Journal of Computations & Modelling, vol. 12, no. 3, 2022, 13-23
ISSN: 1792-7625 (print), 1792-8850 (online)
https://doi.org/10.47260/jcomod/1232

IoT Actuator Networks Based on Inverse Directed Diffusion

Shun Murata[1], Noriko Matsumoto[2] and Norihiko Yoshida[3]

Abstract

An IoT system comprises servers or a cloud at the center, possibly surrounded by edge cloudlets and fog computing devices, a group of sensors on one side, and a group of actuators on the other side if the purpose of the system is not only data analysis but also automated control.

Actuators are connected to the central servers or a cloud via an actuator network. Due to progress in smart robotics and related technologies, actuator networks are now getting large, wide, and complex, and should be decentralized and self-organizing as well as sensor networks are. However, as far as we know, there have been no noticeable researches on actuator networks yet.

In this paper, we propose a new routing protocol for actuator networks based on an inverse of Directed Diffusion, which is one of the most well-known protocols for sensor networks. Our protocol aims at realizing efficient dissemination and delivery of signals, commands, and data for control, and exhibits good results in simulation-based experiments regarding network traffic reduction.

Keywords: Internet of Things, Actuator networks, Directed Diffusion.

---

[1]Nomura Research Institute, Ltd.
[2]Rissho University, Japan
[3]Rissho University, Japan

Article Info: ***Received:*** September 2, 2022. ***Revised:*** September 28, 2022.
Published online: October 4, 2022.

1 Introduction

An IoT system comprises servers or a cloud at the center, possibly surrounded by edge cloudlets and fog computing devices [1, 2], a group of sensors on one side, and a group of actuators on the other side if the purpose of the system is not only data analysis but also automated control. Actuators includes motors, robot arms, conveyors, drones (UAVs), and unmanned vehicles for example.

Actuators are connected to the central servers or a cloud via an actuator network as well as sensors are connected via a sensor network. Due to progress in smart robotics and related technologies, actuator networks are getting large, wide, and complex for manufacturing, agriculture, and road traffic control for example. Actuator networks are different from conventional multicast networks, content distribution networks, or streaming networks because actuator networks should be decentralized and self-organizing as well as sensor networks are. However, as far as we know, there have been no noticeable researches on actuator networks yet [3], compared to many active researches on sensor networks having already been done for a long time [4, 5].

In this paper, we propose a new routing protocol for actuator networks. It is based on an inverse of Directed Diffusion [6], which is one of the most well-known protocols for sensor networks, and realizes efficient aggregation of sensor data in a decentralized and self-organizing fashion. Our protocol aims at realizing efficient dissemination and delivery of signals, commands, and data for control in a similar fashion to Directed Diffusion.

The rest of this paper is organized as follows. Section 2 summarizes the original Directed Diffusion, and Section 3 describes our proposed protocol, i. e. an inverse of Directed Diffusion, and its improvement. Section 4 presents simulation-based experiments to evaluate our protocol, and their results together with some consideration. Section 5 contains concluding remarks and future work. This paper is a revised and extended version of our previous conference paper [7] with some new experiment results added.

2 Directed Diffusion

Here we summarize the original Directed Diffusion ("DD" hereafter). Although DD is classic, it is still being vigorously studied: J. Sengupta, et al. [8], I. Yasri, et al. [9], J. Mu, et al. [10], and J. Wang, et al. [11] for example.

DD is based on the Publish/Subscribe communication model [12]. In this model, a subscriber expresses an interest in a data item, and when a publisher issues a data item which matches the interest, the subscriber obtains the item. In other terms, a producer publishes a data item in some shared space, and a consumer subscribes to the item which it wants to receive from the shared space. The model realizes asynchronous loosely-coupled communication.

The essence of the DD protocol is as below [6].

(1) Interest propagation:

There is a single node for data collection (subscription) called a sink. In an IoT system, a sink is a server or a cloud. The sink spreads (broadcasts) an interest packet to its neighbor nodes. A node which receives the packet stores the interest information and some information called gradient, and then forwards the packet to its neighbors in a recursive manner. Eventually the interest packet reaches all the nodes in the network. The gradient indicates the neighbor node from which the interest has come.

