Chapter 1
Routing in Post-Disaster Scenarios

Christian Esposito, Zhongliang Zhao, Gianluca Rizzo, Florin Pop, Elena Apostol, Cătălin Leordeanu and Stefan Preda

Abstract The current networks should provide disaster-resilience by coping with the possible failures and misbehaviors caused by massive natural or man-made disasters. This is demanding to keep a suitable level of Quality-of-Service after a disaster and to support the possible evacuation, rescue, assessment, and rescue operations within the affected area. Multiple possible methods and solutions can be put in place in a proactive and/or reactive manner to offer the demanding resilience degree. Among them, a proper routing algorithm can contribute to circumventing network elements damaged by the disaster or applying for spatial/temporal redundancy to guarantee effective communications. This chapter aims at presenting the main routing solutions to offer disaster-resilience communications and some related methods.

Christian Esposito
Department of Electrical Engineering and Information Technologies (DIETI), University of Napoli "Federico II", Via Claudio, 21, 80125 Napoli - Italy e-mail: christian.esposito@unina.it

Zhongliang Zhao
Institute of Computer Science, Universität Bern, Neubrückstrasse 10, CH-3012 Bern - Switzerland e-mail: zhongliang.zhao@inf.unibe.ch

Florin Pop, Elena Apostol, Cătălin Leordeanu and Stefan Preda
Faculty of Automatic Control and Computers, University Politehnica of Bucharest (UPB), PRECIS-601/EG403 Splaiul Independentei nr. 313, sector 6, Bucharest - Romania e-mail: {florin.pop,elena.apostol,catalin.leordeanu,stefan.preda}@cs.pub.ro

Gianluca Rizzo
Institute of Information Systems, University of Applied Sciences of Western Switzerland (HES-SO), Techno-Pôle 3, 3960 Sierre - Switzerland e-mail: gianluca.rizzo@hevs.ch

1.1 Introduction

Any communication infrastructure is susceptible to many unforeseen events, such as environmental disasters, power failures, hardware failures, etc. Such disasters can affect the functionality and performance of the communication network, some more than others. Therefore, network fault tolerance and the way that communication is resumed after the disaster takes place are subjects that need to be taken into account. It is crucial to achieving resilience in post-disaster scenarios where the overall network infrastructure has been partially compromised (with physical devices being destroyed or wireless signals attenuated more than usual), or the high increase of the traffic is causing congestion phenomena. As a concrete example of the importance of communication efficiency in post-disaster scenarios, during the worst fire cases in Portuguese history in June 2017, a large number of users were cut off from using fixed-line or cellular telephone service, giving rise to serious traffic congestion. The communications among the rescue teams had been severely affected, obstructing the execution of operations. Bad communications are felt to have increased the overall number of causalities in terms of human life losses. Therefore proper means to achieve resiliency within the communication infrastructure is demanding.

Resilience and security need to be taken care of when designing a communication infrastructure, a middleware or a network. Also, these requirements have to include fault-tolerance mechanisms for the network to withstand a wide variety of challenges. Network resilience is defined as the capability of behaving at an acceptable level of performance and availability, even when faults may occur [30]. Fault tolerance is the property of a system to maintain a certain level of service when confronted with events of failure within one or more of its components. In the best case, the quality of the service should not degrade at all, but in practice, this is usually not possible. The operating quality should be proportional to the severity of the encountered failure. In computer networks, a fault-tolerant design allows a system to continue its operation, possibly with the reduced service level, rather than failing when some of its components cease working. Usually, after encountering a failure, the reduction in service is manifested by a reduction in throughput, but the system as a whole is not stopped.

This chapter aim at presenting the state of the art and future research directions on the provision of disaster-resilient communications by exploiting solutions at the network level and leveraging proper routing solutions. The chapter is structured as follows. Section 1.2 describes the issues to deal with disasters in case of event-based communications infrastructures, which are a key building block for emergency networks and other similar ICT infrastructures. Section 1.3 describes some generic resilience routing for unicast as well as multicast communications. The Section 1.5 presents the possibility of having a network of the unmanned aerial vehicle (UAV) to support network resilience and the routing issues, while the Section 1.6 describes the problem of user association when multiple communication means are available. Some final remarks end the chapter in the last section.

1.2 Resilient Event Notification

Event notification is an extremely popular communication pattern in the current information systems, such as Wireless Sensor Networks, Internet of Things, critical control systems and services and so on. It is a communication pattern [17] where a set of applications generates messages describing occurring events (such as a given stat being reached or a measure returning a certain value), while other applications are interested to receive such messages. Message delivery is conducted in a data-centric manner: based on the content of the messages the interesting destinations are identified and the messages delivered to them. Subscriptions express the interest of receiving a type or class of events, through regular expressions on the message content, or topics associated with them. It consists of a distributed implementation of the well-known observer design pattern. Publish/subscribe services are middleware solutions supporting event notifications, where the publishers generate notifications and subscribers consume them. These entities can be directly connected, or a set of special entities, called brokers, manage the routing of messages and the establishing of connections based on active subscriptions. In fact, on top of the physical computing machines, routers and links, the publishers, subscribers, and brokers build an overlay, illustrated in Fig. 1.1, aiming at distributing notifications among the interested subscribers from the originating publishers. The deployment is typically promiscuous so a physical node can host a single publisher or subscriber, but it can also have all of them, as in the figure. In the last case, a possible failure of such an element can have severe consequences to the overall connectivity of the overlay.

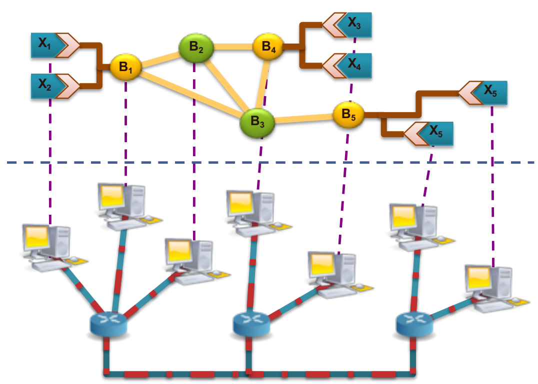

Fig. 1.1 Abstraction of notification overlay

Due to their key role in critical applications [5, 13], such as rescue management for exchanging emergency-related information [6, 4] or support to the multi-organization rescue management operations, a resilient event notification is demanding, but the current solutions available at the marked products fail to properly address such a requirement.

1.2.1 Fault Model

Typically, the fault model of a publish/subscribe service is composed of network misbehaviors and temporary/permanent unavailability of certain networking elements [14], as depicted in Fig. 1.2. First, we have the following fault cases: *Message Losses*: links behave as a fairy-loss channel, *i.e.*, messages in transit through a link may be lost; *Ordering*: messages are received in a different order than the one of their sending; *Corruption*: messages are received corrupted; *Delay*: a message is delivered later than expected; *Congestion*: a link/router is overloaded, suddenly causing several anomalies; and *Partitioning*: network may get fragmented into several disconnected parts. Second, possible unavailability may consist in *Link crash*: a certain degree of persistent message losses over a time frame is experienced along a link, such a complete message loss is not necessarily permanent but may dynamically appear and disappear over the time, and *Node crash*: nodes may become unavailable due to hardware/software failures. Such a model is typically simplified by considering the following considerations. The communication reliability and the delivery success rate may be affected by the delay and corruption faults, since they can be considered at the application level as losses (specifying proper timeouts on the delivery time, while corrupted messages can be detected by leveraging on Cyclic Redundancy Check (CRC) and discarded accordingly). Furthermore, partitioning is treated as a special case of link crashes (in fact, all the approaches handling network partitions also treat link crashes). On the contrary, many efforts have been spent to address crashes, leading to an inconsistent topology among the nodes within the publish/subscribe service. So, putting everything together, a failure model suitable for event notification services comprises only crashes and losses.

