On the Flooding Overhead of Fountain Codes in Wireless Sensor Networks

Anya Apavatjrut, Katia Jaffrès-Runser, Claire Goursaud, Cédric Lauradoux

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

On the Flooding Overhead of Fountain Codes in Wireless Sensor Networks

Anya Apavatjrut — Katia Jaffrès-Runser — Claire Goursaud — Cédric Lauradoux

N° 7397

September 2010

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--7397--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
GRENOBLE - RHÔNE-ALPES

On the Flooding Overhead of Fountain Codes in Wireless Sensor Networks

Anya Apavatjrut*, Katia Jaffrès-Runser* , Claire Goursaud* , Cédric Lauradoux*

Theme :
Équipes-Projets SWING

Rapport de recherche n° 7397 — September 2010 — 13 pages

Abstract: This paper concentrates on the proper use of fountain codes for the transmission of sporadic data in a wireless sensor network (WSN). Fountain codes offer great perspectives for the self-organization of WSNs: they self adapt to the channel error rate without any control data. When deploying fountain codes on a WSN, two problems arise. First, the size of the data transmitted by a sensor is small in comparison to the size considered traditionally with fountain codes. The analysis of the decoding overhead for fountain codes is often done for large data. Second, the communications are done in an hop-by-hop fashion. It implies that the destination of the data can not acknowledge instantaneously its reception to the source. Therefore, the transmissions of useless packets for the destination can not be prevented. The impact of this *flooding traffic* is analyzed. It depends on the data size k and on number of hops n between the source and the destination. Our work can be viewed as the networking counterpart of the results presented by *Pakzad and al.* at ISIT 2005 applied to WSNs. The context of our study is a line network, i.e. a cascade of n erasure channels. The flooding traffic has been evaluated as well through realistic simulations for three different relaying strategies where packets are lost due to both small scale fading and collisions for an unslotted IEEE 802.15.4 medium access layer.

Key-words: flooding overhead, fountain codes, acknowledgement, wireless sensor networks

* CITI laboratory, INSA de Lyon, France

Centre de recherche INRIA Grenoble – Rhône-Alpes
655, avenue de l'Europe, 38334 Montbonnot Saint Ismier
Téléphone : +33 4 76 61 52 00 — Télécopie +33 4 76 61 52 52

Résumé : Cet article propose l'optimisation de l'usage d'un code fontaine pour une transmission sporadique de données dans les réseaux de capteur. Dans ce contexte, ces codes présentent un avantage relatif à l'auto-organisation du réseau: son taux de transmission de données est variable et s'adapte intrinsèquement à la qualité du canal radio sans qu'il n'y ait besoin de paquets de contrôle supplémentaires. Cependant, pour pouvoir utiliser les codes fontaines dans les réseaux de capteurs, il est nécessaire de prendre en compte les contraintes suivantes : premièrement, le surcoût lié au décodage n'est plus négligeable lorsque la taille des données devient petite et deuxièmement, il existe un débordement du flux des paquets codés relatif à la transmission en mode multi-sauts de l'acquittement. En effet, si l'acquittement mets plus de temps à atteindre la source relativement à la vitesse de relayage des paquets codés par la source, certains relais ainsi que la source émettent un nombre de paquets inutiles au décodage avant d'arrêter la transmission. C'est principalement le cas pour une transmission en cascade multi-sauts pour laquelle le nombre de relais est élevé. Dans cet article, nous étudions ce surcoût qui varie avec la quantité de paquets à transmettre et le nombre de sauts entre la source et le destinataire. Ce surcoût de transmission est analysé pour plusieurs stratégies de relayage, tout d'abord analytiquement puis par simulation pour un protocole Zigbee IEEE 802.15.4.

Mots-clés : surcoût lié à l'acquittement du flux de transmission, codes fontaines, réseaux de capteurs, IEEE 802.15.4.

1 Introduction

This paper is dedicated to the deployment of fountain code in a wireless sensor network (WSN). A WSN is composed of *sensor nodes* with restricted capabilities (memory, energy and computational power) and a set of *sink nodes* which gather the sensed data. It is assumed here that the sink nodes have unlimited resources compared to the sensors. A direct communication between a source $\mathcal{S}$ (any sensor) and a destination $\mathcal{D}$ (any sink) is not always possible: relaying nodes $\mathcal{R}_i$ are used to carry the data following a hop-by-hop model. The cascade of n channels [13] forms the *main path* and the reverse path forms the *feedback path* (Fig. 1). Such a cascade is established by a routing algorithm whose study is out of the scope of this paper.

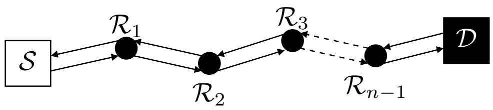

Figure 1: A main and feedback path with n hops.

In a WSN, the capacity of each channel in a path is unknown prior roll-out and even during the lifetime of the network: the nodes are usually scattered across the monitored area in a random way and the environment may vary. Fountain codes are a promising solution to provide robustness here [5, 14, 1, 6]. Indeed, they are *rateless*, *i.e.* a source $\mathcal{S}$ can potentially generate a limitless number of encoded packets until it receives an acknowledgement from $\mathcal{D}$. They can adapt to the channel on the fly. Another advantage of fountain codes over schemes such as automatic repeat request (ARQ) is the limited use of the feedback path. $\mathcal{D}$ only acknowledges a successful decoding to $\mathcal{S}$. This is why fountain codes have found important applications in satellite communications [4], content distribution [3] or underwater networking [5].

