Differences in Robustness of Self-Organized Control and Centralized Control in Sensor Networks Caused by Differences in Control Dependence

Masashi Sugano
School of Comprehensive Rehabilitation
Osaka Prefecture University
3-7-30, Habikino, 583-8555 Osaka, Japan
sugano@rehab.osakafu-u.ac.jp

Yuichi Kiri and Masayuki Murata
Graduate School of Information
Science and Technology,
Osaka University
1-5, Yamadaoka, Suita, 565-0871 Osaka, Japan
{y-kiri, murata}@ist.osaka-u.ac.jp

Abstract

Self-organized control has received significant attention in the area of networking, and one of the main factors for this attention is its robustness. However, it should be stressed that deciding whether self-organized control is robust or not is not a trivial task. Even if it is in fact robust, the factors underlying its robustness have not yet been explored in sufficient detail. In this paper, we provide the first quantitative demonstration of the superior robustness of self-organized control through comparison with centralized control in a sensor network scenario. Through simulation experiments, we show that self-organized control maintains the functionality of its data collection even in a variety of perturbations. In addition, we point out that the difference in the robustness of the abovementioned control schemes stems from the degree to which the comprehension of a given node about the state of the network depends on information obtained from other nodes.

1. Introduction

As networks are becoming increasingly larger and more complex, a critical issue in today's dynamically changing and uncertain environments is to maintain the functionality of networks in a manner which allows them to adapt to environmental changes. A control scheme which maintains the performance even when the network state changes dramatically or unforeseeable circumstances occur is preferable for present and future networks, even if the basic network performance in such cases is inferior to that of networks operating with other control schemes. The property which allows a system to maintain its functionality despite external and internal perturbations is called "robustness" [5]. In this age when networks play an essential role in our everyday lives, the robustness of networks is becoming increasingly important.

Distributed control has been said to be superior to centralized control with respect to robustness. Currently, a type of distributed control scheme which is beginning to attract considerable attention is one of self-organized control [2, 7]. In this control scheme, each component autonomously decides the following action on the basis of local information, and the simple microscopic actions of the components collectively provide structure and functionality at macroscopic level without any centralized coordination [6]. Such behavior is distinct from plain distributed control, where individual components act autonomously but depend on global information. Although scalability, adaptability, and fault tolerance, which are included in the concept of robustness in a broad sense, are "known" as properties inherent to self-organized control, we stress that this knowledge is certainly not trivial. Even assuming that the notion of robustness is true, to the best of our knowledge the reasons why self-organized control is robust and the factors which determine the superiority of its robustness as compared to other control schemes have not been examined with sufficient rigor.

In our previous work [3, 4], we provided quantitative evidence of the robustness of self-organized control with respect to transmission errors and node failures, and concluded that the robustness of the self-organized control scheme is superior to that of other control schemes. However, since sensor networks face a wider range of perturbations, the purpose of this paper is to demonstrate the advantages of self-organized control against perturbations different from those in our previous work. Furthermore, based on the results of the evaluation, we also pose interesting questions such as why and how self-organized control is robust. In this paper, we first demonstrate the superior robustness of self-organized control by comparing it with centralized control, as in our previous work. Furthermore, from the results of this comparison, we point out that the difference

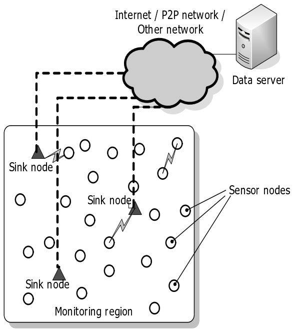


Figure 1. Network model.

in the robustness is derived from the degree to which the comprehension of a given node about the state of the network depends on information from other nodes. This is the key to differentiating the degrees of robustness of those two control schemes.

2 Self-organized and centralized control schemes in sensor networks

We provide an overview of our self-organized and centralized control schemes, which are the subjects of robustness evaluation in the present study. The operations of both control schemes are based on the premise that multiple sinks are deployed in their respective monitoring regions. Using this multi-sink configuration, both control schemes take a cluster-based approach, in which the same number of node clusters and sinks is formed, and individual sensor nodes transmit their sensed data to the sink located in their cluster (Fig. 1). The reader may refer to [3, 4] for more details.

