Autonomous Relay Device Placement Algorithm for Avoiding Cascading Failure in D2D-Based Social Networking Service

Hanami YOKOI†(a), Member and Takuji TACHIBANA†(b), Senior Member

SUMMARY In this paper, in order to avoid the cascading failure by increasing the number of links in the physical network in D2D-based SNS, we propose an autonomous device placement algorithm. In this method, some relay devices are placed so as to increase the number of links in the physical network. Here, relay devices can be used only for relaying data and those are not SNS users. For example, unmanned aerial vehicles (UAV) with D2D communication capability and base stations with D2D communication capability are used as the relay devices. In the proposed method, at first, an optimization problem for minimizing node resilience which is a performance metric in order to place relay devices. Then, we investigate how relay devices should be placed based on some approximate optimal solutions. From this investigation, we propose an autonomous relay device placement in the physical network. In our proposed algorithm, relay devices can be placed without the complete information on network topology. We evaluate the performance of the proposed method with simulation, and investigate the effectiveness of the proposed method. From numerical examples, we show the effectiveness of our proposed algorithm.

key words: SNS, D2D, D2D-based SNS, relay device, optimization problem, robustness, heuristic algorithm

1. Introduction

Device-to-device (D2D) communication is one of the promising technologies for the future Internet of Things (IoT) [1]–[9]. New applications will be available over future Internet by using D2D communication and Internet connection. Social networking service (SNS) applications [10]–[13] are also available among neighboring users in D2D communication areas [14], [15]. In such D2D-based SNS, the data transmission over SNS can be performed only when D2D communications over the Internet are not available. Therefore, D2D-based SNS is expected as one of the most popular communication tools in future Internet.

Currently, D2D-based SNS applications have been utilized worldwide in various cases, especially when the Internet is not available due to disaster, pro-democracy protests, or no WiFi access [16].

Here, as shown in Fig. 1, the data transmission in D2D-based SNS is represented as a data transmission over two layers of networks; one is a physical network and the other is a social network [17]. Over the social network, the data transmission is performed directly based on friendships among two users and the data is not transmitted in two or more hops. On the other hand, over the physical network, the data transmission is performed via multiple mobile devices with the D2D communication. Thus, the performance of the data transmission is affected by both friendships in SNS and positions of mobile devices.

In such a data transmission, when a failure occurs in mobile devices*, other failures occur sequentially in the social and physical networks. This is called cascading failure [17], and the cascading failure degrades the performance of D2D-based SNS. In order to spread D2D-based SNS as a future Internet application, it is indispensable to avoid the cascading failure. Therefore, several methods have already been proposed [18], [19]. However, these methods are not so effective in D2D-based SNS because the complete information on network topology is needed. Therefore, the authors have proposed an autonomous distributed method that does not utilize the complete information on network topology [20], [21]. This method adds some virtual temporal friendships between two users in order to avoid the cascading failure. By increasing the number of links over the social network, the impact of cascading failure can be reduced. However, the number of links can not be increased for avoiding the cascading failure in the physical network.

In this paper, in order to avoid the cascading failure by increasing the number of links in the physical network in D2D-based SNS, we propose an autonomous device placement algorithm. In this method, some relay devices are placed so as to increase the number of links in the physical network (see Fig. 2). Here, relay devices can be used only for relaying data and those are not SNS users. For example, unmanned aerial vehicles (UAV) with D2D commu-
The novelty and contribution of this paper is summarized as follows:

- Relay devices are utilized to avoid the cascading failure in D2D-based SNS.
- Robustness of D2D-based SNS application can be improved in an autonomous way.
- Optimization problem is formulated for placing relay devices in D2D-based SNS.

Moreover, the proposed autonomous relay device placement increases the number of links in the physical network, although our previous method in [20], [21] increases the number of links in the social network. Our proposed method is used in D2D-based SNS that has been operated and managed by network management staffs, event staffs, and so on. The relay devices are also added with our proposed method by these staffs.

The rest of this paper is organized as follows. Section 2 explains related work about the cascading failure in D2D-based SNS. Section 3 describes our formulated optimization problem for relay device placement. Section 4 proposes an autonomous relay device placement algorithm. Section 5 shows some numerical examples, and finally we conclude this paper in Sect. 6.

2. Related Work

In this section, we explain a cascading failure for the data transmission in D2D-based SNS and a performance metric for the cascading failure as related work. Moreover, we explain our previous work about D2D-based SNS.