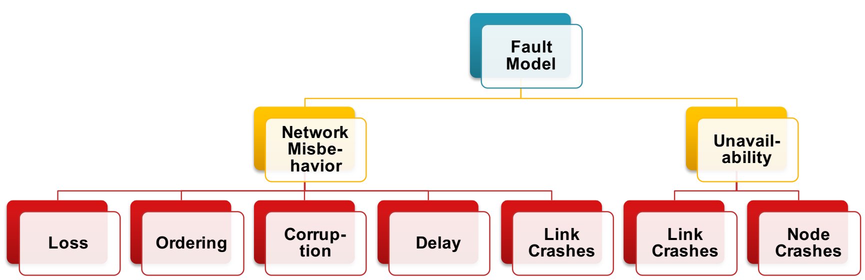

Fig. 1.2 Overview of the fault model for event services

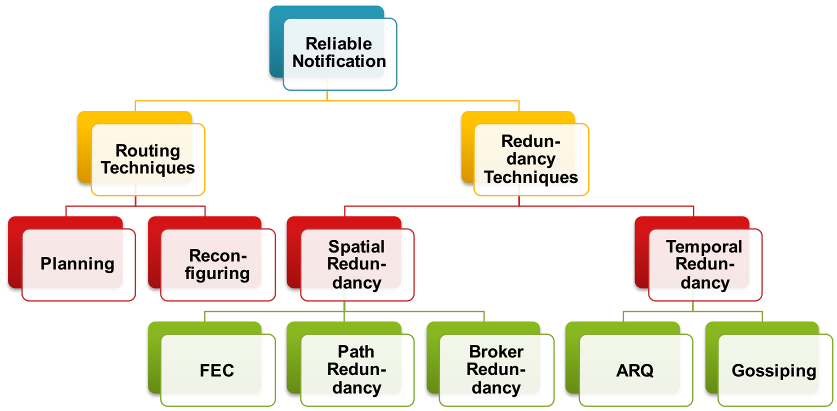

Fig. 1.3 Taxonomy of fault-tolerant techniques for event notification services

1.2.2 Literature on fault-tolerance in event notification

Despite these faults are experienced in the current networks, with a higher probability in the wireless networks, a disaster can considerably increase their occurrence by physically destroying the networking devices, or disturbing the RF signals with a higher attenuation or interference. Therefore, it is important to adopt proper means to deal with such faults. A classification of the current approaches adopted in literature to implement reliable event notification [14] is illustrated in Fig. 1.3. We can distinguish among two classes of techniques that can be roughly described as follows: Routing techniques aims at proactively providing fault-tolerance by choosing the most resilient path among the available ones. The *planning techniques* estimate the reliability level of several paths from a source to a destination, and pick the one that exposes the highest reliability. This approach realizes the so-called *Fault-Avoidance* by reducing the probability of occurrence of losses, but it is not able to react if a notification loss happens. Therefore, *Reconfigurations* can be commenced to recover the lost connectivity within the forwarding tree or mesh in case of any node/link crash's occurrences. This can be done by having the entries of the routing tables being valid within a limited "time to live", periodically renewed otherwise they are deleted. This approach aims to guarantee consistent connectivity for the system, *i.e.*, all the processes are connected. Therefore, its scope is to provide *Fault-Handling* capabilities in the system, *i.e.*, it tries to prevent the fault to occur again. But, it does not cope with losses, so not all the subscribers will receive the messages of interest if network omissions may occur, which violates the agreement.

The second class of techniques aims to offer redundancy to recover from losses (and coping with temporary crashes). Redundancy can be applied in space or time and allows us to realize fault-tolerance by *Compensation*, *i.e.*, lost data is compensated with enough redundant information to enable the loss to be masked and the data delivery is guaranteed. Temporal redundancy aims at replicating notifications over time using retransmissions, while spatial one consists of sending additional data along with the ones to be delivered to reconstruct the original message if losses occur.

Centralised Retransmissions (ARQ) [27] is a way to achieve reliable event dissemination by having publishers/brokers storing outgoing notifications so that subscribers can request retransmissions when losses are somehow detected (*e.g.*, employing timeouts or sequence numbers). Despite the ease of implementation, this approach is not optimal:

- it can deal with only message losses;
- the time to recover dropped notifications is unpredictable since depends on the number of retransmissions and, consequently to the experiences loss pattern;
- there are no guarantees to achieve agreement, due to a link crash that disconnects partially/completely publishers from a set of the interested subscribers or the possibility that a publisher may crash before all the subscribers have recovered lost notifications;
- notifications can be received twice due to false-negative detection of dropped notifications

A more distributed retransmission scheme is based on *Epidemic Algorithms* [18], where the leaf nodes of the forwarding tree/mesh can commence retransmissions. Each participant to the event service exchanges at a random time its history of the received notifications with a randomly-chosen process among the ones constituting the system. Subsequently, inconsistencies, *i.e.*, message drops, are detected by comparison and corrected by requesting retransmissions. A different push gossip consists in having the nodes to send directly a subset of the least received notifications, to avoid the request operation. Gossip provides more guarantees than using ARQ since it can also recover notifications lost due to crashed nodes and/or links, it avoids having to receive the same message twice. Moreover, the agreement is reached in a probabilistic manner (*i.e.*, it is possible to compute a given probability of agreement reachability, based on the algorithm parameters and network/node conditions), due to the randomness behavior of the algorithm. However, despite reducing the overall recovery time, the notification latency is still unpredictable and strongly depending on the network conditions.

Spatial redundancy aims at replicating data at different parts of the event notification services. The widely-known technique is *Forward Error Correction (FEC)*, where redundant data obtained by encoding the content of the notification to be exchanged are used to recover from losses, instead of using retransmissions. This solution enforces timeliness since the worst case latency is predictable; however, it presents several drawbacks. The main one is that agreement is reached depending on the redundancy used at the sender's side. A receiver can obtain the message only if the sender has delivered more additional data than the packets lost by the network. Otherwise, the dropped notifications are more than the ones the receiver can reconstruct and the message is considered lost. Choosing the right redundancy is a very hard task on the Internet since losses are highly variable over time. Losses are the only fault that can be tolerated with this approach. *Broker Replication* achieves fault-tolerance by having multiple replicas of the brokers in the notification service [39]. The state of a broker is replicated to its neighbors so in case of a broker failure it can be easily substituted without losing subscription consistency into the system. This solution is tailored only for node failures, and link crashes may involve that some subscribers are not reachable and so there may be no agreement in the system. *Path Redundancy* consists of establishing redundant paths among the nodes of the system, and can be also combined with broker redundancy. A notification is sent through multiple paths, and only the first-received replica of the notification is delivered to the application. This solution appears to be tailored to link failures but can also cope with node crashes thereby circumventing the failed node. All the abovementioned requirements for the reliable publish/subscribe service plus timeliness are met under one condition: the diversity of all the found paths to a destination is guaranteed, *i.e.*, no more than one path is compromised by a certain failure, as they do not share any routing element [21].

1.2.3 Novel approaches

1.2.3.1 Improving the gossiping efficiency

There is no winning solution among the ones available within the literature. In particular, retransmission-based schemes provide high resiliency at the expense of poor performance, while FEC schemes offer the opposite. Fig. 1.4 summarizes such an aspect. Specifically, on the left side of the figure, the chart presents the high fluctuation in the achievable latency when ARQ is used, compared with the more limited variability of such a measure when only FEC is used. By combining both schemes (such sending coded messages along side the plain ones when triggering a retransmission), the perceived performance is more stable, with a lower variability around the median, and the number and height of the spikes in the measured latency are minimized (as in the right side of Fig. 1.4. Starting from such a shred of evidence, FEC and gossiping have been integrated as described in [15], as shown in Figure 1.5. When a node has to gossip with another one, in case of a push mode, not only the latest received messages are sent but also encoded data to tolerate their losses. Similarly, in the pull mode, the retransmissions also contain code data to fast reconstruct lost data without requesting any additional retransmissions. The used coding function is the simple Random Linear Coding (RLC), where the datagrams composing a notification are xored by considering as coefficients a set of randomly selected numbers.