The data delivery route may be disconnected if a node moves or breaks down. Therefore the sink issues interest packets periodically.

(2) Route establishment:

Each source node issues an exploratory packet to search a route for delivering its data packet. In an IoT system, a source is a sensor. The exploratory packet is relayed recursively following the gradient at each node until it reaches the sink.

The sink, when receiving an exploratory packet, issues a reinforce packet to establish a route to its source node. The reinforce packet is relayed to the opposite direction to the exploratory packet. Each node receiving the reinforce packet "reinforces" its gradient.

(3) Data delivery:

The data packet issued by the source node is delivered along the reinforced gradients so as to reach the sink.

3 Proposed Protocol

Actuator networks should be decentralized and self-organizing, based on asynchronous loosely-coupled communication when they are getting large, wide, complex, and mobile. A packet of signal, command, and data for control should be disseminated and delivered to actuators who need it on demand, instead of being broadcasted to all actuators every time it is issued.

There are some lightweight notwork protocols proposed for IoT, MQTT [13] and CoAP [14], aiming at replacing HTTP(S). MQTT utilizes the Publish/Subscribe communication model, however it requires a central message broker in the network. The broker is the single point of congestion, therefore MQTT is not scalable, nor suitable for large, wide, and complex networks.

Our idea is to apply DD in an inverse manner. The sink and the sources in the original DD become the single source of packets ("origin" hereafter) and destinations of packets ("targets" hereafter) respectively in our protocol. In an IoT system, the origin is a server or a cloud, and a target is an actuator. This proctor is scalable and self-organizing as well as the original DD is.

The inverse version of DD is outlined as below.

(1) Interest propagation

Each target node (actuator) spreads an interest packet to its neighbor nodes, possibly other actuators or relay stations. A node which receives the packet forwards it to its neighbors in the same manner as the original DD does. Eventually the interest packet reaches all the nodes in the network including the origin node (a server or a cloud).

(2) Route establishment:

The origin issues an exploratory packet to search a route for delivering its control packet. The exploratory packet is relayed recursively following the gradient at each node until it reaches the target.

The target, when receives an exploratory packet, issues a reinforce packet to establish a route to the origin. The reinforce packet is relayed in the same manner as in the original DD.

(3) Control delivery:

The control packet issued by the origin is delivered along the reinforced gradients so as to reach the target.

There is an issue to be improved in this inverse DD: network traffic reduction.

In the inverse DD, there are more than one, possibly many targets issuing interest packets, and there may be the same interest packets among them.

If a node receives an interest packet which is the same as another which is already forwarded and corresponds to an established route, this interest packet need not be forwarded any further. This node, instead of the origin, issues an exploratory packet towards a target which have issued the new interest packet, so as to establish a route between itself and the new target.

Suppose there is already an established route between the origin node N_0 and a target node N_1 via another intermediate node N_x for an interest packet i_1 issued by N_1. When N_x receives the same interest packet i_1 issued by another target node N_2, N_x does not forward i_1 any further. Instead, N_x issues an exploratory packet e_1 corresponding to i_1 towards N_2. Eventually, an route for control delivery between N_x and N_2 is established instead of between N_0 and N_2. Notice that the route between N_0 and N_x has already been established.

This improvement reduces the amount of interest packets, exploratory packet, and reinforce packets.

4 Experiments

We conducted some simulation-based experiments to evaluate our proposed protocol. We implemented the inverse DD and the improved inverse DD together with a flooding-based protocol for comparison. The conventional flooding-based protocol is used in actual IoT systems now, and it delivers all packets for control to all targets whenever they are issued regardless of when and which target needs the packet.

- A virtual field for the experiments are 1000×1000 square virtual length unit ("VLU" hereafter) wide.
- In the field, there is a single server (origin) at the center, and there are no more than 100 actuator nodes (targets) at random locations.
- All the nodes have communication range of 100 VLU radius. A node receiving a packet from a neighbor node may be unable to forward it if there is no other nodes within its range.

- All the packets has an upper limit 10 for hop (recursive forwarding) counts so as to prevent infinite forwarding. A packet may not always reach its destination in case it is too far.
- Interest packets include 10 classes of interests. Two interests are regarded as the same if their classes are the same.
- Each trial lasts 5,000 virtual time unit ("VTU" hereafter). Each result presented in the charts below is an average of ten trials.