2.1 Data Transmission in D2D-Based SNS

In D2D-based SNS, a data transmission is only performed directly between two users who have a friendship in SNS via multiple mobile devices with the D2D communication. Figure 1 shows a data transmission from user $u_1$ to $u_6$ in SNS over a social network and a physical network, where the $i$th user $u_i$ has a mobile device $d_i$. Over the social network, the data is not transmitted in two or more hops between any two users because there is no friendship among those users.

On the other hand, in the physical network, the data is transmitted from $d_1$ to $d_6$ in three hops via $d_2$ and $d_3$. Note that the data can be transmitted to all adjacent devices within an D2D communication area over the physical network. Thus, the message transmission in D2D-based SNS corresponds to the combination of relationship in one hop over the social network and message transmission in multiple hops over the physical network.

2.2 Cascading Failure

When a failure occurs in D2D-based SNS, cascading failure occurs [17]. In the cascading failure, many users (mobile devices) are isolated in and removed from the social (physical) network because they cannot communicate with other users (mobile devices).

Figure 3 shows how the cascading failure occurs in D2D-based SNS. In this figure, at first, a failure occurs at mobile device $d_2$ in the physical network, and then all links of $d_2$ are removed from the physical network and all links of $u_3$ are removed from the social network (see Fig. 3 (a)). As a result, as shown in Fig. 3 (b), the physical network is divided into two networks. Then, a link between $u_1$ and $u_2$ and a link between $u_1$ and $u_6$ are removed from the social network because they cannot communicate with the D2D communication (see Fig. 3 (b)). Moreover, as shown in Fig. 3 (c), all links of $d_2$ are removed from the physical network because $u_2$ was isolated in the social network. This is because $u_2$ leaves the D2D-based SNS without motivation to participate in the D2D-based SNS. Finally, most users are isolated and can not communicate each other (see Fig. 3 (d)).

Thus, the cascading failure occurs easily due to the dependencies between the social network and the physical network [17], [23], [24]. This degrades the performance of data transmission significantly in D2D-based SNS.

2.3 Node Resilience

In order to evaluate the robustness of D2D-based SNS, it is...
important to evaluate the impact of cascading failure on the data transmission. In [22], a performance metric called node resilience $\psi$ was proposed to evaluate the cascading failure. In the following, we introduce the node resilience simply, and please see the detailed explanation in [22].

The node resilience denotes the fraction of nodes that are not affected by the cascading failure during those communication time. Now, a mobile device with one or more links in the physical network is called residual device, and the fraction of residual devices at time $t$ is denoted as $n'(t)$. Similarly, a user with one or more links in the social network is called residual user, and the fraction of residual users at time $t$ is represented as $u'(t)$. Let $n(t)$ and $u(t)$ be the fraction of residual devices and the fraction of residual users in the largest component of the physical network and the social network, respectively. The largest components of the physical network (the social network) is an induced subgraph with the largest number of mobile devices (users) where any two mobile devices (users) are connected to each other by links [22]. Here, $n'(t)$ and $u'(t)$ are calculated as follows [17].

$$n'(t) = (1 - p_f)g_f(u'(t - 1)),$$  
(1)

$$u'(t) = (1 - p_f)g_f(n'(t - 1)).$$  
(2)

In these equations, $g_f(n'(t))$ is the fraction of devices that are included in the largest component of the physical network and $g_f(u'(t))$ is the fraction of users that are included in the largest component of the social network. Moreover, $p_f$ denotes the fraction of nodes where the initial failure occurs. Note that it is assumed that the initial failure occurs at randomly selected nodes.

With these equations, the fraction $u_{\infty}$ of users that are never affected by the cascading failure is given as

$$u_{\infty} = u'g_f(u'),$$  
(3)

where $u'$ can be derived in a case of $u'(t) = u'(t - 1) = u'$. Moreover, let $1/\lambda$ be the average communication time of each user, and let $T_{\text{max}}$ be the maximum isolation time that is the largest time until a user or a device is isolated by the cascading failure.

When the communication time follows an exponential distribution, the upper bound and the lower bound of node resilience $\psi$ are calculated as follows

$$\psi \geq u_{\infty} + (1 - p_f - u_{\infty})(1 - e^{-\lambda t})$$  
(4)

$$\psi \leq (1 - p_f - u_{\infty})e^{-\lambda T_{\text{max}}} - p_f.$$  
(5)

The second term in (4) means the minimum fraction of nodes whose communication has already been completed before the cascading failure. Moreover, the second term of (5) denotes the maximum fraction of nodes whose communication fails due to the cascading failure expect for the initial failure. When $\psi$ is large (small), a large (small) number of nodes can complete those communication before the cascading failure. Therefore, in order to avoid the cascading failure, the node resilience $\psi$ should be large.

