Novel Concepts for Device-to-Device Communication Using Network Coding

Peyman Pahlevani, Martin Hundebøll, Morten V. Pedersen, Daniel Lucani, Hassan Charaf, Frank H. P. Fitzek, Hamidreza Bagheri, and Marcos Katz

Peyman Pahlevani, Martin Hundebøll, Morten V. Pedersen, Daniel E. Lucani, and Frank H.P. Fitzek are with Aalborg University, Denmark.

Hassan Charaf is with Budapest University of Technology and Economics, Hungary.

Hamidreza Bagheri and Marcos Katz are with the University of Oulu, Finland.

This work was financed in part by the Green Mobile Cloud project granted by the Danish Council for Independent Research (Grant No. DFF-0602-02661B), the Colorcast project granted by the Danish Ministry of Science (Grant No. 12-126424/FTP), and performed in part in the framework of the Celtic Green-T project.

ABSTRACT

Device-to-device communication is currently a hot research topic within 3GPP. Even though D2D communication has been part of previous ad hoc, meshed and sensor networks proposals, the main contribution by 3GPP is that the direct communication among two devices is carried out over a dynamically assigned, licensed spectrum; thus, it is under full control of the cellular network. D2D communication creates a market potential for new services, new approaches to efficient spectrum use, and security concepts. This is especially true if D2D communication is extended to larger communication groups organized in meshed clusters. In this article, we discuss the potential and shortcomings of D2D communication as proposed today, advocating for the use of network coding as an enabling technology for enhanced security and communication efficiency using the PlayNCool and CORE protocols as key examples to deliver smarter D2D systems.

INTRODUCTION

Device-to-device (D2D) communications have been identified by the Third Generation Partnership Project (3GPP) as a potential candidate technology to offload heavily used cellular communication systems. Even though D2D communications has been around in industry and research for the last two decades, the 3GPP decision to include this concept in the cellular context is groundbreaking. There are several fundamental differences between D2D communications in an ad hoc fashion and under cellular control. While the former typically uses a distributed topology, the latter relies on centralized control from a base station assigning communication resources. It also provides a natural market for novel services (including payment models) as well as easing the setup of D2D communication, as node and service discovery is supported or fully carried out by the cellular network. The result is faster setup and less time- and energy-consuming maintenance of the system as a whole.

In the 1990s, the D2D communications concept was introduced for ad hoc, sensor, and mesh networks. Ad hoc communication was initially intended to allow fast setup between two peers. Bluetooth is an example of D2D cable replacement solutions. Sensor and meshed networks extended the D2D concept by allowing not only two devices but multiple devices to communicate with each other. While sensor networks focused on accumulation of services (sensor data) from distributed nodes, mesh networks intended to offer coverage extension of devices with the same set of capabilities. The communication among devices was often realized in the industrial, scientific, and medical (ISM) bands in a distributed fashion without the help of any control unit.

Currently, D2D communications is realized by logical channels communicating via the base stations of a network (Fig. 1). Long Term Evolution-Advanced (LTE-A) D2D allows direct communications between devices using licensed spectrum if the mobile devices are in close proximity with the overlay network acting as a spectrum broker. This idea was originally introduced in 2006 [1] followed by several publications in that field summarized in [2].

The advantages of cellular controlled D2D communications are many-fold. From the mobile device perspective, the service discovery and node discovery can be performed by the network, which results in large energy savings compared to standard approaches. The reason is that the network is already aware of the position of its devices as well as the services that are needed or advertised by each device. Thus, the network can inform interested devices when to start the search procedure instead of constantly searching for the required service until a suitable device is found. Additionally, cellular controlled D2D can assign devices a series of resources to use, from wireless access to providing security keys. This provides devices with a common broker and frees them from negotiating directly with other devices. This reduces complexity and energy consumption for supporting these capabilities, and enables additional security mechanisms to detect and block malicious devices.

For the network, one of the key advantages is the potential to improve spectral reuse. When a set of devices communicate in a certain area, they are allocated a certain spectrum to use. As the power level of the devices is smaller than that of the base station, the network can assign this spectrum again in a different region of its coverage area. If we assume three pairs attempt to exchange one packet each, in state-of-the-art cellular systems these devices would relay their packets via the base station, resulting in six communication slots. If the pairs are spatially separated, a D2D approach controlled by the cellular network can reduce the exchange to a single communication slot with each device transmitting directly to its intended destination (i.e., reduced by a factor of six). The gain is higher if the transmission rate for the direct link is higher than the cellular link. This assumption typically holds as shorter distances result in higher signal-to-noise ratios even if the power is limited for local communication.

However, cellular controlled D2D communication is not limited to point-to-point scenarios for only two devices. The D2D network may be composed of multiple devices connected in a multihop fashion, with some devices relaying packets from others. Although technically interesting, users of relaying devices need to receive the appropriate incentives to cooperate to allow their devices to support these new services [3]. An alternative to ad hoc incentives is for the cellular network to monitor and reward cooperative behavior. If the D2D communication is composed of multiple devices, we refer to this communication cluster as a mobile cloud. A mobile cloud is a cooperative arrangement of dynamically connected devices sharing resources opportunistically [3]. In general, a mobile cloud can exploit different cooperative strategies, where communications between nodes can be established using unicast flows, and multicast and broadcast techniques through single- and multihop communications, as given in Fig. 2.