The finding of such a work is described in Figures 1.6 and 1.7, which the success rate, such as the percentage of published notifications being successfully received by all the interested subscribers, shows a considerable improved when the coding is introduced, both in the push and pull mode (the figure shows only the measure in the first case due to page limits). It is remarkable to notice that FEC applied during the first phase of the gossip is more effective, as it allows to achieve a higher

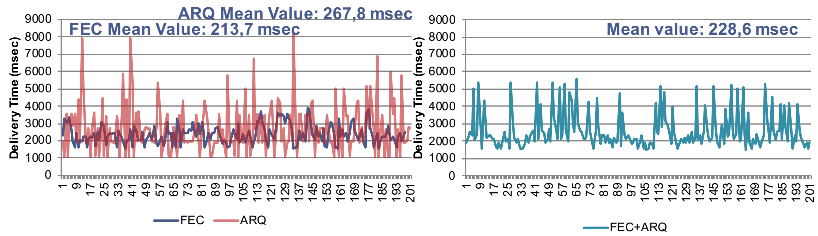

Fig. 1.4 Performance of ARQ and FEC in Isolation and when combined.

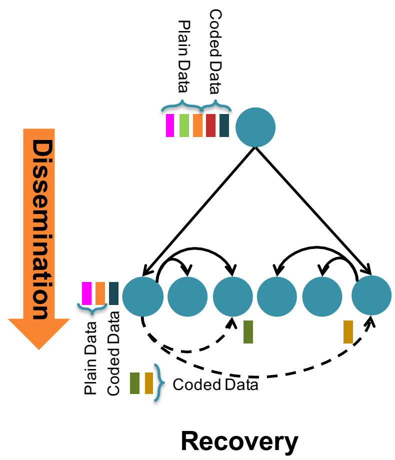

Fig. 1.5 Gossiping combined with FEC.

improvement. This is because FEC reduces the number of needed retransmissions to tolerate losses. This allows to reduce the performance costs of the approach, as depicted in Fig. 1.7 where the latency is lower than the case with no coding, and the overhead (*i.e.*, the number of exchanges packets during the gossip run) is similarly reduced than the case with no coding.

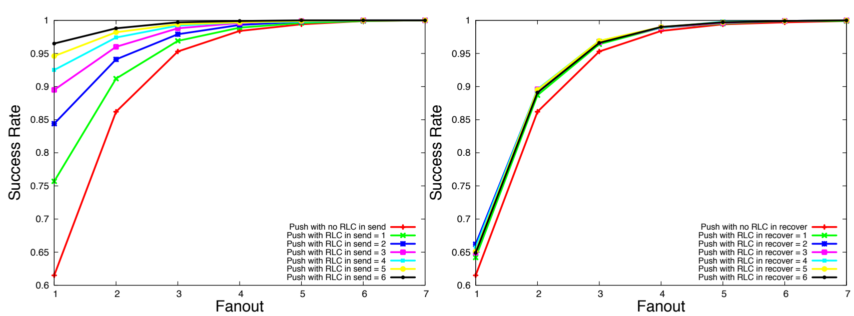

Fig. 1.6 Success rate as a measure of resilience when gossiping and FEC are integrated.

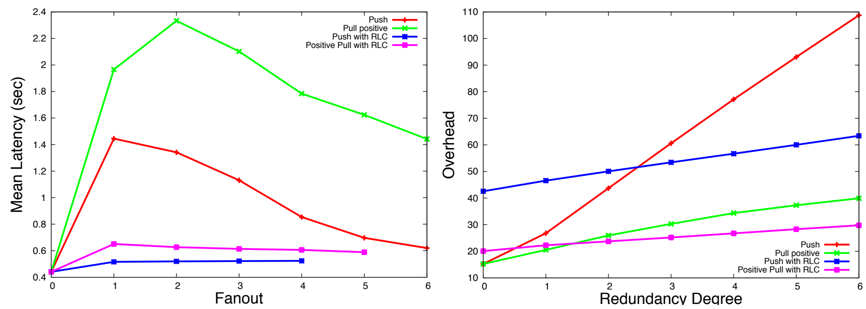

Fig. 1.7 Latency and overhead as measures of performance when gossiping and FEC are integrated.

The authors in [10] noticed that there is still space for improvements for gossiping even if coding is not applied. Fig. 1.8 depicts a utility function for the two kinds of gossip, *i.e.*, a measure of how useful to recover from a loss a gossip message has been for a subscriber. The figure tells us that several messages are a waste since they carry on data related to notifications already received by the destination, and this is not resolved even with coding but depends on the random nature of gossip in selecting with which a node has to gossip. Such unnecessary traffic can be troublesome in case of disasters, where the network has a reduced capacity due to the experienced physical damages or the spike in the incoming communications from users willing to receive updates or sending requests for help.

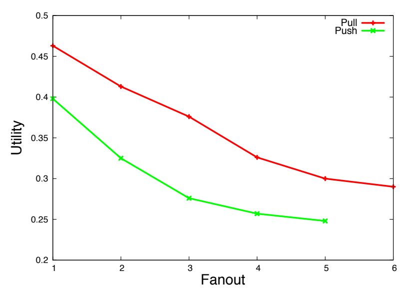

Fig. 1.8 Utility of the gossip messages.

If such an utility is formalized as follows:

$$U = \frac{1}{N}\sum_{i=0}^{N} U_i = \frac{1}{N}\sum_{i=0}^{N} \frac{1}{R \cdot f_{out}} \sum_{k=0}^{R} \sum_{j=0}^{f_{out}} h_{i,j}^{(k)} \tag{1.1}$$

where N represents the size of the system in terms the number of nodes, f_{out} is the number of sent messaged by the i-th node at each gossip round, R is the number of rounds, and $h_{i,j}^{(k)}$ is a function returning 1 if the j-th message sent by the i-th node at the k-th round was able to recover a lost message, while it returns 0 in case of a unsuseful message. To improve the gossiping effectiveness, the following optimization problem needs to be resolved:

$$\max \frac{1}{N \cdot f^{*}_{out}} \sum_{i \in L} \sum_{j \neq i} x^{(k)}_{ij} \cdot h^{(k)}_{i,j} + \sum_{i \in L} \rho^{(k)}_{i}, \tag{1.2}$$

subject to:

$$j \in L_j^{(f^{(k)}_{out,i})} \subseteq L \tag{1.3}$$

$$\sum_{j \in L} x^{(k)}_{i,j} = f^{(k)}_{out,i} = |L_j^{(f^{(k)}_{out,i})}|, \quad \forall i, \tag{1.4}$$