2.4 Autonomous Decentralized Friendship Management for Improving Node Resilience

In this subsection, we explain our proposed autonomous decentralized friendship management for avoiding the cascading failure in D2D-based SNS [20], [21]. In this management, some virtual temporal friendships are created between two users to increase the number of links in the social network. This is because the node resilience $\psi$ increases as the number of links in the social network and the physical network.

In this method, at first, an optimization problem was formulated for the social network to increase the upper bound of $\psi$. Now, a new virtual temporal friendship $a_{ij}'$ in the social network as

$$a_{ij}' = \begin{cases} 
1, & \text{if a new link is added between user } u_i \\
0, & \text{otherwise.} 
\end{cases}$$  
(6)

Moreover, let $A_{\text{max}}$ be the maximum number of virtual temporal friendships that can be added in the social network. The following optimization problem is formulated.

$$\max_{a_{ij}'} 1 - (1 - p_f - u_{\infty})e^{-\lambda T_{\text{max}}} - p_f, \quad \text{subject to:}$$  
(7)

$$\sum_i \sum_j a_{ij}' \leq A_{\text{max}}.$$  
(8)

In this problem, (7) shows that the upper bound of node resilience can be maximized. The maximum number of added
virtual temporal friendships is restricted in (8). When a larger number of virtual temporal friendship would like to be added, a larger value should be set to $A_{\text{max}}$. However, it takes a long time to add the virtual temporal friendships.

In [20], [21], we investigated the impact of virtual temporal friendships on the node resilience. Based on the above optimization problem and the investigation, a heuristic algorithm for the optimization problem was proposed as the autonomous decentralized friendship management. It was shown that the autonomous decentralized friendship management can increase the node resilience without the complete information on network topology.

3. Optimization Problem of Relay Device Placement

In this section, for relay node placement, we formulate an optimization problem for maximizing the node resilience and investigate the impact of relay device placement on the node resilience. A relay device such as an UAV and a base station can be used only for relaying data and those are not SNS users. Therefore, the relay device is included in the physical network but is not included in the social network.

In the following, we consider a case where $N$ users with a mobile device have participated in D2D-based SNS. Here, the $i$th ($i = 1, \ldots, N$) user is denoted as $u_i$ in the social network and the $i$th device is denoted as $d_i$. Now, $(x_i, y_i)$ represents the position of mobile device $d_i$ in the physical network. The connectivity between two users and the connectivity between two mobile devices are denoted with $z_{ij}$ and $z_{ij}^{d}$ as follows, respectively.

\[
 z_{ij} = \begin{cases} 
 1, & \text{there is a friendship between } u_i \text{ and } u_j; \\
 0, & \text{otherwise}. 
\end{cases} 
\]  

\[
 z_{ij}^{d} = \begin{cases} 
 1, & \text{if } d_i \text{ and } d_j \text{ can be communicated directly with D2D communication}; \\
 0, & \text{otherwise}. 
\end{cases} 
\]  

In the following, we formulate an optimization problem for the physical network to improve the upper bound of node resilience, which is given by (5). Let $N_{\text{relay}}^{\text{max}}$ be the maximum number of relay devices that can be placed in the physical network. Here, we define a variable for the $i$th relay device $d_i^{\text{relay}}$ as

\[
 d_i^{\text{relay}} = \begin{cases} 
 1, & \text{if a position of } d_i^{\text{relay}} \text{ was determined}; \\
 0, & \text{otherwise}. 
\end{cases} 
\]  

In addition, when $d_i^{\text{relay}}$ has one or more links in the physical network, $c_i^{\text{relay}}$ is equal to one as follows.

\[
 c_i^{\text{relay}} = \begin{cases} 
 1, & \text{if } d_i^{\text{relay}} \text{ has some links}; \\
 0, & \text{otherwise}. 
\end{cases} 
\]  

Note that links for a relay device $d_i^{\text{relay}}$ is defined as (10) by replacing $d_j$ with $d_i^{\text{relay}}$, and $(x_i^{\text{relay}}, y_i^{\text{relay}})$ represents the position of $d_i^{\text{relay}}$.