Close proximity of multiple devices improves the potential benefits of the mobile cloud, as the devices may share more diverse resources, from sharing multimedia content from one device to many devices [4] to improving communication with the cellular and/or overlay network by implementing local retransmissions among the devices in the D2D cluster. An example of the latter is the transmission of content in a sports stadium with thousands of users (e.g., instant replay). Transmitting the content using unicast flows from the cellular network is not an option, and broadcasting may require very robust coding and modulation, which translates into low-rate transmissions. However, users in a mobile cloud using D2D communications can help each other by sharing successfully received packets with those devices that did not receive enough of them. These local retransmission schemes are more efficient and scalable for such large deployments, and will help create new services and offload the cellular/overlay network.

This article focuses on different fields of use for D2D communication under the control of cellular network operators and the benefits of using network coding as an enabler of D2D communication in mobile clouds with a focus on two novel protocols: PlayNCool and CORE.

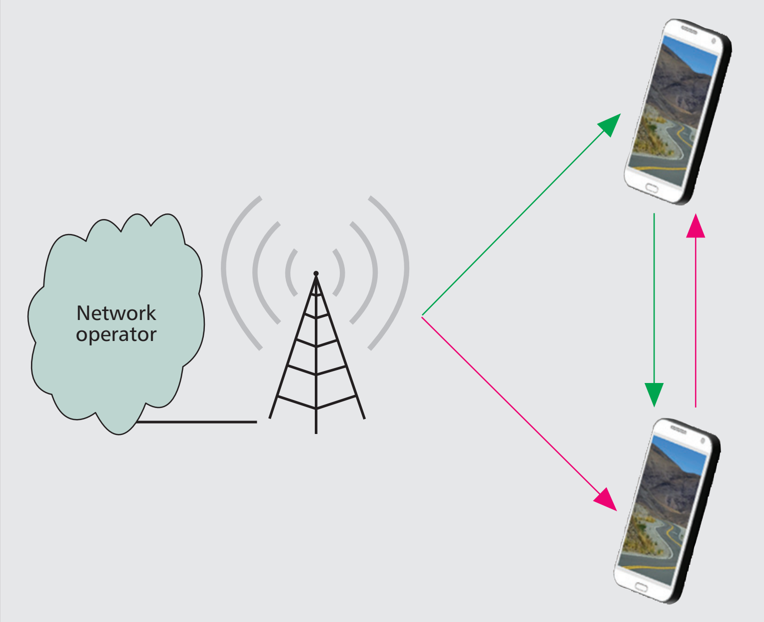

Figure 1. Device-to-device communication under cellular control.

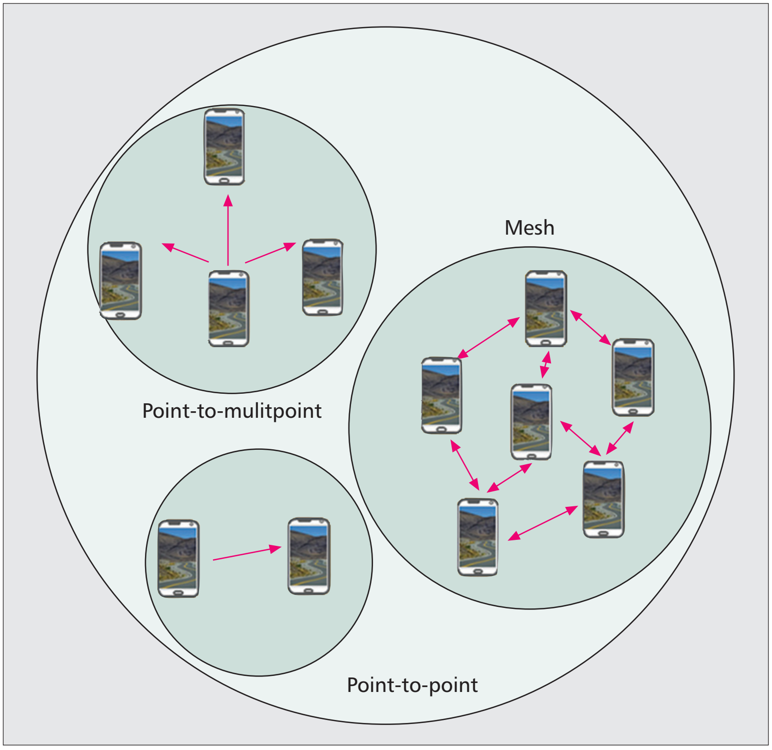

Figure 2. Different D2D cluster architectures: point-to-point, point-to-multipoint, and mesh.

ON THE NEED FOR NETWORK CODING

As the mobile cloud concept is a very promising use case, we look into this concept in more detail. We advocate for network coding [5] as the key technology to support the mobile cloud concept as proposed in [3]. The choice of tech-

The improved performance of network coding over the store-and-forward paradigm comes from the fact that packets are now seen as algebraic entities introducing the new concept compute-and-forward. This idea is groundbreaking as it is no longer focused on a philosophy of atomic nature of information packets.

