Efficient Distribution of Large Files in UMTS Supported by Network Coded M2M Data Transfer with Multiple Generations

Larissa N. Popova, Wolfgang H. Gerstacker, and Wolfgang Koch

Chair of Mobile Communications
University of Erlangen-Nuremberg, Germany
{popova,gersta,koch}@LNT.de

Abstract. This paper is a sequel of previous work, in which we have studied the traffic management problem in UMTS.

The main objective was to improve the spectral efficiency of cellular networks by employing network coding supported direct mobile-to-mobile communication on usually underloaded uplink channels for the distribution of a delay insensitive popular content in multicast manner. Simulations demonstrate the noticeable performance enhancement in terms of file download times and at the same time a substantial reduction of the number of necessary transmissions in both uplink and downlink directions.

However, popular content is often too large to be processed with a straightforward realization of network coding; combining the entire file to construct a single encoded block is impractical due to high encoding and decoding costs. To circumvent this problem, a novel concept is proposed in this paper. The scheme is based on processing popular content of virtually arbitrary size by embedding multiple generations into network coding.

Keywords: improvement in information distribution, direct mobile-to-mobile communication, network coding, multiple generations.

1 Related Previous Work

The aim of the analysis performed in the previous paper [1] was to improve the interaction between cellular and peer-to-peer networks by generalizing the traditional scheduling paradigm. A network coding technique [2,3] was embedded as a solution to the scheduling problem in the distributed dynamic environment of wireless large-scale networks. We have investigated the performance of the Universal Mobile Telecommunication System (UMTS) in terms of improvement in information distribution and dependability of information distribution among users by using direct mobile-to-mobile (m2m) data transmission mode. We have shown how, and in which terms network coding can thereby help. We have been especially interested in the distribution of the file download times for m2m users in different states of the download progress.

J. Zheng et al. (Eds.): ADHOCNETS 2009, LNICST 28, pp. 129–143, 2010.

Numerical results reveal the following findings:

- The NC-m2m (network coding – mobile-to-mobile) technique outperforms the performance of the replicate-and-forward m2m solutions proposed in [4,5], obviating the need for proper scheduling for non-real time file sharing applications.
- The proposed NC technique requires only local neighborhood information to perform the cooperative data transfer and therefore avoids flooding the network with signaling information.
- NC-m2m file dissemination outperforms even most-utile m2m algorithm, which is supported with perfect knowledge about the network topology [6].
- Simulations demonstrate the enhanced performance of the file distribution in terms of file download time. Furthermore, the obtained results highlight that network coding based m2m data transfer allows distribution of popular files to a large number of users while placing minimal bandwidth requirements on the central server, reducing at the same time the number of redundant uplink transmissions.

Hence, network coding ensures optimal network resource utilization.

In many applications, however, the files to be distributed are too large to be processed with the straightforward realization of network coding. Combining the entire file to construct a single encoded block is impractical due to high encoding and decoding costs not compatible with limited processing capabilities of mobile devices.

Furthermore, the amount of data on which a mobile terminal (MT) can perform coding operations is constrained by the only small-to-moderate storage available at a mobile device.

In this paper we discuss how an efficient NC-m2m file dissemination can be achieved even with strong memory constraints. For obtaining the numerical results the simulation experiments were conducted using a MATLAB based Radio Network Simulator (RANSim) tool.

2 Enhanced Network Coding for Operation on Data of Arbitrary Size

For clarity we first briefly recapitulate the NC based m2m file dissemination policy used in our previous work.

Mobile terminals voluntarily participate in file sharing via direct mobile-to-mobile data transfer with the purpose to reconstruct the original popular content. In each step, the respective user sends the linear combination of all packets he has received so far.

The following *Initialization* procedure is performed:

Each active m2m user creates two zero matrices $\mathbf{X} = \mathbf{0}_{m,w}$, $\mathbf{G} = \mathbf{0}_{m,m}$, where m is the number of logical packets, the source file is divided into, and w is the length of a logical packet in number of Bytes. $\mathbf{X}$ will be called the information matrix or decoding matrix. It stores the received encoded data (vectors) and is used for decoding later on. We call $\mathbf{G}$ encoding matrix. In this matrix the corresponding coefficient vectors will be stored.

The rank ν of $\mathbf{G}$ is initialized with zero.

Initially, no user has any information (packets) of the popular content; the logical packets are available in the core network only. The original packets are periodically distributed by the Base Station (BS) packet by packet to active m2m users in order to generate one complete copy of the content within a radio cell.