2.1 Self-organized control

We used a data gathering scheme we proposed earlier [3] as the self-organized control scheme. This scheme is based on pheromone-mediated ant-swarm behaviors, i.e., ant colony optimization [1] and ant clustering [9]. Sensor nodes are divided into as many clusters as there are sinks by using ant clustering based on a “cluster pheromone”, and routing is also performed in each cluster by using a “routing pheromone”. In our scheme, sinks flood control packets called “backward ants”, whose role is to produce a gradient of the routing pheromone concentrations in which next-hop nodes which are better suited for handling the traffic have higher concentrations of the routing pheromone. Using these concentrations, each node stochastically selects the next-hop node in the same manner real ants are attracted to higher concentrations of pheromones. Sensor nodes also stochastically choose their cluster membership on the basis of the cluster pheromone, which is calculated from the concentration of their routing pheromone.

Sensor nodes are prone to failure due to their cheap production cost. Moreover, their power is inevitably depleted during long periods of operation of the sensor network. Therefore, it is necessary to detect these failures and take appropriate countermeasures in order to be able to gather data over long periods of time. We applied a soft-state model for detecting failures by using a periodically transmitted “hello” message. If node n_j does not receive a hello message from node n_i after the predefined expiry time t_{expire}, node n_i is deemed to have failed, and node n_j eliminates node n_i from the candidate set of its next-hop nodes for n_j. Detecting a sink node failure is also based on the same soft-state model, as sinks periodically broadcast hello messages similarly to other sensor nodes.

2.2 Centralized control

The centralized control scheme used here is based on [8], with a number of appropriate modifications. We assume the existence of a control station, which is wired to all sinks. The station knows the initial power and locations of all nodes and sinks, and manages the overall network. The station initially divides the nodes into as many clusters as there are sinks by Voronoi tessellations, with sinks as base points. After the clusters are determined, the station constructs routes from each node to a sink, which minimizes the total link cost. Eventually, the station transmits a command packet, which includes the route information, to the sensor nodes via the sinks.

The detection of node failures in the centralized control scheme is the same as in the self-organized control scheme explained above. However, an explicit failure indication packet must be transmitted to the control station, since new routes must be provided such that packets circumvent the failed node. Even when node n_i works properly, it is possible that hello messages from n_i do not arrive at its neighboring nodes within t_{expire} due to interference or transmission errors. As a measure against such false detections, if n_j receives a hello message from node n_i after the time t_{expire} has passed, it regards the detection of the failure of node n_i as false-positive, and transmits a failure recovery packet in order to inform the station about the false detection. The station then recomputes new routes and transmits them to the sensor nodes.

3 Evaluation and discussion

3.1 Simulation Environment

We implemented our self-organized and centralized control schemes on the ns-2 network simulator. In the following experiments, it is assumed that we have randomly placed 300 sensor nodes over a square monitoring region

Table 1. Simulation parameters.

Communication range	10 m
t_{hello}	1 s
t_{expire}	5 s
Size of a hello packet	10 bytes
Size of a failure detection packet	10 bytes
Size of a failure recovery packet	10 bytes
Size of a data packet	64 bytes

with a side of 100 m, unless stated otherwise. Furthermore, it is assumed that there are four sinks at locations $(25, 25)$, $(75, 25)$, $(25, 75)$, $(75, 75)$ within the monitored region, where the numbers indicate distance in meters from one side of the square. Although we performed tests for other sink positions as well, the obtained results were almost the same.

We used the MAC and PHY layers followed the IEEE 802.15.4 specification. Since the size of the command packet in the centralized control simulation can easily exceed the value specified in IEEE 802.15.4, we set *aMaxPHYPacketSize*, which determines the maximum length of a packet, to infinity. The size of the command packet transmitted from sink S_j is calculated by $\sum_i 6 \cdot e_{n_i} \cdot num_{S_j} + 7$ where e_{n_i} is the number of previous- and next-hop node pairs assigned to node n_i, and num_{S_j} is the number of sensor nodes within cluster S_j. We assume that 6 bytes are enough for a pair, and that 7 bytes are enough for a header. The simulation parameters are also listed in Table 1. We do not assume an error correction system like FEC, and therefore the packet is discarded even if an error occurs in a single bit. In the data collection model described below, sensor nodes send the information they obtain to their sinks in a multi-hop way at a predefined interval t_{intval}=10 s. Sensor nodes do not synchronize with each other, and the transmission time of any node is independent of that of the others.