Figure 1 (left) proves that the inverse DD reduces the whole amount of packets significantly in the actuator network compared to the conventional flooding-based protocol, owing to the selective nature of the publish/subscribe communication, and our improved protocol reduces even better. Figures 1 (right) and 2 show the details broken down into data, interest, and exploratory packets respectively according to the number of target nodes. The amount of reinforce packets also decreases, however they occupy only 0.3% of the whole amount, therefore negligible. The flooding-based protocol does not use interest, exploratory, nor reinforce packets, therefore not included in Figures 2.

As stated earlier, a packet may not always reach its destination, and there is possibility to fail. This is partly because of the communication range limit for a node and partly because of the hop count limit for a packet. Figure 3 (left) shows the success ratio of data delivery, and Figure 3 (right) shows the time in VTU required for data delivery. When the target nodes increase, The hop counts, i. e. how many times the packet is forwarded, of interest packets and exploratory ones increase even more, and exceed the maximum more often. This is the reason why the inverse DD scores the worst when all the 100 nodes are targets in Figure 3. On the other hand, the improved inverse DD protocol suppresses the interest and exploratory packets, so as to avoid this performance degradation.

Figure 4 and 5 show the number of packets and their details according to the number of interest classes comparing the inverse DD and the improved inverse DD. The number of targets is 50 in all the experiments hereafter. Again, the improved inverse DD outperforms the inverse DD in every chart.

In Figure 6 (left), which shows the success ratio of data delivery, the improved inverse DD makes worse scores than the inverse DD. In the sparse case of 50 targets, a target must have less neighbor nodes within its communication

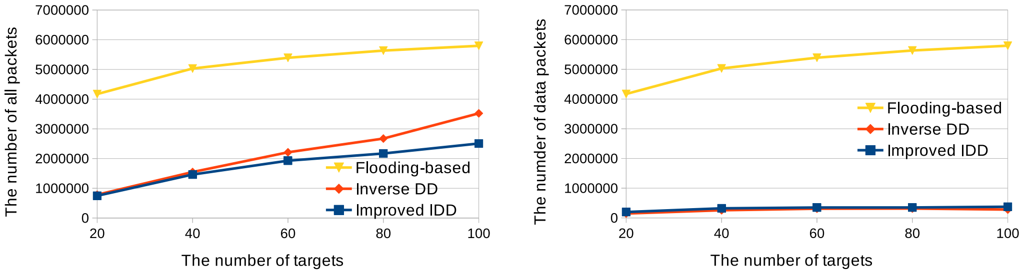

Figure 1: The number of all packets (left) and data packets (right)

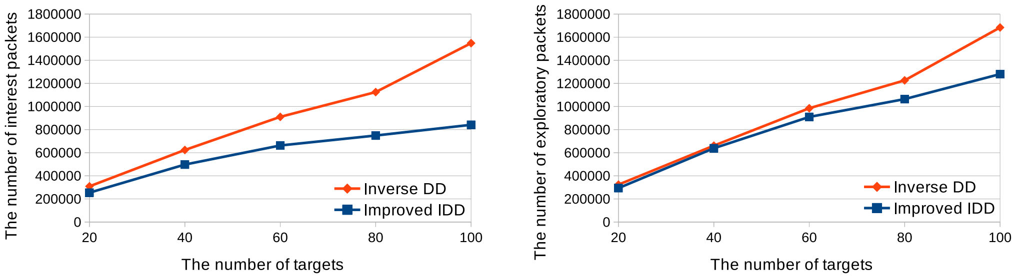

Figure 2: The number of interest packets (left) and exploratory packets (right)

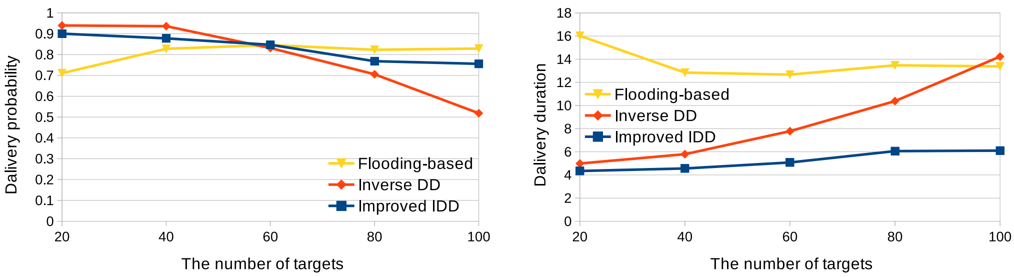

Figure 3: Delivery probability (left) and duration (right) of data packets

range, and may cause communication failure more often. In the dense case of more than 60 targets, the improved inverse DD makes better scores than the inverse DD as shown in Figure 3 (left).