A user which received a packet from the BS acts as a server for the obtained packets. The packet transfer is performed on the user's own, currently not used uplink carrier frequency and receivers switch to listen on the uplink.

To avoid collisions on the uplink channels and to minimize the interference among m2m users, the MTs are organized into multiple groups, based on their locations and radio propagation conditions. The group organization policy, as well as the file transfer policy are performed as described in [6,1].

The following algorithm is executed:

Step 1. The packet $\mathbf{d}_j$ with elements $d_{j,l} \quad (j \in \{1, \ldots, m\}, \quad l \in \{1, \ldots, w\})$. received by a MT from its BS is stored in the first all-zero row $\nu+1$ of $\mathbf{X}$ ($X_{\nu+1,l} = d_{j,l}$). The corresponding row $\nu + 1$ in $\mathbf{G}$ ($G_{\nu+1,j} = g_j, \quad j \in \{1, \ldots, m\}$) will be a unit vector with the 1 at the ith position, which means that the currently received information vector is not encoded.

Step 2. An MT, which is entitled to send in the current frame[1], uses the information stored in $\mathbf{X}$ to create a new encoded packet $\mathbf{x}'$. For results shown in this work all operations are done in the finite field $\mathbb{F}_{2^8}$. In general, for the network under consideration the field size must be sufficiently large [3].

The MT-sender has to execute the following steps:

– Create a vector $\mathbf{g}$ of uniformly distributed random coefficients g_i, where

$$i \in \{1, \ldots, \nu\} \text{ and } g_i \in \mathbb{F}_{2^8}$$

We call it local encoding vector.

– Compute the new encoded vector, which we call encoded packet,

$$\mathbf{x}' = \sum_{k=1}^{\nu} g_k \cdot \mathbf{x}_k$$

where $\mathbf{x}_k$ is the kth row of matrix $\mathbf{X}$.

The encoding is done separately for each Byte position in a packet, using the same encoding vector.

– $\mathbf{x}'$ is broadcast to the group members together with the corresponding encoding vector $\mathbf{g}'_{\nu+1}$, where

$$\mathbf{g}'_{\nu+1} = g_1 \cdot \mathbf{g}'_1 + g_2 \cdot \mathbf{g}'_2 + \ldots + g_\nu \cdot \mathbf{g}'_\nu.$$

[1] 1 UMTS radio frame=0.01 sec.

We call $\mathbf{g}'$ the global encoding vector, it represents the encoded packets in terms of the source packets.

The MT-receiver(s) has to execute the following steps:

Step 3. If a packet $\mathbf{x}'$ has been received by a MT, it is stored in the row $\nu + 1$ of $\mathbf{X}$, and the corresponding encoding vector $\mathbf{g}'$ is stored in $\mathbf{G}$. This vector is also included in each coded packet's header, as a tag. The overhead this incurs is negligible if packet size ($m \log_2 q$ bits) is sufficiently large, where q is the field size. The benefit of the tag is profound, this gives the ability to be completely decentralized: receivers can compute $\mathbf{G}$ and decode without knowledge about network topology.

$\mathbf{G}$ is transformed to lower triangular form, using Gaussian elimination. If the received packet was innovative, the number of non-zero rows in $\mathbf{G}$ and therefore ν will increase by one.

If $\nu = m$ ($\mathbf{G}$ has full rank) the user has collected enough linearly independent combinations to completely reconstruct the original file. Thus, file reconstruction can be accomplished by simply solving a system of linear equations.

The matrix $\mathbf{G}$ is invertible with high probability, since all of the coefficients of all the local encoding vectors in the network are chosen randomly, independently, and uniformly from the sufficiently large field $\mathbb{F}$. The field size, used for investigation in this work (2^8) is sufficient, as it has been proven in [7] and [8].

Step 4. Users, which have not found any useful packets within a specified time interval, try to connect to the BS for packet delivery.

Additionally, our elimination routine outputs the IDs of all packets $\mathbf{d}_j$, which are guaranteed to increase ν and therefore can be ordered in Step 4.

All encoding/decoding operations are performed in the same way for each m2m user.

2.1 Generations: Optimized Packet Combination

In order to operate on files of virtually arbitrary size the above described network coding technique must be modified (for a more detailed description of NC-based m2m file dissemination policy see [1].