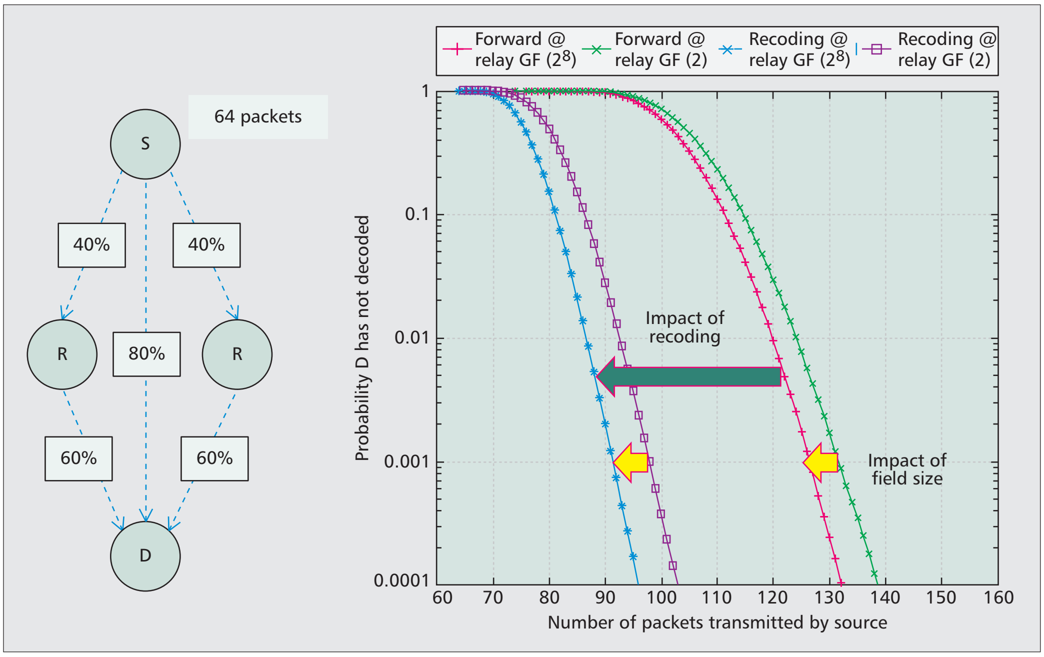

Figure 3. Inverse probability density function of receiving 64 packets at the destination (D) from a source (S) using two relays (R).

nology is motivated by the fact that network coding provides efficient communications within a meshed D2D cluster handling the dynamics of the wireless channel and device mobility seamlessly. We assume that a D2D cluster may be composed of multiple devices connected in a multihop fashion (Fig. 2). Network coding will also help if there are only two devices providing erasure correction capabilities, but its true potential over store-and-forward becomes evident as the number of devices increases.

The improved performance of network coding over the store-and-forward paradigm comes from the fact that packets are now seen as algebraic entities, introducing the new concept of compute-and-forward. This idea is groundbreaking as it is no longer focused on a philosophy of the atomic nature of information packets. This concept provides a new means of enhancing reliability, throughput performance, security, and network design and operation.

Conceptually, there are two different network coding approache: inter-flow and intra-flow network coding. The former focuses on the combination of packets from different flows across the network, while the latter focuses on the combination of packets of the same flow only. The inter-flow approach is typically implemented by identifying sets of nodes that can form a coding region. Packets from different flows can then be XORed bit by bit for attaining higher spectral efficiency. Thus, this approach only has low computational complexity for coding, but requires heavy signaling and bookkeeping among communication nodes to realize its full potential. A drawback of this approach is that it is not very resilient to packet losses in the system.

On the other hand, intra-flow is more computationally complex, but does not require overwhelming signaling information and provides higher resiliency to packet losses in the system. Thus, intra-flow network coding is well suited for D2D communications in the mobile cloud. Intra-flow network coding usually relies on random linear network coding (RLNC), which constitutes a distributed technique to code and recode packets in the network. RLNC organizes uncoded packets in a group, called a generation, to be linearly combined using randomly chosen coefficients from the elements of a finite field. These linear combinations are the coded packets sent through the network. Both the field size and generation size determine the complexity and efficiency of the code.

The name network coding comes from the fact that each node in the network is able to encode, recode, or decode packets. This makes it fundamentally different from end-to-end codes. While encoding and decoding are two well-known functionalities, recoding is a new one. Recoding allows each node to code again on already coded packets without needing to decode the packets first, and can be done even with partial information. This has several advantages, especially for distributed systems. Figure 3 illustrates the performance of different transmission strategies for a source sending 64 packets to the destination via one direct path and two relays. Each link is given a certain packet loss probability. Figure 3 shows the probability that the destination has received all 64 packets after a certain number of transmissions given on the x-axis. The protocol for the relays is quite simple and assumes that after one transmission of the source, each relay can also send a packet. We discuss more sophisticated schemes later. The four transmission schemes differ in the field size used and the option to recode or not at the relays. Without recoding at the relays, a mean number of 102 and 105 transmissions by the source are needed to satisfy the destination for small ($GF(2)$) and large fields ($GF(2^8)$), respectively. Therefore, the impact on the field size is rather small for this scenario. However, a larger gain can be seen if we allow recoding at the relays. In fact, 26 fewer transmissions are needed from the source. The reason behind this gain is that recoding diminishes the probability that both relays convey the same information to the destination. Even if no new packets arrive from the source due to losses, the relays are able to send new coded packets to the benefit of the receiver.

Another advantage of network coding is that it is able to code on the fly. While other end-to-end codes need all packets of a generation in order to be able to encode packets, network coding performs the coding of packets as they arrive. In fact, it does not need to define a specific generation size [6]. In the case of video streaming with an inter-packet arrival time of 40 ms, an end-to-end code would need to wait 1 s in order to encode 25 packets and start to send out the first encoded packet. Therefore, coding is erroneously assumed to be delay afflicted. With network coding, each incoming (coded) packet can be coded at the source (recoded at the relays) with previously received/stored (coded) packets, and transmission can happen immediately. Clearly, the first packet would go out uncoded, but the second packet would allow us to create a coded version of packet one, two, and so on.

These recoding advantages and on-the-fly coding functionalities make network coding well suited for D2D communication as it will be able to adapt locally and efficiently to the dynamics resulting from the use of mobile devices.