One of the most important metrics for sensor networks is the reliability with which information is brought to a sink. We therefore defined a metric called the "data collection rate", where the number of sensor nodes which work properly is N_{act}, and consequently the number of data packets generated within t_{intval} is N_{act}. When the number of packets reaching a given sink is r, the data collection rate is defined as r/N_{act}.

3.2 Measures against sink failure

Figure 2 presents the results for the case in which a sink located at $(25, 25)$ fails at 400 s. After the sink failure, the data collection rate drops sharply to about 75%, except in the case of centralized control with 10^{-5} BER (Bit Error Rate), where the rate drops to only 90%. A rate of 75% means that one cluster suffered catastrophic damage (the ratio of data packets gathered within a cluster is about 25%). Not only is the drop in the data collection rate in the case of centralized control and low BER small, but also the recovery is almost immediate. The control station which is wired to the sinks becomes aware of the failure within a short amount of time (in our simulations, it is set to 0 s), after which the clusters are reconstructed and the routes are recomputed upon receiving the command packet, in order to adapt the whole network to the failure. Sensor nodes immediately modify their cluster membership and routing table according to the instructions contained in the command packet, and the data collection rate is restored soon after that. Indeed, in cases where the channel quality is poor, the data collection rate in the centralized control scheme is unable to recover within the simulation time shown in Fig. 2, since centralized control is weak with respect to transmission errors, as indicated in [4].

In contrast to the centralized control scheme, the self-organized control scheme needs more time for the distant sensor nodes to adapt to the sink failure. In addition, since the network has no supervisor and no explicit instructions, some nodes might be prone to taking contradicting actions based on the possibility of receiving inaccurate information about the condition of the network. For these reasons, in low BER environments, the self-organized control scheme exhibits worse recovery than the centralized one. In high BER environments, however, the relationship between self-organized control and centralized control is reversed, since the self-organized control scheme inherently does not have critically important information whose loss can bring serious and adverse influence to the network.

3.3 Measures against node failure

We already demonstrated the robustness against node failure in our previous work [4]. Moreover, we showed that although most of the sensor nodes other than the failed ones exhibit data collection rates of about 100% in the self-organized control scheme, failures in the case of the centralized control scheme have considerable influence on the data collection rates at the cluster level, where many sensor nodes are unable to transmit packets to their sinks, and this influence is especially notable when concentrated and simultaneous failures occur. However, when we tested random failures in a 100 m ×100 m monitoring region containing 300 nodes, the difference in the robustness of the self-organized and the centralized control schemes was not clear due to the connectivity degradation caused by the continual node failures. Therefore, here we temporarily used a narrower monitoring region of 50 m × 50 m while keeping the number of nodes and sinks, and defined p_{fail} as the failure rate per second for each sensor node.

The variances of the data collection rates of both control schemes among trials are shown in Fig. 3. The variance in the self-organized control scheme is small and not as sensitive to the failure rates. However, in the centralized control scheme, the data collection rates in some trials experience

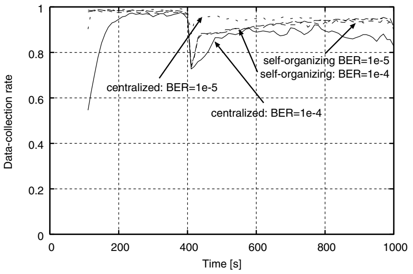


Figure 2. Features of the process of recovery from sink failure.