One way to achieve this is to organize the packets of the information source (popular file) into groups of packets, called generations with f packets per generation, where f will be the generation size. It determines the size of matrices the receiver needs to buffer, and later invert to decode the information. Hence, there exists one matrix per generation and only packets of the same generation will be combined (encoded). To keep track of packets in the same generation, each packet is tagged with its generation number z (one byte is sufficient for this purpose), where $z \in \{1, \ldots, u\}$, and u is the total number of generations the file is divided into (see also [9,10]).

Figure 1 demonstrates the basic concept of handling generations in NC based m2m file dissemination mode.

- Packets arrive in arbitrary order at a MT.
- If the MT is entitled to send in the current frame it generates a new encoded packet as a linear combination from packets tagged by generation number one, or number two, or number x, depending on the generation distribution strategy (GDS), which will be discussed in detail in the next section. Thus, the coding is performed only inside the generations. Packets are encoded symbol-wise.
- To trace the file download progress at each MT, each arriving packet is added to a generation buffer, according to its generation number. A generation number is appended to the packet headers. Gaussian elimination is then performed on those encoding matrices, that correspond to generation buffers that have new data available. This allows to store the encoding matrices in row reduced echelon form and to check whether a new packet is innovative immediately after its reception.
- A packet received by a MT is innovative if it increases the rank of the set of the encoding matrices of packets at this MT.
- Non-innovative packets will be ignored.
- The BS supports MTs with the signaling information and allows only one MT within the group to transmit in a given time interval, avoiding collisions within a particular group.

2.2 Analysis of Relationship between Different Figures of Merit

To obtain an extensive statistical characterization of the performance of m2m data dissemination in the UMTS system supported by network coding with multiple generations (NC-MG), we have conducted a substantial number of simulations with different parameter settings. In the paper, however, we outline only some of them. Our work distinguishes itself from other work in this area by considering a wide variety of aspects in the analysis.

The major goal is to analyze the trade-off between file size, size of generations, memory requirements, computational complexity and Quality of Service (QoS) like transmission delay, etc. System performance for generation-based NC-m2m data dissemination is then compared with that of simple straightforward network coding in terms of the above mentioned characteristics.

Memory Requirements and Computational Complexity. We start with an outline of some fundamental properties of multi-generation NC, namely, computational complexity and memory requirement at the MT.

As mentioned earlier one of the major issues that emerge in a system which applies network coding is extensive usage of quite limited memory resources in mobile devices.

Evidently, large files must be split into a larger number of packets, which, by applying straightforward NC (with linear combining of all packets available at the MT), results in an $m \times m$ decoding matrix (where m is the number of packets