NETWORK CODING CONCEPTS FOR D2D

D2D COOPERATIVE DOWNLOAD

Figure 1 shows two devices connected to the cellular network, which can simultaneously communicate directly with each other using D2D communication technologies (e.g., Wi-Fi, Bluetooth). If the two devices are interested in receiving the same data, each device can receive a fraction of the data from the cellular network to later exchange them with its partner. If a single user is interested in the data, a cooperating device could help speed up the download by using its own resources. Compared to the standalone case, a cooperative download scenario could reduce the download time and potentially energy savings due to the use of low-power and faster links for direct communication. Network coding can help reduce signaling and reduce the number of duplicate packets.

D2D NETWORK CODING FOR UNICAST FLOWS: RETROFITTING CLASSICAL MESH PROTOCOLS

A key issue in D2D communications is to enable reliable and efficient mechanisms for unicast flows between devices. Although network coding protocols have usually developed novel routing approaches (e.g., performing subgraph selection), this raises an important practical question: do we need to redesign our mesh network protocols to reap the benefits of network coding? Interestingly, this is not the case. We focus now on two key mechanisms that provide the following features.

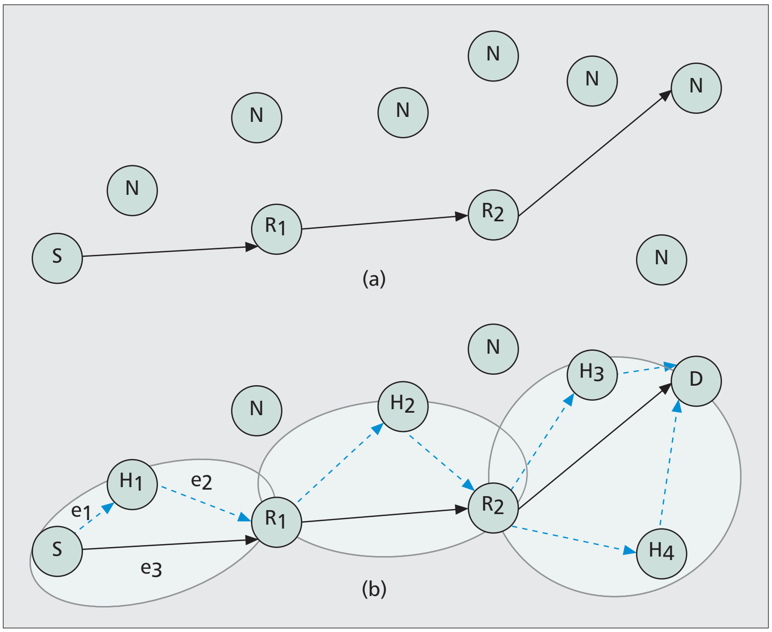

Figure 4. PlayNCool: a) the routing algorithm selects a path; b) the PlayNCool mechanism selects helper nodes for each link and activates them based on channel conditions and active neighbors.

Ad hoc route selection: Each unicast session determines its route(s) independent of other active sessions, which allows the routes to be selected by classical routing protocols.

Local optimization: Exploit local information of link qualities and/or other network flows to increase throughput or reduce energy costs locally with potential benefits end to end.

RLNC recoding at intermediate nodes: Allows for more efficient use of the wireless spectrum and increased impact of each transmission from the network nodes. This inherently provides a link-level erasure correcting capability that operates on the fly and without the cost of either decoding on a per link basis or, on the other end of the spectrum, provide end-to-end erasure protection, which is highly inefficient in multihop scenarios.

PlayNCool — Although previous proposals (e.g., MORE [7]) exploit the benefits of using multiple routes and network coding for additional reliability and throughput performance, these proposals typically rely in customized routing protocols, which are not compatible with existing ones. This becomes a barrier for the short-term deployment of network coding in real systems. To address these issues, we propose the PlayNCool protocol, which can operate in a variety of current routing protocols. The key ideas of PlayNCool are to:

- Exploit the path selected by a routing protocol (Fig. 4a)
- Use local link quality information to appropriately select local helper nodes that can

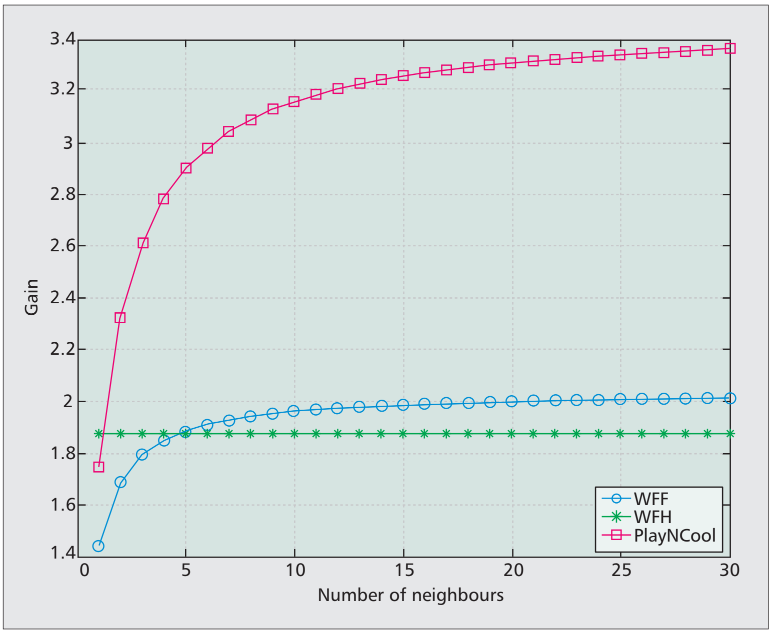

Figure 5. Gains of using helper nodes in a single link with respect to using only the direct link in the presence of different numbers of active neighboring nodes in the network. PlayNCool is compared to more naive approaches. WFF activates the helper once it has enough linear combinations to decode and stops transmissions from the source. WFH activates the helper when the helper and destination jointly have enough linear combinations to decode, but transmissions of the source are still maintained. Loss probabilities between source and helper, helper and destination, and source and destination are 0.3, 0.5, and 0.8, respectively.