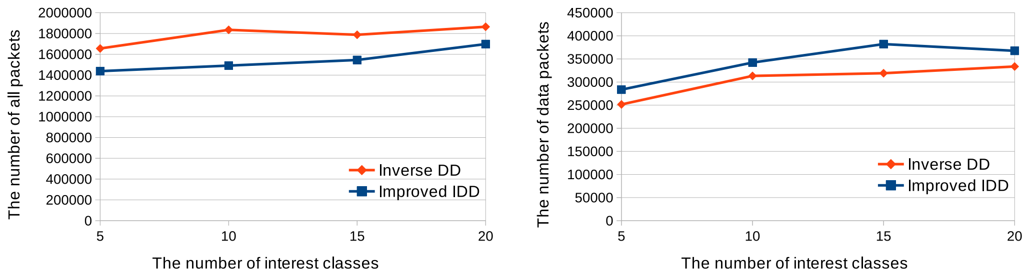

Figure 4: The number of all packets (left) and data packets (right)

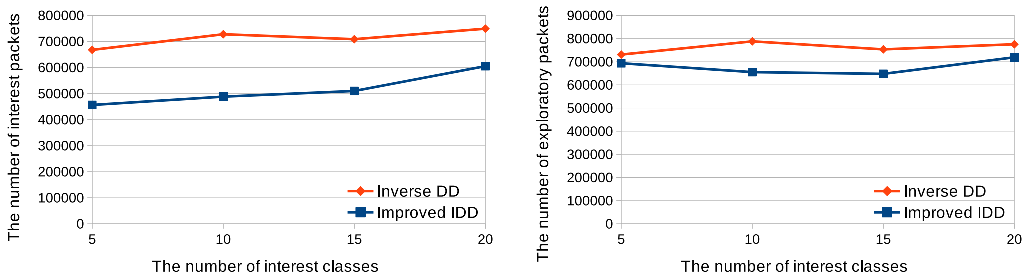

Figure 5: The number of interest packets (left) and exploratory packets (right)

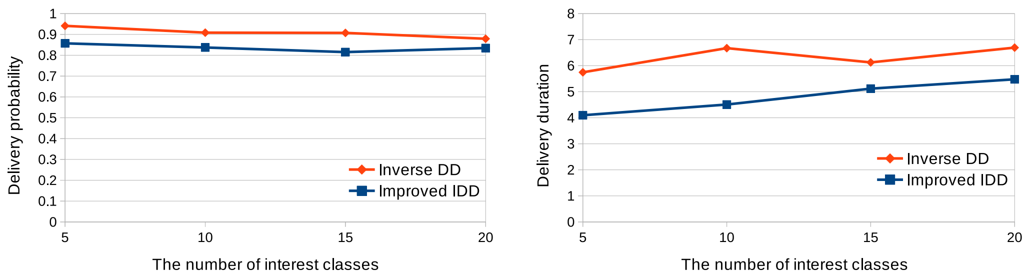

Figure 6: Delivery probability (left) and duration (right) of data packets

5 Concluding Remarks

In this paper, we propose a new routing protocol for IoT actuator networks, which deliver signals, commands, and data for control to actuators. The proposed protocol is based on an inverse of Directed Diffusion, with some improvement for network traffic reduction. We presented by simulation-based experiments that this protocol outperforms the conventional flooding-based protocol used in actual IoT systems now. We suppose the selective nature of our protocol makes the resulting actuator networks efficient in terms of network traffic.

We have obtained some preliminary, yet promising results, however we are still at the beginning of this research, and there are still many issues which must be addressed. One of the most important issues is how the network follows up when any actuator moves around.