Relay devices are assumed to have no initial failure, and hence the fraction $p_f'$ of devices where the initial failure occurs is defined as

\[
 p_f' = \frac{N \times p_f}{N + N_{\text{relay}}^{\text{max}}} 
\]  

In order to manage positions of relay devices appropriately in the physical network, the following optimization problem is formulated.

\[
 \max_{s_i^{\text{relay}}, y_i^{\text{relay}}} 1 - (1 - p_f' - u_{\infty})e^{-\lambda T_{\text{max}} - p_f'}, 
\]  

subject to:

\[
 \sum_i d_i^{\text{relay}} \leq N_{\text{relay}}^{\text{max}}, 
\]  

\[
 c_i^{\text{relay}} = 1, \ \forall i. 
\]  

In this optimization problem, (14) shows that the upper bound of node resilience can be maximized. The maximum number of added relay devices is restricted in (15), and each relay device has one or more links according to (16). Note that (14) is utilized to improve the node resilience in the physical network but (7) cannot be used because decision variables are different.

Figure 4 shows how a topology of the physical network changes by placing a relay device when $N_{\text{relay}}^{\text{max}}$ is equal to one. In this figure, only a relay device $d_1^{\text{relay}}$ is added to a position $(x_1, y_1)$ according to the solution of the optimization problem (see Fig. 4(a)) and three new links are added in the physical network (see Fig. 4(b)).

Now, we consider a D2D-based SNS with 100 users, and each user with a mobile device is randomly placed on a plane whose size is $120 \times 120$. The communication range of each mobile device and relay device is 10 and a link is set between two devices within the communication range in the physical network. On the other hand, a friendship is established between two users so that the user relationships in SNS are created to have similar characteristics to scale-free relationships. Here, in the social network, the maximum node degree is eight. For each user, we assume that the communication time follows an exponential distribution with $1/\lambda = 200$. In the following, a high-degree node is
defined as a node whose degree is equal to or larger than six because the maximum node degree is eight. Moreover, \( h_p(k) \) is defined as the fraction of high-degree nodes in the physical network.

Figure 5 shows how node resilience \( \psi \) changes by adding a relay device in the physical network. This figure shows the node resilience is increased with the relay device if the vertical axis (ratio) is larger than one. From this figure, we find that the ratio increases significantly if a relay device is placed closer to nodes in high-density device group regardless of \( p_f \). This is because high-degree devices can help other devices to avoid the cascading failure more easily by utilizing the relay device. From this result, it is clear that relay device should be added close to devices whose degree is large to improve the node resilience.

4. Autonomous Relay Device Placement Algorithm

In this section, we propose an autonomous relay device placement algorithm for cascading failure in D2D-based SNS. In D2D-based SNS, there is no administrator and controller and each device is not connected to a complete communication network such as the Internet. Moreover, the complete topology information is not available in each mobile device. Therefore, it is not easy to use the optimization problem (14) in D2D-based SNS. Our proposed method obtains a solution of the optimization problem (14) approximately with the partial information that is obtained from adjacent devices. Then, some relay devices are added in the physical network based on the approximate optimal solution, which is formulated in Sect. 3. That is, in our method, the positions of added relay nodes are decided as is the case with the optimization function (14). Note that, the proposed method is performed without the complete topology information in an autonomous distributed manner, and hence relay devices are added near mobile devices.

Our proposed method consists of two algorithms; Algorithm 1 and Algorithm 2. The execution of Algorithm 1 is triggered by a mobile device that is selected at random by an application or selected by network management staffs, an event staff, and so on. Then, as shown in Fig. 6, the target device \( d_i \) that executes these algorithms is changed while forwarding \( N_{\text{max}}^{\text{relay}} \) and the obtained result \( \sum_i a_i^{\text{relay}} \). The obtained outputs are forwarded to network management staffs, an event staff, and so on, and the relay devices are added by these staffs. Obtained outputs can be forwarded to some mobile devices of some staffs before the final outputs are obtained if there are some mobile devices for the staff unlike Fig. 6.