$$x^{(k)}_{i,j} \in [0,1]. \tag{1.5}$$

$$0 \leq \rho_i \leq 1, \;\; \forall i \in L, \tag{1.6}$$

where f^{*}_{out} is the mean for $f^{(k)}_{out}$ over all the nodes at the given k-th gossip rounds. Specifically, the constraint in Equation 1.4 limits the number, namely $f^{(k)}_{out,i}$, of messages that each node has can forward per each round. It is possible that such a value is different and depends of the position of the node within the system and the possibility of providing an optimized utility (nodes within a portion of the network with high loss can have a higher value for $f^{(k)}_{out,i}$ than nodes in a more well-behaving portion), or it can be a constant, *i.e.*, $f^{(k)}_{out,i} = f^{(k)}_{out,j} \forall i \neq j$. Such an optimization problem can be approached in a centralized resolution means, where the selection of the solution can be performed at a single node within the system, by directly resolving it as a Mixed-Integer Program (MIP), or by relaxing the integral constraints and finding a solution to the corresponding pure Linear Program (LP). Such a resolution strategy is not feasible, since it demands the global knowledge of the loss statistics for all the links within the network, and within a large-scale asynchronous system, as the Internet, this cannot be achieved. Moreover, the problem has a large scale, and a vast solution space, if considered for a typical communication infrastructure such as the one supporting the vision of smart city or a rescue system in case of disasters. Another possible approach consists in fragmenting the problem in a number of subproblems and distributing the resolution of such sub-problems among all the nodes, by using local decisions based on the local knowledge acquired by measuring the network performance during the interactions with neighboring nodes. Specifically, in [10], the problem has been modelled as a non-cooperative game where the nodes in the overlay running the gossiping are the players, which are structured in a tree. The strategy of a player is given by an integer representing one of the layers in the tree, while its output is the random selection of $f^{(k)}_{out}$ nodes belonging to such a selected layer. To pick up the best strategy, the player needs to assign a cost or a payoff so as to select the strategy that minimize the cost or maximize the payoff. In our gossip game, by running the selected strategy, a player c obtains a gain, indicated with α, which depends on the utility of its gossip messages to recover lost messages during the previous k-th round, normalised over the total number of sent messages. Such feedback allows to characterize the payoff function, which is not formulated before-hand, but dynamically computed so as to follow the real networking conditions. Specifically, the learning scheme called COmbined fully DIstributed PAyoff and Strategy-RL (CODIPAS-RL) [42] has been used for the dynamic assessment of the payoff function. This is a scheme derived from strategy and payoff (Q-learning) Reinforcement Learning, with the use of the Boltzmann-Gibbs distribution as a strategy mapping. Specifically, the feedback of an action at the time instance t is used to update the payoff estimation, and such an estimation is used to determine the strategy as follows:

$$\begin{cases} x_{j,t+1} = (1-\lambda_{j,t})x_{j,t} + \lambda_{j,t}\widetilde{\beta}_{j,\varepsilon}(\hat{r}_{j,t}) \\ \hat{r}_{j,t+1}(s_j) = \hat{r}_{j,t}(s_j) + \nu_{j,t}\mathbb{1}_{\{o_{j,t+1}\in O_j(s_j)\}} \\ (r_{j,t+1} - \hat{r}_{j,t}(s_j)), \end{cases} , \tag{1.7}$$

$$\text{where } j \in [1,N], t \geq 0, o_{j,t} \in O_j(s_j).$$

The function $\mathbb{1}_{\{o_{j,t+1}\in O_j(s_j)\}}$ returns 0 if the action $o_{j,t}$ has not been played by the j-th node at time t, while it has an output of 1 otherwise. This allows to update only the contribution corresponding to the action that has been played. The game can have two resolution approaches: by means of pure strategies, *i.e.*, to pick a single action and play it, on the contrary to the other approach of or mixed ones, *i.e.*, to have strategies with a probabilistic selection over the set of available actions according to some probability distribution. The game theory assures that for mixed-strategy games it is always present at least one Nash equilibrium, while for pure strategies the existence of a Nash equilibrium point has to be demonstrated. The authors in [10] assumed mixed strategies, so the Nash theorem guarantees that at least one Nash equilibrium, or a solution for the game, it is always present [35].

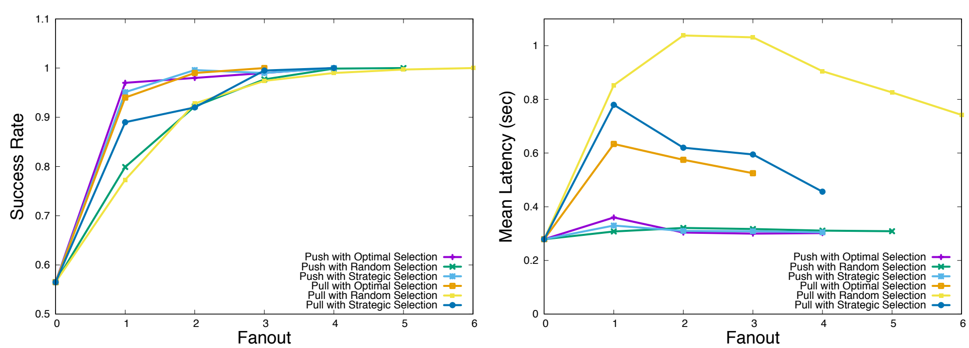

Fig. 1.9 Success rate and latency with strategic gossip.

By running simulations over lossy networks, the only improvement of the node selection brought by the application of game theory can improve the success rate, as illustrated in Fig. 1.2.3.1 by reducing the latency, as in Fig. 1.2.3.1. This is achieved by the reduction of gossip rounds that a destination needs to wait to obtain the notification of interest by tolerating the losses experienced throughout the network.

1.2.3.2 Network Embedded FEC

FEC is typically a centralized approach, where the data source encodes the outgoing messages to generate the deed spatial redundancy. Such a centralization simplifies the implementation of the solution; however, it also causes a serious scalability issue, as illustrated in Fig. 1.10. If within the network, the experience losses are unbalanced and there is a portion having more severe loss rates than the rest of the network, as in the left side of the figure, the encoder produces a high redundancy degree so as to tolerate such losses, even if not all of such a redundancy is useful. This implies that the network will be traversed by considerable traffic, unnecessary for the portions with a lower loss rate and dangerous since it can cause congestion phenomena. Such a case is extremely plausible during weather-based disasters, such as heavy rains and thunderstorms, compromising the RF signals in a portion of a city or critical infrastructure. To deal with such an issue, one would think to have an encoder in any node of the event notification overlay forwarding messages. Such an approach, mentioned as Network Coding [20] within the literature, is not a suitable solution. It exhibits more efficient traffic within the network, avoiding coarse-grain management of the redundancy as in the first case, but implies a high cost in terms of latency since any forwarding operation implies an encoding operation (and decoding to obtain the original notification before encoding it). A more viable solution is represented in the right side of the figure where only the node sending messages to the lossy portion of the overlay generates the additional redundancy to cope with the abnormal loss rate. Such a solution is named as Network Embedded FEC (NE-FEC) [9], and consists in identifying a subset of the interior nodes to act as encoders, to optimize the generated traffic, achieving a high success rate and reducing the performance costs.

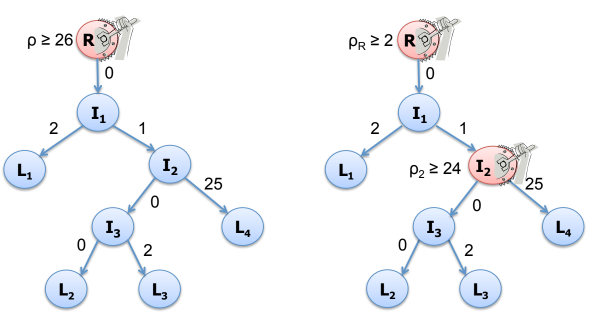

Fig. 1.10 Issues with FEC in unbalanced networks and a possible solution.

