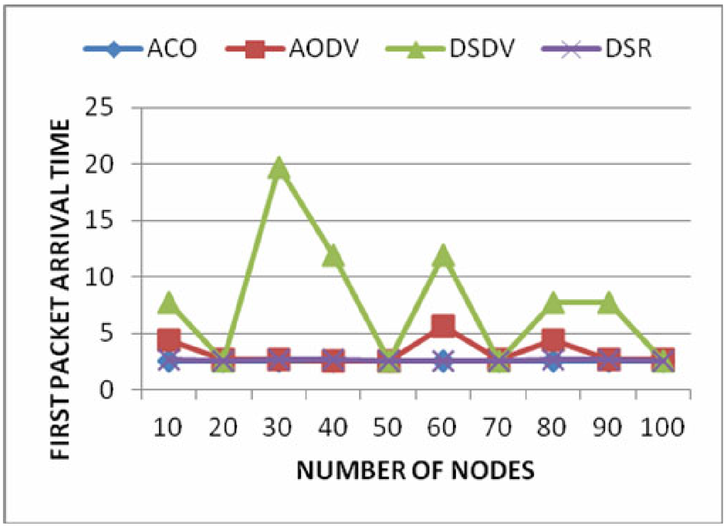

Fig. 11. The comparison of the first packet arrival time in high density traffic

Table 5. The performance comparisons among ACO, AODV, DSDV, and DSR

	ACO	AODV	DSDV	DSR
Route maintain overhead	Low	Low	High	Low
Packet delivery ratio with mobile sensor nodes	High	High	Sometimes low	High
End to end delay with mobile sensor nodes	Sometimes high	Sometimes high	Low	Low
First packet arrival time with mobile sensor nodes	Low	Sometimes high	High	Low

6 Conclusions

In this research, we first proposed an ACO design scheme from the SDN-like perspective. We discover that since SDN conducts the flow-based routing strategy and ant colony optimization (ACO) also works in the same flow-based manner. An ACO based controller design strategy was proposed. We analyzed the performance of four routing protocols, ACO, AODV, DSDV, and DSR in wireless sensor network using ns2. Previous research in ACO focused mainly on energy efficiency. There is a lack of study of ACO performance on other metrics, such as the packet delivery ratio, the end-to-end delay and the first packet arrival time. Our results show that the ACO not only is energy efficient, but also has the highest packet delivery ratio and the shortest first packet arrival time in low density traffic. The end-to-end delay seems higher in the case of 10 and 100 nodes and it is worthy of further investigations in low density traffic. The ACO does not perform well in both the packet delivery ratio and the end-to-end delay with high density traffic, so ACO is more suitable to be used in low density traffic.

Acknowledgement

This research was supported by grants from the Ministry of Science and Technology of Taiwan (MOST 105-2221-E-415 -016).

References

[1] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner, OpenFlow: enabling innovation in campus networks, ACM SIGCOMM Computer Communication Review 38(2)(2008) 69-74.

[2] N. McKeown, Software-defined networking, INFOCOM keynote talk, 2009.

[3] N.P. Preve, E. N. Protonotarios, An integrated sensor web grid cyberimplementation for environmental protection, IEEE Sensors Journal 11(9)(2011) 1787-1794.

[4] J.P. Carmo, P.M. Mendes, C. Couto, J.H. Correia, A 2.4-GHz CMOS short-range wireless-sensor-network interface for automotive applications, IEEE Transactions on Industrial Electronics 57(5)(2010) 1764-1771.

[5] L. Wang, G.-Z. Yang, J. Huang, J. Zhang, L. Yu, Z. Nie, D.R.S. Cumming, A wireless biomedical signal interface system-on-chip for body sensor networks, IEEE Transactions on Biomedical Circuits and Systems 4(2010) 112-117.

[6] H. Kim, R. Chitti, J. Song, Novel defense mechanism against data flooding attacks in wireless ad hoc networks, IEEE Transactions on Consumer Electronics 56(2)(2010) 579-582.

[7] J. Yick, B. Mukherjee, D. Ghosal, Wireless sensor network survey, Computer Networks 52(12)(2008) 2292-2330.

[8] M. Dorigo, L.M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, IEEE Transactions on Evolutionary Computation 1(1)(1997) 53-66.

[9] R. Misra, C. Mandal, Ant-aggregation: ant colony algorithm for optimal data aggregation in wireless sensor networks, in: Proc. 2006 IFIP International Conference on Wireless and Optical Communications Networks, 2006.

[10] V. Kawadia, P. Kumar, Power control and clustering in ad hoc networks, in: Proc. INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications, 2003.

[11] C.E. Perkins, P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, ACM SIGCOMM Computer Communication Review 24(4)(1994) 234-244.

[12] D.B. Johnson, D.A. Maltz, Dynamic source routing in ad hoc wireless networks, in: T. Imielinski, H. Korth (Eds.), Mobile Computing, Kluwer Academic, Norwell, MA, 1996, pp. 153-181.

[13] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, S. Shenker, NOX: towards an operating system for networks, ACM SIGCOMM Computer Communication Review 38(3)(2008) 105-110.