Some actuators, drones and vehicles for example, moves around. Then the shape of their network changes, and a route established for delivery is no longer valid. A new route will be established when the actuator issues an interest packet again, however it takes time. A similar problem regarding node mobility is addressed in the technology of Information Centric Networking (ICN), and combination of mobility prediction and proactive caching has been proposed and thought to be promising [15, 16, 17]. We are now trying to apply the idea underlying these preceding related studies to mobility follow-up in actuator networks.

References

[1] M. Laroui, B. Nour, H. Moungla, M. A. Cherif, H. Afifi, and M. Guizanie, "Edge and Fog Computing for IoT: A Survey on Current Research Activities and Future Directions,", Computer Communications, Vol.180, 2021, pp.210–231.

[2] Y. Kalyani and R. Collier, "A Systematic Survey on the Role of Cloud, Fog, and Edge Computing Combination in Smart Agriculture," Sensors, Vol.21, No.5922, 2021, 27 pages.

[3] S. K. Lee, M. Bae, and H. Kim, "Future of IoT Networks: A Survey," Applied Science, Vol.7, No.1072, 2017, 25 pages.

[4] J. N. Al-Karaki and A. E. Kamal, "Routing Techniques in Wireless Sensor Networks: a Survey," IEEE Wireless Communications, Vol.11, No.6, 2004, 6–28.

[5] C. Nakas, D. Kandris, and G. Visvardis, “Energy Efficient Routing in Wireless Sensor Networks: A Comprehensive Survey,” Algorithms, Vol.13, No.72, 2020, 66 pages.

[6] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, “Directed Diffusion for Wireless Sensor Networking,” IEEE/ACM Transactions on Networking, Vol.11, No.1, 2003, pp.2–16.

[7] S. Murata, N. Matsumoto, and N. Yoshida, “Directed Diffusion Applied to Actuator Network in IoT,” Proc. 15th International Conference on Applied Computing, 2018, pp.277–284.

[8] J. Sengupta, S. Ruj, and S. D. Bit, “An Efficient and Secure Directed Diffusion in Industrial Wireless Sensor Networks,” Proc. ACM 1st International Workshop on Future Industrial Communication Networks, 2018, pp.41–46.

[9] I. Yasri and E. Safrianti, “Directed Diffusion Modelling for Tesso Nilo National Parks Case Study,” Materials Science and Engineering, No.300, ID:012039, 2018, 7 pages.

[10] J. Mu, X. Yi, X. Liu, and L. Han, “An Efficient and Reliable Directed Diffusion Routing Protocol in Wireless Body Area Networks,” IEEE Access, ID:2019.2914241, 2019, 10 pages.

[11] J. Wang, X. Liu, Z. He, Y. Zhang, and S. Gao, “An Improved Triggering Updating Method of Interest Message with Adaptive Threshold Determination for Directed Diffusion Routing Protocol,” Journal of Sensors, ID:8814839, 2021, 15 pages.

[12] P. T. Eugster P. A. Felber R. Guerraoui, and A.-M. Kermarrec, “The Many Faces of Publish/Subscribe,” ACM Computing Surveys, Vol.35, No.2, 2003, pp.114–131.

[13] Message Queueing Telemetry Transport (MQTT): http://mqtt.org [viewed on August 28th, 2022]

[14] The Constrained Application Protocol (COAP): https://coap.technology/ [viewed on August 28th, 2022]

[15] N. Abani, T. Braun, and M. Gerla, "Proactive Caching with Mobility Prediction under Uncertainty in Information-Centric Networks," Proc. 4th ACM Conference on Information-Centric Networking, 2017, pp.88–97.

[16] H. Khelifi, S. Luo, B. Nour, A. Sellami, H. Moungla, and F. Naït-Abdesselam, "An Optimized Proactive Caching Scheme Based on Mobility Prediction for Vehicular Networks," Proc. 2018 IEEE Global Communications Conference, 2018, 6 pages.

[17] H. N. Arifin, L. V. Yovita, and N. R. Syambas, "Proactive Caching of Mobility Prediction Prefetch and Non-Prefetch in ICN," Proc. 2019 International Conference on Electrical Engineering and Informatics, 2019, pp.418–422.