improve reliability and throughput on each individual link as well as to determine when and how much to use the helper nodes (Fig. 4b)

Protocols such as B.A.T.M.A.N. [8] provide this link quality information directly, while other protocols may require some link quality estimation based on past data transmissions, which is done at the PlayNCool layer using the link quality discovery functionality. PlayNCool sends coded packets to the next hop and limits the use of the selected helper as the key to guaranteeing good performance. In a sense, the helper will *play it cool* and not transmit immediately. It will wait to gather a certain number of coded packets, determined by the channel conditions and competition from neighboring nodes, before the helper starts to transmit recoded packets from its buffer. The reason is that waiting to accumulate more coded packets at the helper allows for its transmissions to be linearly independent from packets already conveyed by the source at the next-hop receiver. Since the relay and the helper are competing for the same wireless medium, it guarantees that the helper will maximize its impact over the receiver.

The PlayNCool heuristics [9] showed that fourfold gains are possible depending on channel conditions and, more important, on the presence of competing flows in a network with a fair medium access control (MAC) mechanism (e.g., WiFi). Figure 5 presents the gains of PlayNCool and other (more naive) schemes that use helper nodes with respect to using only the direct link. In particular, Wait For Full (WFF) activates the helper once it has enough linear combinations to decode and stops transmissions from the source. Wait For Half (WFH) activates the helper when the helper and destination jointly have enough linear combinations to decode, but transmissions of the source are still maintained. Although the schemes show significant gains for all cases, increasing the number of competing nodes provides higher gains for PlayNCool. This is in part explained by the fact that activating both the helper and the sender for some period of time naturally gives that communication link a higher priority in the presence of other active transmitters. Since the transmission is finished in considerably less time, the additional interference introduced by the helper during a fraction of the source's transmission time pays off in the end for the entire network. If the network as a whole is using this idea, advantages are still expected because not all helpers are active simultaneously.

Interestingly, our results suggest that the presence of competing neighbors also increases the region where the helper is useful beyond the limits studied in [10]. Although [10] showed that cases where the loss probability between the helper and the receiver is higher than the loss probability between the source and the receiver, this result did not consider the presence of neighbors. Future work should consider a mathematical characterization of this problem beyond the heuristics proposed in [9].

CORE — CORE enables smart D2D communication by providing a mechanism to combine intra- and inter-flow network coding. In contrast to previous work such as [11, 12], CORE does not treat intra- and inter-flow network coding as coexisting but separate coding mechanisms. Instead, CORE proposes a more refined coding mechanism where the RLNC structure created for each individual flow is exploited to improve the spectral efficiency of inter-flow coding performed in specific regions within the network [13].

CORE then identifies relays where two or more unicast flows intersect, forming a potential coding region. Within these coding regions, CORE performs inter-flow network coding to provide greater spectral efficiency in the region by reducing the number of transmissions from the relay. If new coded packets from each flow have been received, the relay proceeds to XOR the content of these RLNC packets. CORE remaps the coding coefficients so that the original coefficients of each flow are preserved, and the receivers can exploit this information. In other words, the coded payloads of each flow are recoded with the other flow, while the encoding vectors for different flows are simply appended. Similar to COPE [14], if a receiver in the coding region overhears a packet and receives an XORed packet containing the overheard packet, the receiver will be able to recover the intended data packet from its own flow. This can be done since XORing packets in this way is equivalent to having coding coefficients of 1 for any finite field of the form $GF(2^k)$. Once transmitted, the relay stores the coded packets of each flow to be used later. To guarantee XORing of the packets, the relay defines an appropriate holding time that maintains new packets unsent until a new one is received from the other flow. If it is exceeded, the relay does not forward the packet as COPE would do. Instead, the relay will XOR the novel packet with a recoded packet from the other flow using the RLNC packets stored in its buffer. This provides a way to compensate possible losses of the other flow while still transmitting the intended, new packet. In other words, a CORE relay can implement recoding, which is a standard feature of RLNC that provides additional throughput and robustness against losses. Another design feature of CORE is the fact that the relay does not retransmit packets, instead sending a new coded packet in every transmission, thus providing additional benefits in spectral efficiency, as shown in [13].

Receivers in CORE exploit the structure of RLNC to speed up packet recovery and attain higher robustness against losses, particularly due to unsuccessful packet overhearing. Receivers accomplish this by not only overhearing transmissions corresponding to other flows to be used immediately, but store them for later use. Since CORE creates linear combinations of the RLNC packets of two flows, but it is meant to separate the flows before they leave the coding region, receivers may need to perform some degree of decoding. In the worst case, merging coded packets from generations of two flows requires the destinations to decode a generation twice as large as the intended one in the worst case. However, [13] showed that only a partial decoding is generally needed, if any.