The encoder placement problem can be formulated as follows. Let be L the set of the m possible nodes acting as a codec, U the set of n subscribers, and D the n x m

matrix containing the transportation costs $d_{i,j}$ for delivering a message from the j node to the subscriber i, for all $j \in L$ and $i \in U$. In our problem, we have two sets of decision variables: (i) $y_j = 1$, if the $j \in L$ node is a codec, and 0, otherwise; and (ii) $x_{ij} = 1$, if the destination i receives redundancy from the node $j \in L$ being a codec, and 0, otherwise. The objective is to determine which nodes act as codecs so as to minimize the sum of these costs:

$$min \sum_{i \in U} \sum_{j \in L} d_{i,j} x_{i,j} + \sum_{j \in L} \alpha_j, \tag{1.8}$$

subject to:

$$\sum_{j \in L} x_{i,j} = 1, \quad \forall i, \tag{1.9}$$

$$x_{i,j} \leq y_j, \ \ \forall i, j, \tag{1.10}$$

$$\sum_{j \in L} y_j = p \leq n = |L|, \tag{1.11}$$

$$x_{i,j}, y_j \in \{0,1\}. \tag{1.12}$$

where the transportation costs $d_{i,j}$ are the redundancy generated by the codec in location j, and α_j are the costs of having a codec at the j-th node. Such costs are required in order to avoid the naive solution where each interior node acts as a codec. Such optimization is characterized by a series of constraints. First, Equation 1.9 limits the case where each subscriber has to receive redundancy only from one codec. Equation 1.10 prevents each subscriber having FEC messages from inactivated codecs. Last, Equation 1.11 specifies that the total number of codecs is set to p, lower than the total number of interior nodes. Within the literature, such optimization belongs to the class of the so called *P-Median Problems*, with the only exception being that the number of median points to be found is not fixed beforehand. This problem is typically approached as a Mixed-Integer Program (MIP), or as a pure Linear Program (LP) [26, 37] by relaxing the integral constraints. Resolving an optimization problem by selecting the best codec placement and the consequent tuning with a low convergence time, limited workload and resource consumption is a real challenge.

The traditional resolution approach is the centralized one considered previously for the optimized gossiping. As argued before, such a solution is not feasible in large-scale context, and a distributed approach is preferable, even if a sub-optimal placement is achieved. In [11, 12, 8], the authors have investigated several concepts coming from the game theory to design such a distributed resolution. Specifically, they have based their research on the rich literature on facility location games, rooted in the seminal work of Hotelling [24] and Downs [7]. Specifically, the simplest approach is to formalize the problem as the non-cooperative one-shot game, where each node of the overlay is assumed as a player deciding to be a codec or not. Each strategy has a cost, expressed as the generated redundancy when being a codec and consequent success rate at the nodes in the lower level of the multicast tree. The cost of using a given strategy to be spent by i-th player is the following one:

$$C_i(S_i) = \alpha_i \cdot S_i + (\rho_i + \max_{j \in U_i} \lambda_j) \cdot (1 - S_i), \tag{1.13}$$

where ρ_i represents the number of redundant messages received by the i-th node, λ_j is the number of messages sent by the i-th node and not received by the j-th node, and α_i expressing the cost for a node acting as a codec. Based on such a cost, the game is to have each player to minimize the costs of pursuing the strategy, so a strategy profile s represents a Nash Equilibrium (*i.e.*, a solution of our game) if and only if the two following conditions are guaranteed:

$$\exists i \in Y \text{ s.t. } \rho_i + \max_{j \in U_i} \lambda_j \leq \alpha_i \tag{1.14}$$

$$\nexists i \in Y \text{ s.t. } \rho_i < \alpha_i - \max_{j \in U_i} \lambda_j \tag{1.15}$$

The first condition tells that the number of redundant messaged sent by a codec running at the i-th node is never greater than α characterizing its children; so, none of its children is encouraged to promote itself as a codec. The second condition formulates the fact that a codec is not encouraged to stop being a codec since the paid cost is already minimized. Such an approach has been assessed using simulations has done before for gossip, and results are shown in Fig. 1.11. The distributed game-theoretic NE-FEC has been compared with a centralized resolution of the codec placement by using a Genetic Algorithm (GA). Such a solution offers high resiliency without incurring in an excessive latency worsening, even if the found placement is sub-optimal. This is clear in the figure as the measured performance is slightly lower than the case of a placement determined by the centralized approach.

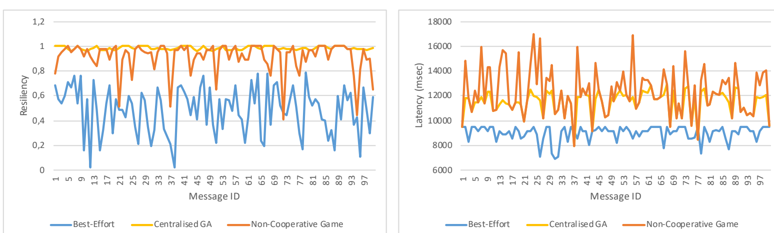

Fig. 1.11 Success rate and latency with NE-FEC.

The non-cooperative game does not return a Pareto-optimal solution for the codec placement and tuning (and this is a well-known result in the literature), due to the lacking coordination among the nodes. However, since the game is repetitively run over the time since the loss statistics changes continuously, the players are aware of the strategies followed by the others and may consequently adapt their strategies. To this aim, a cooperative formulation, is more suitable, such as the one by Nash Bargaining, which consists of running a non-cooperative game until reaching the equilibrium and afterward the iterative bargaining is started by exchanging feedback with the other nodes. Fig. 1.12 shows that the cooperative formulation is more optimal since the placement allows achieving a higher resiliency degree, even if the complete success rate is not feasible and this is an intrinsic limitation of an FEC scheme due to the fact that the experienced loss rate in a time frame may diverge from what measured in the previous frames.

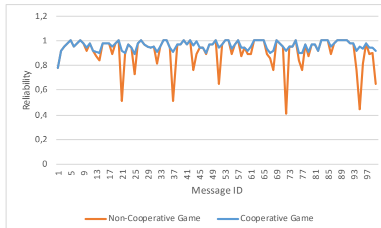

Fig. 1.12 Cooperative codec placement game versus the non-cooperative one.

1.3 Resilient Unicast Communications

Many research papers discuss issues and technical solutions to natural disasters in the context of network communication. Most of these solutions operate at the network level, by enhancing the routing algorithm. This section introduces routing approaches not tailored to event-based communications and or multicasting, but has a more generic perspective and can be used for unicast communications as their first application case.

1.3.1 Spatial Redundancy via Floating Breadcrumbs

Many post-disaster scenarios are characterized by low densities of agents (vehicles, smartphone-enabled users). Such low densities of users might be due to the disaster itself, which often makes it uninhabitable or difficult to reach and traverse a port ion of the territory. For these reasons, delay-tolerant routing techniques (DTN) for content routing and delivery play a central role in such scenarios, in which ”store, carry and forward” is the main way in which information moves in space and across agents.

We consider scenarios, where the exchanged messages are not time-critical, and where there is a low likelihood to find an end-to-end path between two peers, due to node sparsity in space and to their high dynamicity. In a disaster scenario, a peer might search for a person or a service without knowing exactly the identity of the other peer providing the information or the service, nor its exact geographical location, but only having an approximate knowledge of the area where the person (or the information about him/her) could be located.

The traditional approach to this problem is geocasting, and it consists of sending a query into a destination area. The query is typically forwarded to the area via one of the many techniques available of DTN routing. Once the message reaches the area, it is replicated locally to have the query reach the intended recipient with a sufficiently high level of likelihood. The reply is assumed to be routed again via some form of DTN routing strategy, based on the spatial coordinates of the sender and of the originator. Indeed, as the originator can indicate his position in space, this information can be used to route back the reply.