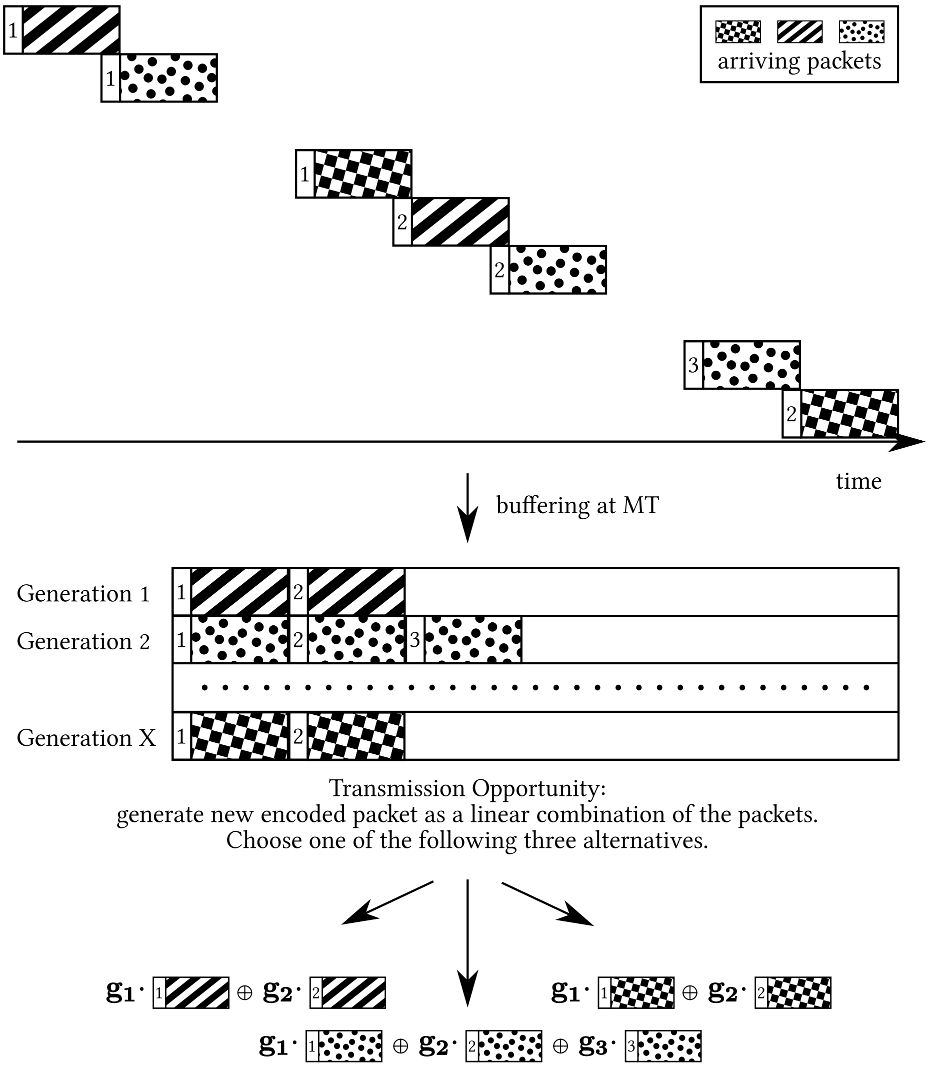


Fig. 1. The basic concept of handling generations for the network coding based m2m file dissemination scheme

in the file). Thus, the effective memory required in each MT increases with $\mathcal{O}(m^2)$, which leads to a quick memory exhaustion of mobile devices. Because

$$\left(\frac{m}{u}\right)^2 \cdot u = \frac{m^2}{u} \leq m^2, \quad \text{with} \quad 1 \leq u \leq m, \tag{1}$$

the arrangement of the packets into groups can reduce the memory complexity, if the number of generations u is chosen depending on the total file size m and the fixed generation size f. With $u \sim \frac{m}{f}$ memory complexity becomes $\mathcal{O}(m)$, which means that by applying generation-based NC, memory consumption increases only linearly with the file size.

The next figure of merit we would like to analyze is computational complexity. It is well known that the Gaussian elimination procedure has a complexity of $\mathcal{O}(m^3)$. In case of NC with multiple generations, assuming constant file size, a larger number of generations results in a smaller generation size and reduced decoding matrix size (within each generation), respectively. Thus, the computational complexity (complexity of encoding operations) is $\mathcal{O}(u \cdot f^3)$, where $u \cdot f = m$, which is lower than that Gaussian elimination in standard NC implementation.

Generation Distribution Strategy. We investigate and compare performance of file dissemination for three generation distribution strategies (GDSs): local rarest-first, sequential and random.

Each of the above mentioned strategies requires a different degree of knowledge about the network topology.

Local Rarest-First Generation Distribution Strategy (GDS): Realization of generation based NC-m2m data dissemination by using local rarest-first GDS requires explicit knowledge about users in each m2m group. The algorithm, which decides that packets of a certain generation will be encoded, must be supported with information regarding the packet dissemination progress throughout a group. For the local rarest-first case the packets from the most unpopular generation (local rarest) will be combined in order to generate a new encoded packet, which the MT-sender multicasts to its group members.

However, local neighborhood information (within a m2m group only) is sufficient to perform local rarest-first GDS-based cooperative data transfer.

Sequential GDS: In order to perform NC-m2m data transfer by using sequential GDS, even less information about neighboring MTs is needed.

Instead, the data exchange algorithm performed at the BS decides on the number of the next generation (in sequential order) to be sent in a particular m2m-group in the current time interval. Along with the generation number, the sender identification information is transmitted.

Random GDS: The sender chooses at random the generation number of the packets to be encoded and sent from the set of non-empty generations. No additional knowledge about packets existing in nearby MTs is necessary.

Note: For all distribution strategies the following rule is valid: If the generation to be sent in the specified time interval is empty the packets of some other (according to the generation distribution strategy currently in use) generation will be picked, coded and sent.

3 Numerical Results

We examine now some parameters of generation construction and their impact on the system performance. The performed simulations return following observations.

3.1 Generation Size and File Size

As mentioned before, the generation size has a significant impact on the performance of network coding. To achieve sufficient performance of the algorithm and at the same time assure reduced memory requirements and computational complexity, it is necessary to determine the optimal generation size for a given file size.

In this section we compare the performance of the NC-m2m file dissemination algorithm for different generation sizes. For the file size fixed at 40 kbyte we distinguish small (20 packets), medium (100 packets) and large (720 packets) generation sizes.