As an example, Fig. 6a shows the case of an X topology as part of a coding region in a wireless mesh network with two unicast flows going from S_1 to D_1 and from S_2 to D_2. The former attempts to convey packets $P_1^{(1)}$, $P_2^{(1)}$, $P_3^{(1)}$ while the latter attempts to transmit packets $P_1^{(2)}$, $P_2^{(2)}$, and $P_3^{(2)}$. As part of the CORE protocol, D_2 and D_1 will overhear transmissions from S_1 and S_2, respectively. The first coding coefficient matrix shows the contents of D_1 after each source and the relay have transmitted four times each. The relay transmits an XOR of packets of the content of the two flows, but preserving the coding coefficients intact. The first two rows of Fig. 6b illustrate the case where D_1 was unable to overhear a transmission from S_2 and thus is unable to eliminate the contribution of such a packet from the XORed packet coming from the relay. The following rows show D_1 overhearing S_2 correctly and canceling out the effect on XORed packets from the relay. The key difference from previous approaches is that we do not drop packets overheard from D_1 after recovering an intended (coded) packet, and we do not discard XORed packets from the relay that are not immediately useful. A COPE-like technique would be left with two linear combinations of three variables at this point, while CORE already has six linear combinations of six variables. Figure 6c shows that CORE can perform partial decoding to free a new linear combination of packets belonging only to the intended flow (i.e., S_1 to D_1). Note that if D_1 is only an intermediate node in the network, it need not decode packets $P_1^{(1)}$, $P_2^{(1)}$, $P_3^{(1)}$, but only use them to generate recoded packets for the next hop. CORE can implement simple signaling mechanisms where the receivers in a coding region request more coded packets to the relay. The relay can also signal the sources to transmit additional coded packets to the relay. The relay will signal a source to stop transmissions if it has received enough coded packets to decode. Of course, the relay has no intention to decode, but signaling the node upstream to stop transmitting reduces overall transmissions and collisions in the coding region. A benefit of RLNC is that the relay may send packet requests to the sources and only specify the amount of desired packets, without knowing the specific packets received.

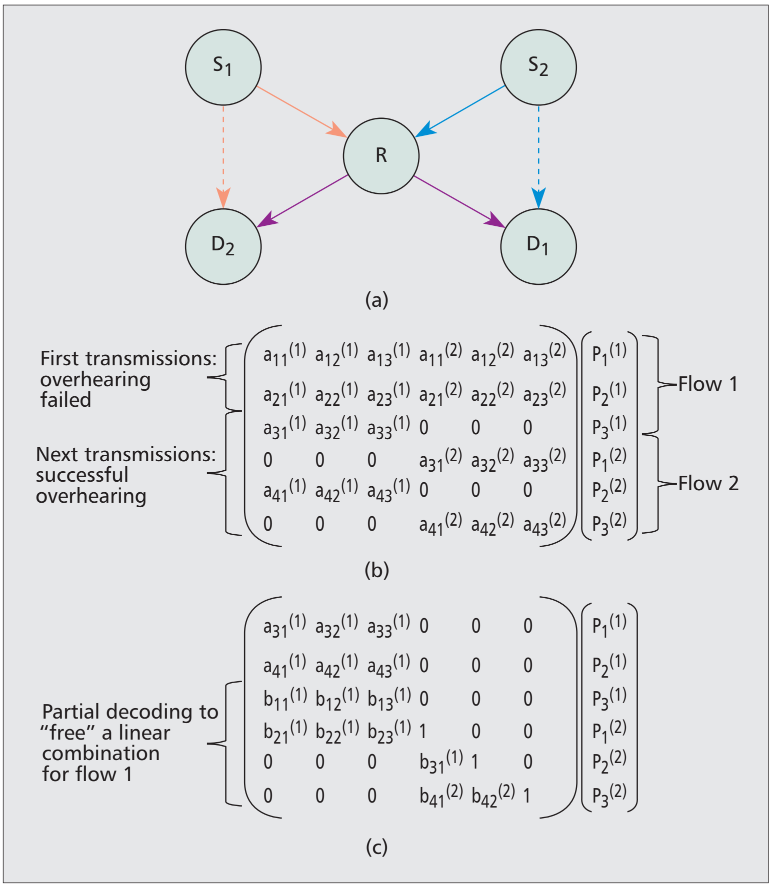

Figure 6. CORE: a) example of an inter-flow coding region based on an X topology in CORE; b) coding coefficient matrix at receiver D_1 in the coding region after some transmissions; c) partial decoding step to use RLNC to recover additional linear combinations of only flow 1.

Figure 7 shows the normalized throughput of four different schemes under various loss probabilities (erasure rates). This figure compares the performance of a full CORE scheme with feedback and recoding capabilities at the relay, and a CORE scheme with no recoding and more limited feedback with state-of-the-art forwarding and a COPE-like scheme with measurements on a real testbed. More details of the setup are provided in [15]. The two CORE schemes outperform forwarding and COPE for the entire range, but particularly for high packet loss rates, showing that COPE is highly dependent on the quality of the overhearing channels. Figure 7 also provides the optimal normalized throughput in this setting. Using RLNC arguments for the case

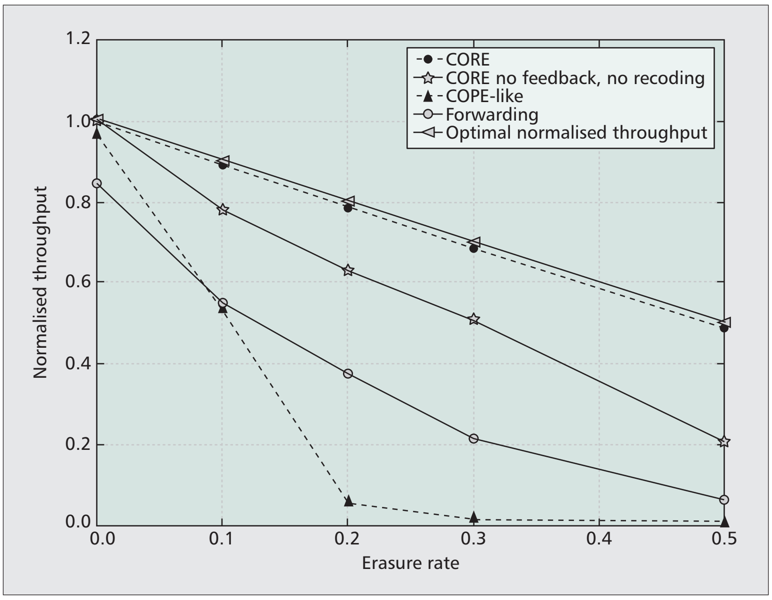

Figure 7. Throughput measurements of different schemes as a function of the erasure rate and the optimal normalized throughput. The throughput is normalized to the lossless throughput of CORE [15].