In Algorithm 1, In this algorithm, \( N_{\text{max}}^{\text{relay}} \) relay devices are always added in the physical network because the maximum number of relay devices increases the node resilience. From line 4 to line 20, a relay device \( d_j^{\text{relay}} \) is placed to a position \((x_j^{\text{relay}}, y_j^{\text{relay}})\) close to \( d_j \) until the number \( \sum_j a_j^{\text{relay}} \) of

---

**Algorithm 1: Main algorithm**

1. **Input**: \( T_h, L \)
2. **Output**: \( \sum_i a_i^{\text{relay}} \cdot y_j^{\text{relay}} \)
3. A device \( d_i \) is selected as target device \( d_t \)
4. while \( \sum_i a_i^{\text{relay}} \leq N_{\text{max}}^{\text{relay}} \)
5. if \( r_i = 0 \)
   6. \( r_i = 1 \)
   7. else
   8. \( T_h = T_h - L \)
   9. \( C = 0, D = 0 \)
10. for \( j = 1 \) to \( N \)
11. if \( C_j \) \( = 1 \)
12. \( C = C + 1 \)
13. if \( R_j \geq T_h \)
14. \( D = D + 1 \)
15. \( D_i = D/C \)
16. if \( D_i \geq T_h \)
17. \( a_i^{\text{relay}} = 1 \)
18. \( d_i \) randomly selects \( d_j \) as the next target device among devices with \( \sum_j a_j^{\text{relay}} = 1 \)
19. \( R_i = R/C \)
20. return \( R_i \)

**Algorithm 2: DRC function**

1. **Input**: \( d_i \)
2. **Output**: \( R_i \)
3. \( C = 0, R = 0 \)
4. for \( j = 1 \) to \( N \)
5. if \( C_j \) \( = 1 \)
6. \( C = C + 1 \)
7. if \( p_s(i) \leq p_s(j) \)
8. \( R = R + 1 \)
9. else if \( C_j \) \( = 1 \)
10. \( C = C + 1 \)
11. if \( p_s(i) \leq p_s(j) \)
12. \( R = R + 1 \)
13. \( R_i = R/C \)
14. return \( R_i \)
relay devices becomes $N_{\text{rela}}^{\text{relay}}$. This is because this method has to be performed without the complete topology information in an autonomous distributed management. Here, $r_i$ denotes whether $d_i$ has already been selected as target device $d_i$ or not as follows.

$$r_i = \begin{cases} 1, & \text{if } d_i \text{ has already been selected} \\ 0, & \text{otherwise.} \end{cases}$$  \tag{17}

At line 5, $r_i$ is set to one if $d_i$ is selected as $d_i$ for the first time.

In this algorithm, the node density $D_i$ of $d_i$ is calculated at line 9 to 15. Now, let $C$ be the number of devices whose degree ranking is compared with $T_h$, and let $D$ be the number of devices whose degree ranking is higher than $T_h$. A degree ranking is set to each node in order to find nodes in high-density device group. Degree ranking shows the descending order of node in terms of node degree. At line 12, for each adjacent node $d_j$ of $d_i$, its degree ranking $R_j$ is derived within a function DRC. Next, at lines 13 and 14, the degree ranking $R_j$ and $T_h$ are compared, and the number of devices whose degree ranking is higher than $T_h$ is counted as $D$. Finally, the node density $D_i$ of $d_i$ is derived from $C$ and $D$ at line 15.

Then, each node is categorized into the high-density device group or the low-density device group according to $T_h$ at line 16. When $D_i$ is equal to or larger than $T_h$, $d_i$ is classified into the high-density device group. In this case, a relay node is placed close to $d_i$ as shown in Fig. 5, and $d_i$ is set to 1. Otherwise, a relay device is not placed close to $d_i$ because $d_i$ is in the low-density device group.

After checking whether the relay device is placed or not, $d_i$ randomly selects $d_j$ with $z_{ij}^{dl} = 1$ as a new target device at line 19. Moreover, threshold $T_h$ is updated at lines 6 and 7. When the mobile device $d_i$ whose $r_i$ is equal to 1 is selected as the target device, there are no devices in the high-density device group. Therefore, $T_h$ is decreased by $L$ to increase the number of devices in the high-density device group. In Fig. 6, a central device is selected twice after black dashed arrow 1 and black dashed arrow 4. Therefore, $T_h$ is decreased by $L$ before black dashed arrow 5. As a result, $\sum_i d_i^{\text{relay}}$ can be increased until line 4 is satisfied in Algorithm 1 by decreasing threshold. $N_{\text{rela}}^{\text{relay}}$ relay devices can be added regardless of the initial value of $T_h$.