Simulations have been performed for the system with a high traffic load scenario using sequential GDS solely. All users were assumed to be Pedestrian. No arrivals of new m2m users are simulated.

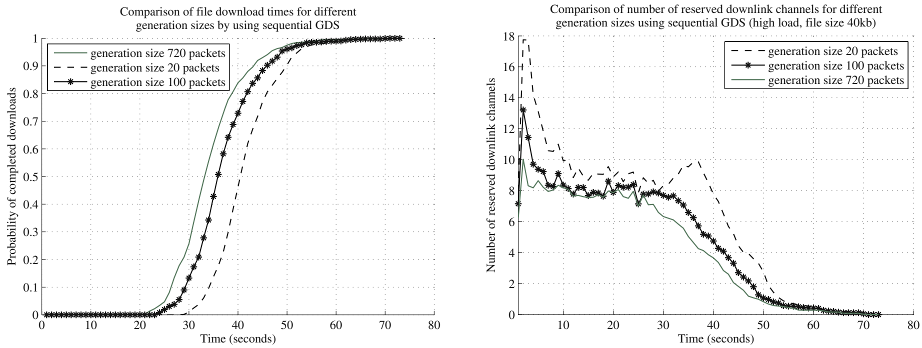
The most important parameters used in our simulations are summarized by Table 1.

Table 1. Main simulation parameters

Traffic and environmental settings	
Traf. load (max. num. of m2m users/cell)	50 (high)
Maximum m2m group size	13
Antenna type	omni-directional
Cell radius	50 m
User profile	Pedestrian
Radio interface and algorithm settings	
File size	500 kbyte
Required gross data rate	60 kbit/s
Maximum user data rate with 1/2-rate coding	30 kbit/s
Size of logical packets for m2m data	225 bit (coded)
Receiver sensitivity	−112 dBm
Transmission power in m2m mode	−44 dBm
E_b/N_0 target	3 dB
Inner loop power control for m2m sender	OFF
Simulation step size	1 radio frame (0.01 s)
Group update period	100 radio frames (1 s)

Fig. 2a shows the impact of the generation size on the file download time. As can be seen from the diagram, with an increasing number of generations (in turn, decreasing the generation size) we observe NC-m2m performance degradation.

This is obvious, since every time an m2m-sender picks some generation, according to the GDS, he creates an encoded packet (using the whole set of data stored therein) and sends it to its neighbors, together with the corresponding encoding vector. Some choices will be more beneficial for the group members, others will only provide them with little information. If only one generation is



(a) Comparison of file download times for different generation sizes and fixed 40 kbyte file size.

(b) Impact of different generation sizes on downlink resource consumption.

Fig. 2. Impact of generation size on the performance of NC-based m2m file dissemination in a high loaded UMTS system

present the choice is always trivially optimal. As the number of generations increases, the probability of choosing a generation for coding and transmitting packets that is of inferior use for the receivers (compared to the optimal choice) also increases, consequently, decreasing the benefits of NC.

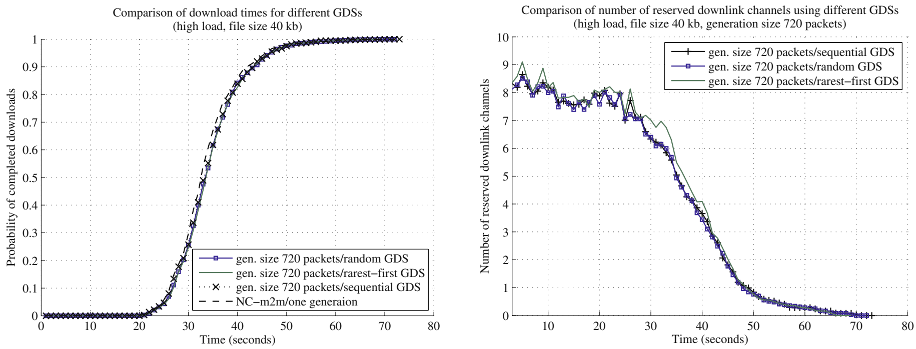
Such an unfavorable choice of generation size coerces m2m users to request packets of the desired file from the BS more frequently. This results in the allocation of additional downlink channels, thus demanding more of the cell's valuable downlink capacity and impairing the efficiency of the m2m concept, cf. Fig. 2b.

Furthermore, when the number of generations approaches the total number of packets, the performance of the NC-MG-m2m algorithm will converge to that of random replicate-and-forward m2m file dissemination. Thus, a single, large generation will always yield the best achievable performance; and it is desirable in the multiple-generation case to apply a fairly large generation size.