However, in highly dynamic environments such as those which characterize the aftermath of a disaster and in which store, carry and forward is the main way to spread a message, a significant delay might incur between the time in which the query is issued, the time at which it finally reaches the intended recipient(s), and the time at which the reply to the query reaches the query originator. The mobility of the originator and the delay between the query origination and the delivery of the reply might make delivery of the reply a challenging task, as the originator might not be any more at the position in which he was at the time of issuing the query.

A possible way to tackle this issue is through some form of geographically constrained flooding. That is, once the query reaches the target area, or when the reply reaches the location of the originator, the content could be replicated opportunistically in a region around the target location or area. Such region should be of such size and shape as to strike a compromise between likelihood of having the message reach the intended recipient (or mean time required by the message to be delivered to its intended recipient), and amount of resources involved (e.g., mean number of message replications, or mean number of agents possessing a copy of the message).

Such geographically constrained flooding, which in the literature is also denoted as Floating Content (FC), has been recently well studied, and a set of analytic tools can be employed in order to address the issue of the optimal dimensioning of the replication area, both at the destination site of the query, and at the location of the originator.

However, for what concerns the issue of having the reply message efficiently reach the query originator, FC approach alone might be highly inefficient, possibly requiring a large floating content area and hence employ a significant amount of resources. To deliver the reply to the originator in a more resource-efficient manner, the Breadcrumbs Geocasting Routing (BGR) has been proposed [43].

In BGR, the originator periodically spreads around him (via FC) a message (the "breadcrumb") containing information about his current position. The goal of this information is to support a location-based DTN routing protocol to route the reply to the originator. As a result, once reached the location in which the originator was at the time at which it issued the query, the reply follows the originator by collecting the information in his breadcrumbs. The effectiveness of the BGR approach in decreasing the resources required to deliver the reply to the originator depends on several parameters, among which are the radius of the FC of the breadcrumbs, the frequency at which they are produced, as a function of the mobility patterns of the agents and their density.

1.3.2 Natural disaster management system based on location-aware distributed sensor networks

The work in [36] presents an efficient routing protocol for disaster recovery in wireless sensor networks. This routing algorithm can be used in natural disaster management systems that monitor wind velocity, tectonic waves, tides, etc.

The algorithm uses the hierarchical transmission of packets from sensor nodes to a base station by identifying a path from one node to a subsequent node along the route. The purpose of this is to achieve optimal path discovery for routing information packets.

The sensor network is divided into logical concentric zones based on energy transmission. Frames are routed from one zone to a zone with lesser depth based on a greedy algorithm. To determine the next hop, the local angular deviation between two communicating nodes is used. The implementation relies heavily on the location awareness sensor nodes and can be applied only where data can be made available at the time of installation. A node location is expressed by polar coordinates concerning the base station and initially, they are recorded during the sensor installation. Once installed both the nodes and the base station are stationary.

After installation, each node sends “hello” packets to form a list of neighbors and their coordinates. After this flood of messages, clusters are formed in each zone based on the distance between nodes. Every node is located in a single zone, a single cluster of that zone and each cluster has a designated node responsible for the transmission of packets towards the base station. Within a cluster, all the members send towards the cluster head their data. When the primary node’s energy falls below a certain threshold, a new primary node is appointed in that cluster. The primary node is chosen solely for having the highest power in the cluster and before quitting the primary node appoints its successor.

Primary nodes maintain a list of secondary nodes which are all other nodes within one hop in the next zone in case of primary node failure. If a node faults, the node discovery phase is re-initiated for all the nodes with a higher depth which were communicating with it. Over a longer time, routes are formed and packets are routed automatically along previously discovered routes.

This algorithm leads to local path discovery and alternative path information maintenance. Minimum redundancy in terms of packets transmitted is maintained. Also, energy consumption is balanced by shifting the role of the primary node.

1.3.3 Analyzing Geo-Path diversity and improving routing performance in optical networks

Regarding optical networks, paper [3] discusses a network vulnerability identification mechanism and different vulnerability scales using real-world optical network data. The authors also propose two heuristics for solving the path geo-diverse problem (PGD) which are implemented in Geo-path Diverse Routing Protocol (GeoDivRP).

The primary concern of GeoDivRP is to achieve increased resilience and improved performance of routing by taking into consideration the geographical diversity of the paths and selecting alternative routes to circumvent challenged areas. Geographical diversity is defined as how much are two paths separated from each other in a geographical context and it represents the minimum distance between any node of the path and the minimum path. Path geo-diversification is a new mechanism used to quantify the Geo-Path diversity graph formed with the nodes in the network by selecting multiple geographically diverse paths between a source and a destination node, using a quantified geo-diversity measure.

A new distance routing algorithm is incorporated in GeoDivRP which considers geographical diversity and provides multiple paths to circumvent challenged areas. Challenges are modeled as a circular area with a certain challenge radius. The algorithm follows the link-state routing methodology. It starts by gathering link-state update messages within a certain radius which include geo-location (latitude and longitude), computes paths and stores them in a cache. The paths are used for routing and circumventing challenged areas, standard Open Shortest Path First (OSPF) is used for any other routing decisions. The minimum distance between paths is user-defined and can be used by a network administrator to quickly circumvent challenged areas.

GeoDivRP uses Suurballe's algorithm [41] to compute paths by adding additional weight to the edges and uses a modified version of Dijkstra's algorithm to choose the shortest path with a certain minimum distance between it and the absolute shortest path. Tests have been done over a simulated environment and not using real-world routers. They show that it obtains better results than OSPF when challenges occur but have added computational overhead when there are no challenges and about 1-millisecond delay.

1.3.4 Capacity Constrained routing algorithm for evacuation planning

A heuristic algorithm that produces sub-optimal solutions to the evacuation planning problem is presented in [28]. Capacity Constrained Route Planner (CCRP) is an algorithm that models capacity as a time series and uses a routing approach that incorporates route capacity constraints. This algorithm is meant to address the limitations of the existing linear programming approach, it does not require any prior knowledge of evacuation time, it can use the existing evacuation network and it should scale better with the size of the network.

The algorithm is given an evacuation scenario which consists of nodes with initial capacity and maximum capacity, edges which also have a capacity and a travel time, and a list of destination nodes. It then attempts to minimize the time for the last evacuee to reach the destination. The currently used algorithms are better suited for evacuation scenarios with moderate-size networks and cannot scale to large networks in urban evacuation scenarios so a heuristical approach is much better suited for such scenarios.

CCRP uses an iterative approach. Each iteration consists of a search for the route with the earliest destination arrival time from any source node to any destination node, taking previous reservations and possible waiting time into consideration. Next, it computes the actual amount of evacuees that will travel through that route, this amount being affected by the capacity of the route, and also computes the remaining number of evacuees. The capacity of the route is reserved while a group of evacuees is using it. The iteration continues until all evacuees reach a destination node. A modified version of Dijkstra's algorithm is used for this search. To reduce computational time, a virtual node that is connected to all other source nodes is added.

Tests were done over a simulated network of 20 million nodes and 80 million edges and also in a real environment and the results show a significantly reduced computational cost and improved evacuation time from previous solutions. There is also a fault-tolerance aspect to this algorithm which is represented by the iterative computation where any change in the network topology is quickly taken into account. This solution assumes global control over the network flow which is currently not applicable for real computer networks and also comes with a high computational cost.

1.4 Weather Disruption-Tolerant Self-optimizing Millimeter Mesh Networks

A network-layer approach in the form of predictive weather assisted routing protocol (P-WARP) is discussed in [25] to address the problem of slow recovery and attenuation of network speed in wireless networks which use millimeter-wavelengths, due to storms. These types of wireless networks operate at extremely high frequency and raindrops cause a high reduction in frame transfer rate.