As explained at line 12 in Algorithm 1, degree ranking $R_j$ is derived within a function DRC. In Algorithm 2 of DRC, let $C$ be the number of devices whose degree ranking is compared with $d_i$, and let $R$ be the number of devices whose degree is higher than the degree of $d_i$. Moreover, $p_d(i)$ and $p_s(i)$ denotes the node degree of $d_i$ and $u_i$ in the physical network and the social network, respectively. Here, in this algorithm, the node degree is compared with an adjacent node in the social node and/or the physical node $(z_{ij}^{dl} = 1 \text{ and/or } z_{ij}^{ds} = 1)$. At lines 5 to 8, the degree of node $d_i$ is compared with adjacent node $d_j$ in the physical network $(z_{ij}^{dl} = 1)$. Then, at lines 9 to 12, the degree of node $d_i$ is compared with adjacent node $d_j$ in the social network $(z_{ij}^{ds} = 1)$. Finally, the degree ranking $R_j$ of $d_i$ is derived from $R$ and $C$ at line 13.

5. Numerical Examples

In this section, we evaluate the performance of autonomous relay device placement algorithm for the D2D-based SNS. In the following, the number $N$ of users is 100, and each user is placed on a plane whose size is $120 \times 120$. The communication range of each mobile device and relay device is 10 because the bluetooth communication range of the mobile device is about 10 m, and a link is set between two devices within the communication range in the physical network. On the other hand, a friendship is established between two users so that the user relationships in SNS are created to have similar characteristics to scale-free relationships. Moreover, the fraction $p_f$ of nodes where the initial failure occurs is equal to 0.3. The initial failure includes not only the device failure but also some cases where mobile devices are turned off or users leave D2D-based SNS. Note that the communication time follows an exponential distribution with rate $\lambda$.

5.1 Impact of the Number of Relay Device on the Upper Bound of Node Resilience

First, we investigate the impact of the number of relay devices on the node resilience. In the following, the average communication time $1/\lambda$ is set to 200 and $L$ is equal to 2 for our proposed algorithm.

Figure 7 shows the impact of the number $N_{\text{rela}}^{\text{relay}}$ of relay devices on the node resilience $\psi$. In this figure, for the performance comparison, we evaluate the GA method where an approximate optimal solution for the optimization problem (14) by using genetic algorithm (GA) that is one of the meta-heuristic algorithms. Therefore, relay devices are placed according to the solution obtained by GA. The GA method can obtain an approximate optimal solution of our optimization problem (14) easily, but the complete information of network topology has to be utilized unlike our proposed method. In the GA method, the calculation finishes when there is enough convergence. We also evaluate the Random method where relay devices are randomly placed on the plane.

From this figure, we find that the node resilience for
the GA method is the largest among the three methods. This is because the GA method can place relay devices according to the approximate optimal solution. On the other hand, the node resilience for the Random method is the smallest although the complete information of network topology is not needed. By using our proposed method, the node resilience can be improved effectively without the complete information of network topology. The difference between the proposed algorithm and the GA method is smaller than that between the proposed method and the Random method. Especially, the node resilience of our proposed algorithm increases as the number of relay devices as is the case with the GA method. As a result, our proposed algorithm is effective to avoid the cascading failure.

5.2 Impact of $\lambda$ on the Upper Bound of Node Resilience

Next, we investigate the impact of $\lambda$ on the node resilience. Figure 8 shows the impact of $\lambda$ on the node resilience $\psi$ in a case of $L = 2$. In this figure, $1/\lambda$ is set to 200, 400, 600, 800, and 1000. From this figure, we can find the node resilience becomes small as $1/\lambda$ increases. This is because the average communication time increases and the impact of cascading failure increases. However, by increasing the number $N_{\text{relay}}^{\text{max}}$ of relay devices with our proposed algorithm, the node resilience becomes large. This results shows the effectiveness of our proposed method for avoiding the cascading failure.

5.3 Impact of $L$ on Node Resilience and the Number of Hops

We also investigate the impact of $L$ for our proposed algorithm on the node resilience and the number of hops for the data transmission. Figure 9 shows the impact of $L$ on the node resilience for our proposed algorithm when $1/\lambda$ is 200. From this figure, we find that $L$ should be set to a smaller value. Comparing a result for $L = 2$ with that for $L = 20$, the upper bound of node resilience for $L = 2$ is larger than that for $L = 20$. Here, $T_h$ is decreased by $L$ in Algorithm 1. When $T_h$ is small, a large number of devices is included in the high-degree device group. As a result, inappropriate devices are more likely to be selected for increasing the node resilience when $L$ is equal to 20. On the other hand, when $L$ is equal to 2, nodes whose node degree is high are more likely to be selected, resulting in the high upper bound of node resilience.