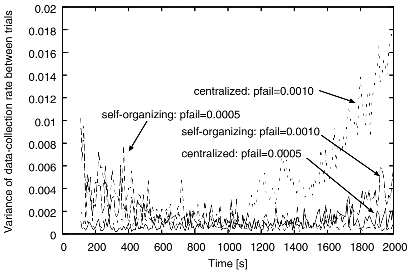
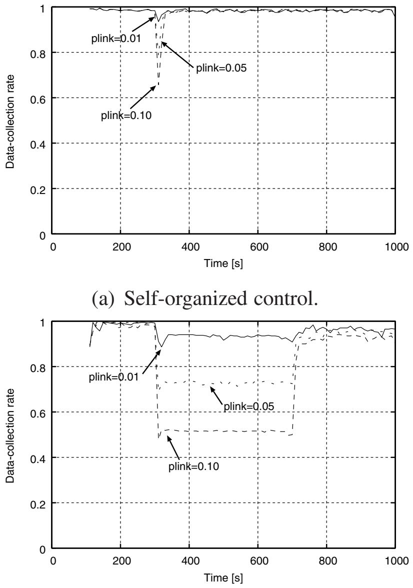


Figure 3. Variances of the data collection rates among trials.



(a) Self-organized control.

(b) Centralized control.

Figure 4. Influence of link disconnections on the data collection rate.

sudden drops, which lead to the higher variance of the data collection rates, as shown in Fig. 3. The high variance in the case of centralized control indicates the difficulty of predicting the data gathering capability in harsh environments, although all of the plots are prepared using the same parameters.

3.4 Measures against link disconnection

As links can become disconnected intermittently in wireless networks, in the case where the link between nodes n_i and n_j is disconnected but the link between n_i and n_k is still connected, there is a possibility that the status of n_i as seen from the perspective of n_j and n_k is inconsistent. Therefore, in order to study the differences in the robustness of the two schemes, we randomly disconnected a percentage of the links. We assume that each node is linked to an arbitrary neighboring node, and each link is disconnected with probability p_{link} in both directions. This disconnection process was conducted for all nodes, and the duration of the disconnection was 400 s, from t=300 s to t=700 s.

In the results shown in Fig. 4, the data collection rate in the self-organized control scheme immediately recovers to the rate before the disconnection, although it experiences a declination for a short amount of time. The centralized control scheme, on the other hand, suffers greatly from the disconnections, where detection of massive node failures occurs since neighboring nodes regard disconnected nodes as failed due to their inability to transmit hello messages. In other words, sensor nodes cannot distinguish failures from link disconnections in our centralized control scheme. Furthermore, after the detection of a missing link, the neighboring nodes transmit failure-indication packets, which are in fact false-positive detection packets, to the control station. As a result, the control station does not provide routes to the node which is considered as failed, and the packets from the disconnected node are discarded, which is the main reason for the decay of the data detection rate in Fig. 4(b).

4 Dependence on control information

4.1 Factors influencing the difference in robustness

In the evaluation presented in Section 3 and in previous works, there was a significant difference between the robustness of self-organized control and centralized control. We are inclined to explain this trend in terms of "dependence on control information". In this case, "dependence" has almost the same meaning as that used in fault management. The dependence is a relation in which an error or failure in an object may cause an error or failure in another object. We define control information as the information exchanged between entities of a given network which coordinates their joint operation.

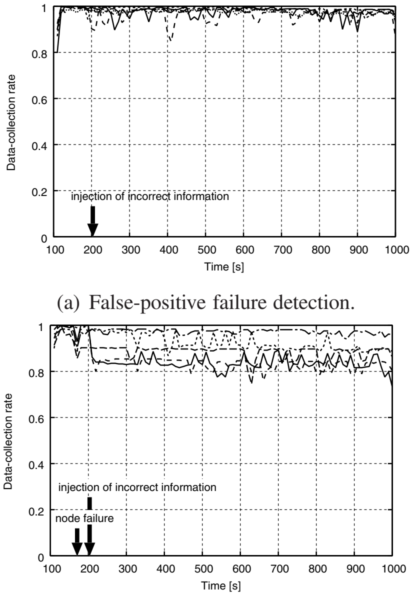
In Sections 3.3 and 3.4, even the control station itself did not comprehend the correct state of the network. This is caused by the fact that the control station also depends on control information received from the nodes in the network. The control station constructs a precise view of the whole network by integrating each piece of information about the state of the network. In other words, the problem of the dependence is that the control information from potentially unreliable nodes in environments where reliable communication is not guaranteed plays a critical role in generating the control scheme at the control station. In Section 3.3, failure indication packets, which notify the command node about the correct state of the network, did not reach the control station, resulting in a sudden drop of the data collection capability of the clusters. In Section 3.4, one node considers a neighboring node to be operating correctly, while another node considers the same neighboring node as faulty, resulting in the transmission of failure indication packets even though no nodes have failed. In this way, information which does not reflect the correct state of the network brings vulnerability to the centralized control scheme.