3.2 Quality of Service (QoS) Requirements

To analyze the Quality of Service (QoS) resulting for the proposed generation-based NC technique, we have implemented all of the above mentioned generation distribution strategies: local rarest-first, sequential and random. Among the figures of merit we are interested in the data rate degradation caused e. g. by unfavorably chosen GDS.

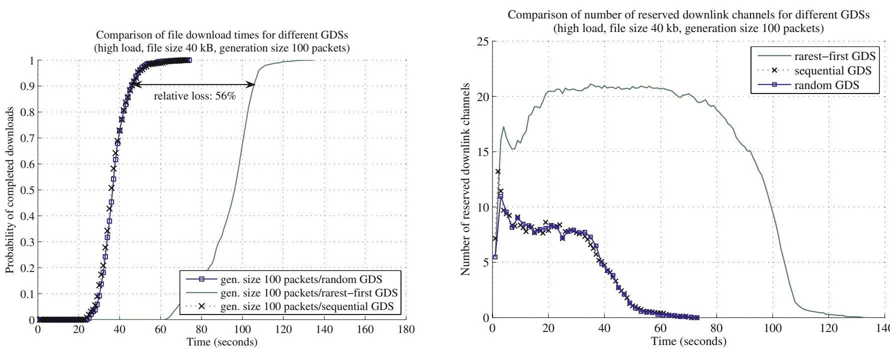
As one can see from Fig. 3a the performance of generation-based NC-m2m file dissemination is almost identical for all three generation distribution policies in terms of download time, with a given file size and a large generation size (two generations only). The diagram shows that all policies approach the performance of the optimal single generation case.



(a) Impact of generation distribution strategy on the file download time using a large generation size (file size 40 kbyte).

(b) Downlink resources consumption in NC-m2m mode using different GDSs and large generation size.

Fig. 3. Performance comparison of NC-MG m2m algorithm in high loaded system for different GDSs and fixed generation size



(a) Impact of generation distribution strategy on the file download time using generation size of 100 packets (file size 40 kbyte).

(b) Downlink resources consumption in NC-m2m mode using different GDSs and generation size of 100 packets.

Fig. 4. Performance comparison of NC-MG m2m algorithm in high loaded system for different GDSs and fixed generation size

The amount of consumed resources on the downlink is also almost the same for all three GDSs, as shown in Fig. 3b. However, with increased number of generations and unchanged file size, the rarest-first GDS performs significantly inferior to the other two policies, while the difference in performance between these two policies is still very minor, see Fig. 4a. These observations can be explained as follows.

The gap of 56 % in file download time is due to the fact that we assume a large number of generations and at the same time a relatively small file size. The performance of the rarest-first GDS, in terms of download time, then degrades, since the "rarest" generations always contain less packets at each MT than the remaining ones.

Thus, the linear combination of the packets currently available at a sender candidate formed from the rarest generation are less likely to be innovative for other group members. As a result the time interval between the reception of innovative packets at the MTs increases, reducing data rate and prolonging file download time, consequently.

Related observations can be made from Fig. 4b. A lack of novel packets, received by MTs in m2m mode, has to be compensated for. Users with expired waiting counters for new packets will request more data directly from their BS. This results in unavoidable occupation of additional downlink resources.

In contrast, the distribution of generations in random order prevents users from having a strict subset of the generations, which yields a better performance of the m2m algorithm.

The slight performance degradation of the random GDS compared to the sequential GDS is simple to explain. This is due to the fact that by using the random policy a generation can accidentally be chosen multiple times in a short time interval, thus triggering redundant data transmissions.

We concentrate our further investigations therefore on sequential GDS, which appears to be the most natural generation distribution strategy.

3.3 Comparison of NC-MG-m2m File Sharing with Replicate-and-Forward m2m Data Dissemination

In the following the system performance for network coded m2m data dissemination using multiple generations (NC-MG-m2m) is compared with that of the simple replicate-and-forward m2m schema, which has been analyzed in our previous paper [1], in terms of usage of network resources, duration of file download, as well as processing costs.