Figure 10 shows the relationship between $L$ and the number of hops for our proposed method. From this figure, we find that the number of hops increases when $L$ becomes small. This is because it takes a long time to find a high degree node.
5.4 Computation Time

Finally, we evaluate the calculation time for determining positions of 10 relay devices in a case where $1/L$ is set to 200 and $L$ is set to two. The calculation time of the Random method, the GA method, and the proposed autonomous decentralized device management method is investigated by using a computer with Microsoft Windows 10 Pro 64-bit, Intel Core i5-3470, and 8 GB memory.

Table 1 shows the calculation time of our proposed algorithm, the GA method, and the random method. As explained before, in the GA method, the calculation finishes when there is enough convergence. From this table, we find that the calculation time for our proposed algorithm is much smaller than that for the GA algorithm, although the calculation time for the RA method is the smallest. Therefore, the proposed method can reduce the impact of cascading failure in a shorter computation time than the GA method. In D2D-based SNS, the topologies of the social network and the physical network tend to change frequently. The solution of the optimization problem is expected to be obtained more quickly. Therefore, the proposed method is more effective than other methods in real environments.

6. Conclusions

In this paper, we proposed an autonomous device placement algorithm in order to avoid the cascading failure in D2D-based SNS. In this method, some relay devices are placed so as to increase the number of links in the physical network. In the proposed method, at first, an optimization problem for minimizing node resilience which is a performance metric in order to place relay devices. Then, we investigate how relay devices should be placed based on some approximate optimal solutions. From this investigation, we propose an autonomous relay device placement in the physical network. In our proposed algorithm, relay devices can be placed without the complete information on network topology. We evaluate the performance of the proposed method with simulation, and investigate the effectiveness of the proposed method. From numerical examples, we found that our proposed algorithm is effective to avoid the cascading failure without the complete information on network topology.

Acknowledgments

This work is supported by SCAT foundation.

References

[1] X. Lin, J.G. Andrews, and A. Ghosh, “Spectrum sharing for device-to-device communication in cellular networks,” IEEE Trans. Wirel. Commun., vol.13, no.12, pp.6727–6740, Dec. 2014.
[2] W. Qu, G. Li, and Y. Zhao, “On the coverage problem in device-to-device relay networks,” IEEE Commun. Lett., vol.23, no.11, pp.2139–2143, Nov. 2019.
[3] C. Xu, J. Feng, Z. Zhou, J. Wu, and C. Perera, “Cross-layer optimization for cooperative content distribution in multihop device-to-device networks,” IEEE Internet Things J., vol.6, no.1, pp.278–287, Feb. 2019.
[4] B. Li, H. Li, X. Li, H. Jiang, W. Tang, and S. Li, “Hybrid multicast and device-to-device communications based on adaptive random network coding,” IEEE Trans. Commun., vol.67, no.3, pp.2071–2083, March 2019.
[5] A. Asheralieva and D. Niyato, “Combining contract theory and Lyapunov optimization for content sharing with edge caching and device-to-device communications,” IEEE/ACM Trans. Netw., vol.28, no.3, pp.1213–1226, June 2020.
[6] N. Anjum, Z. Yang, H. Saki, M. Kiran, and M. Shikh-Bahaei, “Device-to-device (D2D) communication as a bootstrapping system in a wireless cellular network,” IEEE Access, vol.7, pp.6661–6678, Jan. 2019.
[7] A. Jamali, S.M. Safavi Hemami, M. Berenjkoub, H. Saidi, and M. Abedini, “A high throughput device-to-device wireless communication system,” IEICE Trans. Inf. & Syst., vol.E102-D, no.1, pp.124–132, Jan. 2019.
[8] X. Shen, “Device-to-device communication in 5G cellular networks,” IEEE Netw., vol.29, no.2, pp.2–3, March 2015.
[9] A. Asadi, Q. Wang, and V. Mancauso, “A survey on device-to-device communication in cellular networks,” IEEE Commun. Surv. Tutor., vol.16, no.4, pp.1801–1819, April 2014.
[10] J. Obregon, M. Song, and J.-Y. Jung, “InfoFlow: Mining information flow based on user community in social networking services,” IEEE Access, vol.7, pp.48024–48036, April 2019.
[11] P.E.N. Lutu, “Using twitter mentions and a graph database to analyze social network centrality,” Proc. 2019 6th International Conference on Soft Computing & Machine Intelligence (ISCMI), Nov. 2019.
[12] L. Wu, L. Chen, R. Hong, Y. Fu, X. Xie, and M. Wang, “A hierarchical attention model for social contextual image recommendation,” IEEE Trans. Knowl. Data Eng., vol.32, no.10, pp.1854–1867, 2020.
[13] T. Chuman, K. Iida, W. Shirohata, and H. Hino, “Image manipulation specifications on social networking services for encryption-then-compression systems,” IEICE Trans. Inf. & Syst., vol.E102-D, no.1, pp.11–18, Jan. 2019.
[14] Y. Zhang, E. Pan, L. Song, W. Saad, Z. Dawy, and Z. Han, “Social network aware device-to-device communication in wireless networks,” IEEE Trans. Wirel. Commun., vol.14, no.1, pp.177–190, Jan. 2015.
[15] J. Dong, B. Chen, L. Liu, C. Ai, and F. Zhang, “The analysis of influencing factors of information dissemination on cascade size distribution in social networks,” IEEE Access, vol.6, pp.54185–54194, Sept. 2018.
[16] F. Malandrino, C. Casetti, and C.-F. Chiasserini, “Toward D2D-enhanced heterogeneous networks,” IEEE Commun. Mag., vol.52, no.11, pp.94–100, Nov. 2014.
[17] S.N. Buldyrev, R. Parshani, G. Paul, H.E. Stanley, and S. Havlin, “Catastrophic cascade of failures in interdependent networks,” Nature, vol.464, no.7291, pp.1025–1028, April 2010.
[18] P.-D. Yu, C.W. Tan, and H.-L. Fu, “Averting cascading failures in networked infrastructures: Post-constrained graph algorithms,” IEEE J. Sel. Top. Signal Process., vol.12, no.4, pp.733–748, Aug. 2018.
[19] M. Korkali, J.G. Veneman, B.F. Tivnan, J.P. Bagrow, and P.D.H. Hines, “Reducing cascading failure risk by increasing infrastructure network interdependence,” Scientific Reports, vol.7, no.44499, June 2018.
YOKOI and TACHIBANA: AUTONOMOUS RELAY DEVICE PLACEMENT ALGORITHM IN D2D-BASED SOCIAL NETWORKING SERVICE