Of course, at the node level, self-organized control is identical to centralized control, meaning that individual nodes potentially have an erroneous understanding about the state of the network. However, individual nodes affect only their surrounding environment or neighboring nodes since all nodes have only partial view of the network, and do not transmit or receive explicit control information. Due to this behavior, the influence of individual nodes on the global state of the network is much smaller than in the centralized control scheme. In this regard, since we have not yet clarified the influence of erroneous information received from individual nodes, in the next section we verify our idea by deliberately injecting incorrect information into the network.

4.2 Influence of incorrect information

The purpose of this demonstration is to determine how strong the influence of information received from individual nodes is, as well as how potentially unreliable nodes affect the behavior of the whole network. Therefore, in this section, we deliberately inject spurious information in order to show unambiguously the influence of information received from individual nodes on the functionality of the network. At first, in the centralized control scheme, we considered two scenarios: 1) we injected false-positive failure detection packets, which convey the misinformation that a properly working node is detected as failed, and 2) false-recovery packets, which inform the surrounding nodes that a node which has failed is detected as recovered.

Although we deliberately injected incorrect information at t=200 s that the node nearest to the coordinate $(25, 25)$ had failed, there was no fluctuation or drop in the data collection rate due to the injection, as seen from the results shown in Fig. 5(a). In fact, the node which was wrongly detected as failed was not able to send its packets to the sink as the control station did not consider the failed node as a member of the data collection cluster. However, routing information was supplied to the other sensor nodes correctly, and thus the influence of the erroneous information was limited.



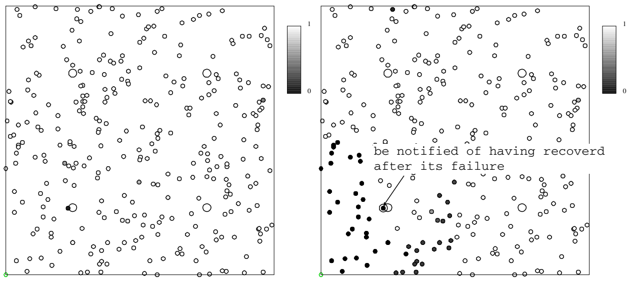
(a) False-positive failure detection.

(b) False-recovery indication.

Figure 5. Results of injecting incorrect information.

Next, we tested the scenario where incorrect information about the recovery of a node is injected into the network. At first, we made the node nearest to the coordinate $(25, 25)$ fail at t=160 s, followed by the injection of information that the node has recovered at t=200 s. Figure 5(b) shows the results of five trials, and it is clear that the behavior of the data collection rates are different among them, i.e., they are different depending on the node deployment. There is a clear drop in two of the plotted lines just after the injection of erroneous information at t=200 s. Given this factor, focusing on one of those lines, in Fig. 6 we visualized the data collection rate of the individual nodes from the time when node fails (t=160 s) until the injection of misinformation (t=200 s), and from the injection (t=200 s) to the end of the simulation (t=1000 s), respectively. As shown in Fig. 6(a), the influence of the node failure can be limited. However, after the injection, data collection in the larger part of the respective cluster becomes impossible.

Self-organized control does not have any means for explicit indication of failure or failure recovery. Therefore, it was impossible to compare it directly with the centralized control in terms of the influence of erroneous information. Instead, we used the indication of sink failure, which is a message which explicitly conveys information about the failure of a sink to the neighboring nodes by using a hello message. Furthermore, we made the sensor node nearest to



(a) From t=160 s to t=200 s (b) From t=200 s to t=1000 s

Figure 6. State of the network after injecting false-recovery information.

the coordinate $(25, 25)$ transmit the information about the sink failure. This indication is spread over the respective cluster through forwarding by nodes which receive the indication.