of symmetric channel losses, the maximum normalized throughput, R, is given by $R = 1 - e$, for e is the packet loss rate of each channel. This shows that CORE follows the same trend as an optimal scheme.

CHALLENGES IN D2D COMMUNICATION

HETEROGENEOUS DEVICES

One challenge in the D2D context is to find solutions for devices with inherently heterogeneous capabilities. In particular, mobile devices have different display sizes, computational power, and wireless data rates. If performing cooperative download of multimedia content with heterogeneous devices, the high-end devices is likely to need more data than a low-end device in order to satisfy higher video quality requirements. For video, multiple description coding (MDC) or scalable video coding (SVC) provide a first approach to deal with this heterogeneity. For example, SVC offers a video stream in several layers, with the base layer needed at all devices. High-end devices additionally receive multiple enhancement layers to increase the video quality/resolution. Nevertheless, efficient protocols have yet to be realized, and network coding can play an interesting part by using specific code structures to add flexibility for the end devices to wield their heterogeneous computational capabilities at the time of decoding.

SOCIAL INTERACTION OF EGOISTICALLY DRIVEN USERS

In our discussion, we assumed that all devices are willingly participating in D2D clusters. But for some services, this might not be true, and incentives are needed to drive a mobile device user to cooperate. In fact, a user may cooperate for several reasons. For example, the user might see his/her own benefit in cooperation, the user might get rewards over social networks, or the user is forced by the network operator as in [3]. Understanding the type of cooperation is key to developing efficient protocols because it affects the dynamics of the overall system.

SECURITY

One recurring argument against D2D communication is driven by security concerns. For example, downloading information via third party devices might be more risky than the state-of-the-art cellular communication. Besides several mutual security protocols (e.g., UMTS-AKA), network coding can provide interesting alternatives. For example, if part of the information is coming from the cellular operator and other packets via D2D communication, an attacker may not be able to decode the information. But even if the attacker can overhear all packets, network coding security mechanisms usually rely on encryption of the coding coefficients, requiring the attacker to break the encryption to then decode the data packets. Additionally, the presence of the cellular network might provide a simple mechanism to determine that decoded data is correct (i.e., not corrupted by an active attacker), as the cellular network can provide a small part of the content and also check on the final data, for example, the hash of a given file.

CONCLUSIONS

This article advocates for the use of network coding in smart D2D communications, particularly in scenarios with multiple mobile devices and multiple hops. Network coding's advantages over other codes, especially the ability to recode coded packets on the fly and/or using online coding techniques (i.e., sliding window coding), set it in a unique position to improve a D2D cluster's throughput, delay, and energy efficiency as well as providing additional security mechanisms.

This article presents two key approaches to improve communication efficiency, PlayNCool and CORE. These approaches not only exploit network coding's recoding potential but are also designed to be compatible with existing routing techniques. PlayNCool constitutes a new protocol family for relays using network coding. At the crux of this approach lies the idea that relays should collect coded packets and only send recoded versions after a silent period to increase the efficiency of its transmissions, while requiring minimal signaling overhead. Even in a simple relay topology, gains can be up to fourfold depending on channel conditions and the presence of active neighboring devices.

CORE is a conceptually new network coding approach, which combines inter- and intra-flow network coding in a nontrivial way. CORE is particularly suited for networks with moderate to high losses, where state-of-the-art mechanisms are considerably less robust. In fact, the article shows real implementation results, where CORE outperforms reliable store-and-forward as well as inter-flow mechanisms many times over, and even an order of magnitude for the case of high losses.

REFERENCES

[1] F. Fitzek, M. Katz, and Q. Zhang, "Cellular Controlled Short-Range Communication for Cooperative P2P Networking," *Wireless World Research Forum 17*, vol. WG 5, Nov. 2006.

[2] F. Fitzek and M. Katz, Cooperation in Wireless Networks: Principles and Applications — Real Egoistic Behavior Is to Cooperate!, Springer, Apr. 2006.

[3] F. Fitzek *et al.*, "Implementation of Network Coding for Social Mobile Clouds," *IEEE Sig. Proc. Mag.*, vol. 30, no. 1, Jan. 2013, pp. 159–64.

[4] P. Vingelmann et al., "Synchronized Multimedia Streaming on the iPhone Platform with Network Coding," *IEEE Commun. Mag.*, Consumer Communications and Networking Series, vol. 49, no. 6, June 2011, pp. 126–32.

[5] R. Ahlswede *et al.*, "Network Information Flow," *IEEE Trans. Info. Theory*, vol. 46, no. 4, 2000.

[6] J. K. Sundararajan, D. Shah, and M. Medard, "ARQ for Network Coding," *IEEE. Int'l. Symp. Info. Theory*, 2008, pp. 1651–55.

[7] S. Chachulski *et al.*, "Trading Structure for Randomness in Wireless Opportunistic Routing," *Proc. 2007 Conf. Applications, Technologies, Architectures, and Protocols for Computer Communications*, pp. 169–80; http://doi.acm.org/10.1145/1282380.1282400.