Proc. 2018 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-TW), Taipei, Taiwan, May 2018.

[21] H. Yokoi and T. Tachibana “Virtual temporal friendship creation: Autonomous decentralized friendship management for improving robustness in D2D-based social networking service,” Under submission to IEEE Access.

[22] S.A. Pambudi, W. Wang, and C. Wang, “On the resilience of D2D-based social networking service against random failures,” Proc. IEEE GLOBECOM 2016, Dec. 2016.

[23] X. Huang, J. Gao, S.V. Buldyrev, S. Havlin, and H.E. Stanley, “Robustness of interdependent networks under targeted attack,” Phys. Rev. E, vol.83, no.6, June 2011.

[24] R. Parshani, S.V. Buldyrev, and S. Havlin, “Interdependent networks: Reducing the coupling strength leads to a change from a first to second order percolation transition,” Phys. Rev. Lett., vol.105, no.4, July 2010.

Hanami Yokoi received the B. Eng. degree from the Faculty of Engineering, University of Fukui, Japan, in 2018. She received the M. Eng. degree from the Graduate School of Engineering, University of Fukui, Japan, in 2020. She joined NTT Network Technology Laboratories in 2020. She is a member of the Institute of Electronics, Information and Communication Engineers (IEICE).

Takuji Tachibana received the B. Eng. degree from the Department of Systems Engineering, Nagoya Institute of Technology, Japan, in 2000. He received the M. Eng. and Dr. Eng. degrees from the Department of Information Systems, Graduate School of Information Science, Nara Institute of Science and Technology, Japan, in 2001 and 2004, respectively. From 2004 to 2006, he was an expert researcher in the Information and Network Systems Department, National Institute of Information and Communications Technology, Japan. From 2006 to 2011, he was an assistant professor at the Department of Information Systems, Graduate School of Information Science, Nara Institute of Science and Technology, Japan. From 2011 to 2019, he was an associate professor at Graduate School of Engineering, University of Fukui, Japan. Since 2019, he has been a professor at Graduate School of Engineering, University of Fukui, Japan. His research interests include network architectures in optical networking and performance analysis of computer and communication systems. He is a member of the IEEE, The Institute of Electronics, Information and Communication Engineers (IEICE), and the Operations Research Society of Japan.