As a result, although spurious sink failure indication was injected into the network at $t = 200$ s, there was no clear difference in the data collection rate before and after the injection, as seen from the data collection rates from five trials presented in Fig. 7. In our self-organized control scheme, sensor nodes invalidate their membership to the respective cluster upon receiving the sink failure indication, and negative influence was expected due to the dynamic change of cluster membership. However, contrary to our expectation, the cluster memberships were restored to those before the injection. In other words, correct information from other nodes naturally adjusts the situation caused by erroneous information, and this fact contributes to the robustness of self-organized control.

5 Conclusion

In spite of growing interest, there are many points regarding self-organization which remain insufficiently understood. In this paper, we studied the robustness of self-organized control against a wide range of perturbations by comparing it with centralized control, and we attempted to answer some important questions. One such question is whether self-organized control is in fact robust, and we quantitatively demonstrated the affirmative answer by examining various scenarios. Although this result is not surprising, it was found that self-organized control has the obvious benefit of superior robustness, especially if applied to systems in dynamically changing environments, although at the cost of reduced system predictability. Furthermore, the questions about why self-organized control is robust and what factors determine the robustness of self-organized control were also addressed, and based on the results obtained from the simulation experiments, we arrived at the conclusion that the dependence on the control information in the system plays a critical role in determining whether or not the robustness is sufficient. In a network which is composed of potentially unreliable nodes and is located in a harsh environment, decreasing the dependence on the control information received from the nodes is critical to yielding sufficient robustness, and self-organized control inherently possesses such properties.

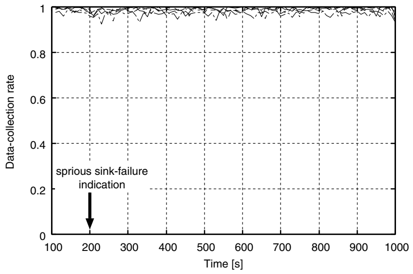


Figure 7. Influence of erroneous sink failure indication.

Acknowledgements

This research is partially supported by Ministry of Education, Culture, Sports, Science and Technology, Japan, Grant-in Aid for Scientific Research (C) 19500060.

References

[1] M. Dorigo, V. Maniezzo, and A. Colorni. The ant system: Optimization by a colony of cooperating agents. *IEEE Trans. Systems, Man, and Cybernetics*, 26(2):29–41, 1996.

[2] C. Gershenson and F. Heylighen. When can we call a system self-organizing? In *Proc. 7th European Conference on Advances in Artificial Life*, pages 604–614, Sept. 2003.

[3] Y. Kiri, M. Sugano, and M. Murata. Self-organized data-gathering scheme for multi-sink sensor networks inspired by swarm intelligence. In *Proc. 1st IEEE Intl. Conf. on Self-Adaptive and Self-Organizing Systems*, July 2007.

[4] Y. Kiri, M. Sugano, and M. Murata. Robustness differences between bio-inspired control and centralized control. In *Proc. of Biological Approaches for Engineering Conference*, Mar. 2008.

[5] H. Kitano. Biological robustness. *Nature Review Genetics*, 5(11):826–837, Nov. 2004.

[6] C. Prehofer and C. Bettstetter. Self-organization in communication networks: Principles and design paradigms. *IEEE Communications Magazine, Feature Topic on Advances in Self-Organizing Networks*, 43(7):78–85, July 2005.

[7] T. D. Seeley. When is self-organization used in biological systems? *Biological Bulletin*, 202:314–318, June 2002.

[8] M. Younis, M. Youssef, and K. Arisha. Energy-aware routing in cluster-based sensor networks. In *Proc. 10th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems*, Oct. 2002.

[9] D. Zaharie and F. Zamfirache. Dealing with noise in ant-based clustering. *IEEE Trans. Evolutionary Computation*, pages 2395–2402, Sept. 2005.