P-WARP uses a predictive approach to completely circumvent storms. This is done using a Doppler radar which offers short term weather prediction, for a warning of a few minutes which gives the protocol enough time to re-route all traffic before the link failure. It is based on OSPF with some notable differences. It uses the weather-radar to forecast future link conditions and adjust link costs. Threshold values for measurements are used to determine when a storm is about to occur or when a storm has ended and weight is added or removed from links for OSPF to find the appropriate route and the storm to be circumvented. Data processing is done by core or edge nodes, and a single update message is flooded for all nodes in an area by core nodes. Routes are recomputed using OSPF but link state advertisements (LSA) are not sent to adjacent nodes.

Tests were done in a simulated environment and also in a real storm. P-WARP reroutes predictively with no packet loss, maintaining service availability. Results show a slightly better performance than standard OSPF. There is an added delay due to packet rerouting and overhead of additional computation. Also, multiple nodes that can withstand the additional traffic are required to avoid congestion.

1.5 UAV support to network resiliency

Unmanned Aerial Vehicles (UAVs) are autonomous cheap aerial devices, without the need of a pilot to be onboard during the flight, with the possibility of being equipped with a camera, sensing hardware and peripherals for RF communications. UAVs are getting more and more popularity for military, public, civil, and even personal applications [44], and in particular UAVs can be properly placed to support the damaged cellular connectivity and performance [34, 19]. The correctness of the placement of a UAV depends on the position of the other ones and the state of the cellular network within the area. Merwaday and Guvenc [29] proposed to use UAVs as aerial base stations (UABSs) to assist public safety communications during natural disasters, as soon as parts of the communication infrastructure become damaged and dysfunctional. They showed that the deployment of UABSs at optimized locations can improve the throughput gains under disaster scenarios. Sharma et al. [40] introduced a neural network-based cost function to assign UAVs to a particular geographical area subject to high traffic demands. Their results showed that leveraging multiple UAVs not only provides long-range connectivity but also better load balancing and traffic offloading. Based on the target application, several attempts to determining the UAV node position and/or trajectories to meet the application needs. The first example, such as in [2, 31, 33, 22] is represented by the proper deployment of the UAV to extend the wireless coverage in rural areas, emerging countries or where BS density is not sufficient to cover the mobile user demands. The use of UAV for network recovery and the design of public safety networks is a recent application closely related to the previous ones, as described in [23]. Another possible application is referred to as the improvement of IoT Applications [32], by having the UAV to move around and collect the data from IoT devices on the ground, to reduce energy consumption concerning the traditional interaction with a cellular BS or the integration of Internet connectivity to each IoT device. UAV-assisted networking paves the way for novel interesting applications, coupled with the advent technologies under the umbrella of 5G. A key aspect is the proper management of placement and mobility of UAVs, and the possible resolution approaches can be centralized or distributed. In the first case, the overall variables of the problem are collected within a single point of decision. Multiple different approaches are spanning from the stochastic geometric framework in Hayajneh et al. [23], to a model as an optimization problem solved by using a mixed-integer non-linear problem in Alzenad et al. [2] or Han et al. [22]. In the second case, each UAV decides where to move and when to change the proper trajectory and placement. The main approach used in this case is the game theory for the coalition formation among UAV, as in Saad et al. [38], or the non-cooperative formation of the placement problem in Abdulla et al. [1], where each UAV is a player determining the position or movement that maximize its payoff (for the energy efficiency).

1.6 User Association in Emergency Networks

The use of Unmanned Aerial Vehicles (UAVs) for resilient networking as additional aerial base stations (UABSs) to assist public safety communications during natural disasters, as well as to substitute or complement parts of the communication infrastructure being damaged and dysfunctional. This leads to the case that a user device can have multiple possible base stations towards the network. A user device may exploit ad-hoc connections established with other user devices with ad hoc networking, with peers acting as forwarders. The user device can leverage surrounding base stations to access the Internet through cellular networks. Last, the user device would make use of UAVs for communication purposes, and the contacted UAV can exploit an ad-hoc network with other UAVs or have direct Internet accessibility utilizing long-range cellular or even satellite communication means. A user device can select one of these three communication opportunities, or even more than one to realize multi-path routing, based on the offered QoS and the required cost in terms of consumed energy. If fact, each of them can provide a certain degree of performance and success rate, but the required power consumption is not uniform as each of them leverages specific communication technologies. For example, ad-hoc networks among user devices are built based on low-range RF technologies such as Bluetooth and WiFi, which exhibits a lower energy cost than the cellular ones to connect to a base station. Reaching UAVs demands long-range RF technologies, where the energy cost can be higher. Such a selection depends on the actual network conditions and demands, and cannot be determined once but must be continuously conducted over time.

It is possible to model such an issue as an optimization problem where minimizing the energy consumption and maximizing the communication success rate and minimizing the latency. Such a problem is intractable in a centralized manner, even with heuristic approaches such as genetic algorithms and so on. In fact, it can only be resolved only in a distributed manner, such as by leveraging on a game-theoretic approach as in [16], where each user device is considered as a player in a non-cooperative game, which picks up a given strategy (i.e., using one of the possible communication means or even more than one) by maximising the obtainable gain (in terms of meeting the demands of the applications and users in terms of performance and success rate) and minimising the consequent cost in terms of energy consumption. Such a work showed through simulations that initially the loss rate experienced by the network is low. After the disaster's occurrence, the loss rate increases to a value that is stationary when the user is kept associated with its initial BS, but later on such a value lower due to the adaptation to the multiple associations determined by the approach. Such an initial study has to be extended by applying approaches able to return solutions better than the suboptimal ones had by a non-cooperative game resolution, or integrating the multiple association with the ad-hoc networks established by user devices and UAVs.

1.7 Final Remarks

This chapter described some state of the art and recent routing algorithms to be used in unicast communications and multicast ones, based on events, to achieve disaster-resilient networks. Generally speaking, disaster-resilient routing is typically possible with proactive and reactive approaches. In the first case, the network is designed and equipped with alternative transmission means (e.g., backup paths, redundant topology) in advance (before the occurrence of failure) so that the damages caused by a disaster cannot compromise the node connectivity within the network or even the perceived QoS from the users. The reactive ones are, in turn, executed only after a failure to try to provide message delivery despite the severe loss patterns caused by the consequences of occurred disasters by introducing redundancy at the message level. Such routing mechanism may exhibit vulnerabilities that can be exploited to realize attacks and/or to compromise the applications built on top of gossip. Dealing with such security-related issues is of central importance. Besides, a recent application of Unmanned Aerial Vehicles (UAVs) is posing a novel possibility and also challenges that future research activities should investigate.

Acknowledgement

This chapter is based on work from COST Action CA15127 ("Resilient communication services protecting end-user applications from disaster-based failures - RECODIS") supported by COST (European Cooperation in Science and Technology).