[8] Freifunk, B.A.T.M.A.N. Advanced Documentation Overview, http://www.open-mesh.org/.

[9] P. Pahlevani *et al.*, "PlayNCool: Opportunistic Network Coding for Local Optimization of Routing in Wireless Mesh Networks," *IEEE GLOBECOM Wksps.*, Dec. 2013.

[10] X. Shi, M. Medard, and D. E. Lucani, "Whether and Where to Code in the Wireless Packet Erasure Relay Channel," *IEEE JSAC*, vol. 31, no. 8, 2013, pp. 1379–89.

[11] X. Zhu, H. Yue, and Y. Wang, "C and M: A New Network Coding Scheme for Wireless Networks," *5th Int'l. Conf. Info. Assurance and Security*, vol. 2, Aug. 2009, pp. 432–36.

[12] H. Seferoglu, A. Markopoulou, and K. K. Ramakrishnan, "I2NC: Intra- and Inter-Session Network Coding for Unicast Flows in Wireless Networks," *Proc. IEEE INFOCOM*, Apr. 2011, pp. 1035–43.

[13] J. Krigslund *et al.*, "CORE: COPE with MORE in Wireless Meshed Networks," *IEEE VTC-Spring*, Dresden, Germany, June 2013.

[14] S. Katti *et al.*, "XORs in the Air: Practical Wireless Network Coding," *IEEE/ACM Trans. Net.*, vol. 16, no. 3, 2006, pp. 497–510.

[15] J. Hansen *et al.*, "Bridging Inter-Flow and Intra-Flow Network Coding for Video Applications: Testbed Description and Performance Evaluation," *IEEE Int'l. Wksp. Computer Aided Modeling Analysis and Design of Communication Links and Networks*, Berlin, Germany, Sept. 2013.

BIOGRAPHIES

PEYMAN PAHLEVANI (pep@es.aau.dk) is a Ph.D. student in the Department of Electronic Systems of Aalborg University, Denmark. He received his B.S. degree in information technology from the Institute for Advanced Studies in Basic Science (IASBS), Zanjan, Iran, in 2008. He also received his M.Sc. degree in computer networks from Yazd University in 2010. His research interests include wireless networks, and network coding theory and its applications.

MARTIN HUNDEBØLL (mhu@es.aau.dk) is a Ph.D. fellow in the Department of Electronic Systems of Aalborg University. He received his B.Sc. in computer engineering and his M.Sc. in networks and distributed systems in 2009 and 2011, respectively, both from Aalborg University. His research focuses on network coding in wireless networks, particularly on protocol design and evaluation.

MORTEN V. PEDERSEN (mvp@es.aau.dk) is a postdoctoral researcher in the Department of Electronic Systems of Aalborg University. He received his B.Sc in electronics engineering in 2007 and M.Sc. in wireless communication in 2009, both from Aalborg University, Denmark. In 2012 he received his Ph.D. in wireless communication from Aalborg University. He is the main author of the Network Coding software library Kodo, as well as a number of other scientific software libraries.

DANIEL E. LUCANI (del@es.aau.dk) is an associate professor in the Department of Electronic Systems, Aalborg University. He was an assistant professor at the University of Porto from 2010 to 2012 before joining Aalborg University. He received his B.S. and M.S. degrees in Electronics Engineering from Universidad Simón Bolívar, Venezuela, in 2005 and 2006, respectively, and his Ph.D. degree in electrical engineering from the Massachusetts Institute of Technology in 2010. His research focuses on communications, network theory, and network coding theory and applications.

HASSAN CHARAF (charaf.hassan@aut.bme.hu) received his Ph.D. in 1998. He is an associate professor and fellow in the Department of Automation and Applied Informatics at the Budapest University of Technology and Economics. He is head of the IT group. As an outstanding figure in teaching, research, and development, he is in key positions at several organizations at the university. His research fields are software modeling and model processing, mobile platforms, distributed systems and cloud computing, and data technologies.

FRANK H. P. FITZEK (ff@es.aau.dk) is a professor in the Department of Electronic Systems, University of Aalborg, heading the Mobile Devices group. He received his diploma (Dipl.-Ing.) degree in electrical engineering from the RWTH-Aachen, Germany, in 1997, and his Ph.D. (Dr.-Ing.) in electrical engineering from the Technical University of Berlin, Germany, in 2002. He has received numerous awards, including the NOKIA Champion Award five times, the NOKIA Achievement Award (2008), the Danish SAPERE AUDE research grant (2010), and the Vodafone Innovation prize (2012). His research focuses on wireless and mobile networks, mobile phone programming, network coding, cross-layer and energy-efficient protocol design, and cooperative networking.

HAMIDREZA BAGHERI (hbagheri@ee.oulu.fi) is working toward his Ph.D. degree under the supervision of Prof. Marcos Katz in the Department of Communication Engineering, University of Oulu, Finland. His current research interests include mobile clouds, cooperative and cognitive networks, and energy efficiency.

MARCOS KATZ (marcos.katz@ee.oulu.fi) received his M.S.E.E. and Dr.Tech. degrees from the University of Oulu. He worked at Nokia (1987–2001), Samsung Electronics (2003–2005), and the Technical Research Centre of Finland (2006–2009). Since 2010 he has been a professor at the Centre for Wireless Communications, University of Oulu. He is currently working on cooperative and cognitive communications, particularly in the area of mobile clouds. He is also working on visible light communications.

This article shows real implementation results, where CORE outperforms reliable store-and-forward as well as inter-flow mechanisms many times over, and even an order of magnitude for the case of high losses.