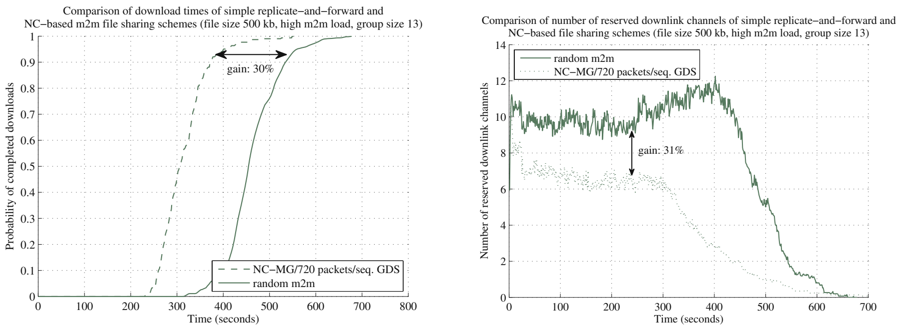
Now, by using generations, files of virtually arbitrary size can be processed. In the following we analyze the dissemination of a file of 500 kbyte size, using NC-MG-m2m data distribution strategy.

For this, we have divided the above mentioned file into 26 generations with 700 packets in each generation and distributed the popular content with the NC-MG-m2m algorithm using the sequential GDS.

In Figs. 5 and 6 the corresponding simulation outcome is compared to the performance results of the most-utile-m2m [6] and random-m2m [1] packet distribution schemes, as well as to the file dissemination performance of the conventional UMTS mode for the same scenario as in Section 3.2 with high traffic load.

The following observations can be made:

- Network coding with multiple generations (NC-MG) m2m data distribution outperforms the random m2m and conventional packet distribution significantly, in terms of both file download time and released downlink capacity.



(a) Comparison of file download times for NC-MG-m2m and random replicate-and forward m2m mode.

(b) Downlink resources consumption in NC-MG-m2m and random replicate-and forward m2m mode in high loaded system.

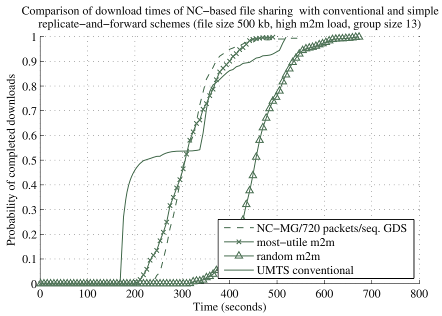
Fig. 5. Performance comparison of large file distribution using NC-MG-m2m and simple random m2m algorithm in a high loaded system (file size 500 kbyte)

We can observe a gain of up to 30 % at the 90 % quantile of completed downloads in the high loaded pure m2m system, compared to the random m2m file distribution, and up to 85 % compared to the conventional file transfer mode.

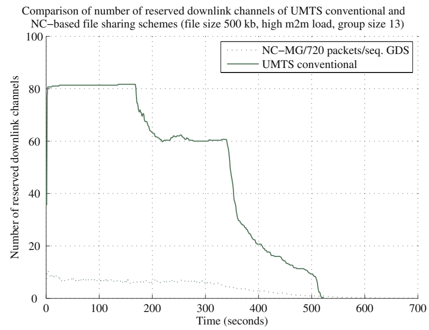
– NC-MG m2m data distribution algorithm reduces the number of (uplink) m2m transmissions by a factor of $\log m$ [11] (compared to random replicate-and-forward), which results e.g. in a reduced battery consumption at the MTs.
– In comparison to the most-utile m2m policy, NC-MG m2m achieves a slightly better performance.
– Although the gain is not significant, we save the amount of signaling information we would otherwise need to transmit in case of the most-utile m2m transfer to keep each MT updated with the knowledge about all packets that exist in its neighborhood.

Practical Issues. First, to support the proposed m2m system mode within the UMTS network from a practical point of view, namely to make MTs able to operate in both uplink/downlink frequency bands, some hardware changes will be necessary. Providing for MTs signal processing capabilities for receiving on the uplink frequencies is one possibility to realize the m2m algorithm. Even though RF hardware is not cheap and this solution would increase the cost for the mobile terminals at the moment, it is known that prices for electronic devices have been decreasing at an enormous pace ever since.

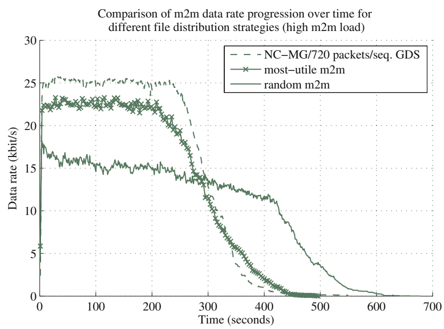
Second, the application of network coding entails further functionality extensions. However, nowadays the processing becomes cheaper, too. Furthermore, as we have seen from the simulation results, NC can be implemented at a lower computational cost, by using multiple generations.