References

1. A. E. A. A. Abdulla, Z. Md Fadlullah, H. Nishiyama, N. Kato, F. Ono, and R. Miura. Toward fair maximization of energy efficiency in multiple uas-aided networks: A game-theoretic methodology. *IEEE Transactions on Wireless Communications*, 14(1):305–316, 2015.
2. M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu. 3-d placement of an unmanned aerial vehicle base station (uav-bs) for energy-efficient maximal coverage. *IEEE Wireless Communications Letters*, 6(4):434–437, 2017.
3. Y. Cheng, M. T. Gardner, J. Li, R. May, D. Medhi, and J. P. G. Sterbenz. Analysing geopath diversity and improving routing performance in optical networks. *Computer Networks*, 82:50 – 67, 2015.
4. M. Cinque, D. Cotroneo, C. Esposito, M. Fiorentino, and S. Russo. A reliable crisis information system to share data after the event of a large-scale disaster. In *Proceeding of the 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous*

Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, pages 941–946. IEEE, 2015.
5. M. Cinque, C. Di Martino, and C. Esposito. On data dissemination for large-scale complex critical infrastructures. *Computer Networks*, 56(4):1215–1235, 2012.
6. M. Cinque, C. Esposito, M. Fiorentino, F. J. P. Carrasco, and F. Matarese. A collaboration platform for data sharing among heterogeneous relief organizations for disaster management. In *Proceeding of the ISCRAM 2015 Conference*, 2015.
7. A. Downs. An economic theory of political action in a democracy. *The Political Economy*, pages 12–26, 1984.
8. C. Esposito, A. Bruno, G. Cattaneo, and F. Palmieri. On the optimal tuning and placement of fec codecs within multicasting trees for resilient publish/subscribe services in edge-iot architectures. *Future Generation Computer Systems*, 88:140 – 150, 2018.
9. C. Esposito, A. Castiglione, F. Palmieri, and M. Ficco. A game-theoretic approach to network embedded fec over large-scale networks. In *Computational Intelligence and Intelligent Systems*, 2016.
10. C. Esposito, A. Castiglione, F. Palmieri, and M. Ficco. Improving the gossiping effectiveness with distributed strategic learning (invited paper). *Future Generation Computer Systems*, 71:221 – 233, 2017.
11. C. Esposito, A. Castiglione, F. Palmieri, and M. Ficco. Building a network embedded fec protocol by using game theory. *Information Sciences*, 433:365–380, 2018.
12. C. Esposito, A. Castiglione, F. Palmieri, F. Pop, and J. Rak. repeated game formulation of network embedded coding for multicast resilience in extreme conditions. In *Proceedings of the 13th International Conference Design of Reliable Communication Networks (DRCN 2017)*, pages 1–8, March 2017.
13. C. Esposito, M. Ciampi, and G. De Pietro. An event-based notification approach for the delivery of patient medical information. *Information Systems*, 39:22–44, 2014.
14. C. Esposito, D. Cotroneo, and S. Russo. On reliability in publish/subscribe services. *Computer Networks*, 57(5):1318–1343, 2013.
15. C. Esposito, M. Platania, and R. Beraldi. Reliable and timely event notification for publish/subscribe services over the internet. *IEEE/ACM Transactions on Networking*, 22(1):230–243, Feb 2014.
16. C. Esposito, Z. Zhao, R. Alcarria, and G. Rizzo. Game theoretic optimal user association in emergency networks. *Proceedings of the 18th International Conference on Ad Hoc Networks and Wireless (AdHoc-Now 2019)*, 2019.
17. P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec. The many faces of publish/subscribe. *ACM computing surveys (CSUR)*, 35(2):114–131, 2003.
18. P. T. Eugster, R. Guerraoui, A.-M. Kermarrec, and L. Massoulié. Epidemic information dissemination in distributed systems. *Computer*, 37(5):60–67, 2004.
19. L. Ferranti, F. Cuomo, S. Colonnese, and T. Melodia. Drone cellular networks: Enhancing the quality of experience of video streaming applications. *Ad Hoc Networks*, 78:1–12, 2018.
20. M. Ghaderi, D. Towsley, and J. Kurose. Reliability gain of network coding in lossy wireless networks. In *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*, pages 2171–2179, April 2008.
21. J. Han, D. Watson, and F. Jahanian. An experimental study of internet path diversity. *IEEE Transactions on Dependable and Secure Computing*, 3(4):273–288, Oct 2006.
22. Z. Han, A. L. Swindlehurst, and K. J. R. Liu. Optimization of manet connectivity via smart deployment/movement of unmanned air vehicles. *IEEE Transactions on Vehicular Technology*, 58(7):3533–3546, 2009.
23. A. M. Hayajneh, S. A. R. Zaidi, D. C. McLernon, M. Di Renzo, and M. Ghogho. Performance analysis of uav enabled disaster recovery networks: A stochastic geometric framework based on cluster processes. *IEEE Access*, 6:26215–26230, 2018.
24. H. Hotelling. Stability in competition. In *The Collected Economics Articles of Harold Hotelling*, pages 50–63. Springer, 1990.

25. A. Jabbar, B. Raman, V. S. Frost, and J. P. G. Sterbenz. Weather disruption-tolerant self-optimising millimeter mesh networks. In *Self-Organizing Systems*, pages 242–255. Springer Berlin Heidelberg, 2008.
26. A. Klose and A. Drexl. Facility location models for distribution system design. *European journal of operational research*, 162(1):4–29, 2005.
27. S. Lin, D. J. Costello, and M. J. Miller. Automatic-repeat-request error-control schemes. *IEEE Communications magazine*, 22(12):5–17, 1984.
28. Q. Lu, B. George, and S. Shekhar. Capacity constrained routing algorithms for evacuation planning: A summary of results. In *Advances in Spatial and Temporal Databases*, pages 291–307, 2005.
29. A. Merwaday and I. Guvenc. Uav assisted heterogeneous networks for public safety communications. In *2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pages 329–334, March 2015.
30. K. Miranda, A. Molinaro, and T. Razafindralambo. A survey on rapidly deployable solutions for post-disaster networks. *IEEE Communications Magazine*, 54(4):117–123, April 2016.
31. M. Mozaffari, W. Saad, M. Bennis, and M. Debbah. Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage. *IEEE Communications Letters*, 20(8):1647–1650, 2016.
32. M. Mozaffari, W. Saad, M. Bennis, and M. Debbah. Mobile unmanned aerial vehicles (uavs) for energy-efficient internet of things communications. *IEEE Transactions on Wireless Communications*, 16(11):7574–7589, 2017.
33. M. Mozaffari, W. Saad, M. Bennis, and M. Debbah. Wireless communication using unmanned aerial vehicles (uavs): Optimal transport theory for hover time optimization. *IEEE Transactions on Wireless Communications*, 16(12):8052–8066, 2017.
34. M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah. A tutorial on uavs for wireless networks: Applications, challenges, and open problems. *IEEE Communications Surveys & Tutorials*, 2019.
35. M. J. Osborne and A. Rubinstein. *A course in game theory*. MIT press, 1994.
36. G. Ranjan and A. Kumar. A natural disasters management system based on location aware distributed sensor networks. In *Proceedings of the IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, 2005.*, pages 3 pp.–182, Nov 2005.
37. J. Reese. Solution methods for the p-median problem: An annotated bibliography. *NETWORKS: An International Journal*, 48(3):125–142, 2006.
38. W. Saad, Z. Han, T. Basar, M. Debbah, and A. Hjorungnes. A selfish approach to coalition formation among unmanned air vehicles in wireless networks. In *Proceedings of the 2009 International Conference on Game Theory for Networks*, pages 259–267. IEEE, 2009.
39. M. R. Selim, Y. Goto, and J. Cheng. A replication oriented approach to event based middleware over structured peer to peer networks. In *Proceedings of the 5th international workshop on Middleware for pervasive and ad-hoc computing: held at the ACM/IFIP/USENIX 8th International Middleware Conference*, pages 61–66. ACM, 2007.
40. V. Sharma, M. Bennis, and R. Kumar. Uav-assisted heterogeneous networks for capacity enhancement. *IEEE Communications Letters*, 20(6):1207–1210, 2016.
41. J. W. Suurballe. Disjoint paths in a network. *Networks*, 4(2):125–145, 1974.
42. H. Tembine. *Distributed Strategic Learning for Wireless Engineers*. CRC Press, 2012.
43. J. Timpner and L. Wolf. Query-response geocast for vehicular crowd sensing. *Ad Hoc Networks*, 36:435–449, 2016.
44. Y. Zeng, R. Zhang, and T. J. Lim. Wireless communications with unmanned aerial vehicles: Opportunities and challenges. *IEEE Communications Magazine*, 54(5):36–42, 2016.