(a) Comparison of file download times for NC-MG-m2m, conventional and replicate-and-forward (most-utile and random) m2m modes.



(b) Downlink resources consumption of NC-MG-m2m and conventional UMTS modes in high loaded system.



(c) Impact of NC-MG-m2m algorithm on the m2m data rate progression.

Fig. 6. Performance comparison of large file distribution in a high loaded system using NC-MG-m2m, simple random m2m, most-utile m2m and conventional UMTS network mode. Simple random m2m and most-utile m2m algorithms display similar behavior in terms of downlink resources consumption (compared to NC-MG-m2m) and therefore are not shown in Fig. 6b

4 Summary

In this paper we have conducted a performance evaluation and analysis of m2m data dissemination of files of virtually arbitrary size in the UMTS system supported by network coding with multiple generations (NC-MG).

In order to achieve a trade-off between memory requirements and computational complexity on the one side and sufficient performance characteristics of the NC-MG algorithm on the other side, we have investigated different generation distribution strategies (GDSs).

The following conclusions can be drawn from the numerical results:

– Size and composition of generations (distribution strategy) have a significant influence on the performance of the proposed file dissemination technique.
– The smaller the number of packets in a generation, the lower the benefits of NC.
– With appropriately chosen generation size the MT is able to hold several generations in memory and to operate on files of large size, consequently.
– By disseminating the popular content with the NC-MG-m2m algorithm using the sequential or random GDS, only simple operations on the generations without any extra coordination or special knowledge about the network topology are necessary.
– This means that the NC-MG-m2m algorithm provides performance benefits even in scenarios where the knowledge of the MTs regarding the state of other MTs is constrained; receivers can decode even if the network topology is unknown.

We have demonstrated that the NC-MG-m2m algorithm is a viable solution for an efficient and reliable dissemination of any media file or streams at low computational cost.

References

1. Popova, L., Schmidt, A., Gerstacker, W., Koch, W.: Network Coding Assisted Mobile-to-Mobile File Transfer. In: Proc. of IEEE Australasian Telecommunication Networks and Applications Conference (ATNAC), Christchurch, New Zealand (December 2007)
2. Ahlswede, R., Cai, N., Li, S.Y.R., Yeung, R.W.: Network information flow. IEEE Trans. Information Theory 46 (July 2000)
3. Fragouli, C., Le Boudec, J.Y., Widmer, J.: Network Coding: An Instant Primer. Technical report, LCA-Report-2005-010 (2005)
4. Popova, L., Herpel, T., Koch, W.: Enhanced Downlink Capacity in UMTS supported by Mobile-to-Mobile Data Transfer. In: Akyildiz, I.F., Sivakumar, R., Ekici, E., Oliveira, J.C.d., McNair, J. (eds.) NETWORKING 2007. LNCS, vol. 4479, pp. 522–534. Springer, Heidelberg (2007)
5. Popova, L., Herpel, T., Koch, W.: Efficiency and Dependability of Direct Mobile-to-Mobile Data Transfer for UMTS Downlink in Multi-Service Networks. In: Proc. of IEEE Wireless Communications & Networking Conference (WCNC), Hong Kong (March 2007)
6. Popova, L., Herpel, T., Gerstacker, W., Koch, W.: Cooperative Mobile-to-Mobile File Dissemination in Cellular Networks within a Unified Radio Interface. Regular Issue of Computer Networks Journal (2008)
7. Ho, T., Medard, M., Koetter, R., Karger, D.R., Effros, M., Shi, J., Leong, B.: A random linear network coding approach to multicast. IEEE Trans. of Information Theory 52 (October 2006)
8. Sander, P., Egner, S., Tolhuizen, L.: Polynomial time algorithms for network informaton flow. In: Proc. of Symp. Parallel Algorithms and Archtectures. ACM, San Diego (2003)

9. Chou, P.A., Wu, Y., Jain, K.: Practical network coding. In: Proc. of Allerton Conference on Communications, Control and Computing, Monticello, IL, USA (October 2003)
10. Widmer, J., Le Boudec, J.Y.: Network coding for efficient communication in extreme networks. In: Proc. of ACM SIGCOMM Workshop on Delay-tolerant networking, Philadelphia, Pennsylvania, USA, August 2005, pp. 284–291 (2005)
11. Fragouli, C., Soljanin, E.: Network Coding Applications. Now Publishers Inc. (January 2008)