In order to take the full advantage of fountain codes in a WSN, it is mandatory to account for the characteristics of the data sent by $\mathcal{S}$ and of the feedback channel.

The data transmitted by a source $\mathcal{S}$ in a WSN depends on the target application. However, most of the time, sensors gather small amounts of data (*e.g.* temperature, light level or any physical quantity. . .) and send them at a regular or irregular pace to the sink $\mathcal{D}$. Hence, it is reasonable to consider that only few packets have to be sent at the same time, traffic is sporadic and data packets are short. This represents a major modification for the application of fountain codes. In previous applications [4, 3], large data are often assumed. The main impact of this change is the modification of the *decoding overhead* ϵ of most fountain codes (Raptor, LT. . .). Random linear codes are less affected by this modification and this is why they are used in this paper despite their high *decoding complexity*.

The feedback path from $\mathcal{D}$ to $\mathcal{S}$ does not allow to transmit instantaneously the acknowledgement to $\mathcal{S}$ at the end of the communication. While the acknowledgement is *en route* to $\mathcal{S}$, the fountain still flows, *i.e.* $\mathcal{S}$ and the $n-1$ relaying nodes $\mathcal{R}_i$ can still transmit data. These additional communications are wasting the nodes resources. We call them the *flooding traffic* by analogy to the amount of water wasted by a hosepipe before somebody in the garden is able to turn-off the tap.

The analysis of the flooding traffic depends on the size of the data to be transmitted k and on the number of hops n. Indeed, the number of packets required to reach a successful decoding at $\mathcal{D}$ is lower bounded by $n(k+\epsilon)$. A simple model based on the acknowledgement progression gives us a flooding traffic in $O(n^2)$ for certain relaying strategies. Therefore, the choices of n and k are critical to prevent the flooding traffic to become dominant. Our analysis is supported by realistic simulations where both small scale fading and collisions for an unslotted ZigBee IEEE 802.15.4 medium access layer [17] are considered using the WSNet network simulator [7].

The rest of the paper is organized as follows. In the next section, existing works on fountain codes are summarized. In Section 3, the flooding overhead problem due to the hop-by-hop acknowledgement transmission is analyzed and section 4 presents the impact of different relaying strategies on this overhead. Finally, Section 5 concludes the paper.

2 Fountain codes

The concept of fountain codes was first presented in 1998 by *Byers and al.* [4]. Since then, many works have been done on this topics. The design of a fountain code for a line network is composed of four tasks: (a) the *encoding algorithm design*, (b) the *decoding algorithm design* including the *decoding overhead* ϵ, (c) the *relaying strategy choice* and (d) the *flooding overhead analysis*. The first three tasks (a to c) are entwined and have been addressed in many works [3, 12, 16, 2, 9]. The design of efficient relaying strategies for fountain codes have recently deserved the attention of the communication community due to the development of network coding techniques [15, 1, 6]. To the knowledge of the authors, this paper is the first one addressing the flooding issue. In what follows, we briefly summarize tasks (a) to (c).

2.1 Fountain encoding

Given a set of k input symbols, the source generates an infinite stream of output symbols which results from a linear combination of a subset of input symbols. For each encoded symbol, a degree $d \in [1, k]$ is chosen according to a degree distribution. Then, d randomly chosen symbols are linearly combined over $\mathbb{F}_2$ to form the encoded symbol. This coding is repeated until an acknowledgement is received.

The degree distribution is the central parameter in the design of fountain codes that is linked to the decoding algorithm. The most simple distribution is the uniform one of random linear codes. Another possibility is Robust Soliton distribution associated to LT-code introduced by Luby [12]. A modification of the encoding scheme has been proposed by *Shokrollahi* with Raptor code [16]. A precode is applied on the k input symbols prior to the fountain encoding. To conclude on the degree distribution, the results [9, 18] are worth mentioning. They study the optimal degree distribution that must be associated to a Gaussian elimination decoding for small value of k.

2.2 Fountain decoding

A reliable decoding algorithm for Fountain codes should allow the recovering of k input symbols from any subset of $k+\epsilon$ output symbols with a probability that is at most inversely polynomial in k [16].

The straightforward decoding algorithm consists in solving the a system of $k+\epsilon$ equations using a Gaussian elimination. This approach has a low decoding overhead, *i.e.* ϵ is $O(\frac{\log(k)}{k})$ but a high decoding complexity paper 'raptor code' of Shokrollahi page 9. $O((k+O(\frac{\log(k)}{k}))k^2)$ Later, *Luby* applied to its LT-codes [12] *Belief Propagation* (BP) decoding. It offers a better decoding complexity at the cost of decoding overhead (see Table 1). Raptor code is an optimized version of LT code that combines an outer code to a regular LT code. Its decoding overhead is smaller than for LT ($\epsilon = 1$ or 2) while its decoding complexity is close to the one of an LT-code[1].

	Random Linear	LT	Raptor
Decoding Overhead	$\frac{\log(k)}{k}$	$\frac{\log^2(k)}{\sqrt{k}}$	see 1
Encoding Complexity	k^2	$k\log(k)$	see 1
Decoding Complexity	$(k+O(\frac{\log(k)}{k}))k^2$	$k\log(k)$	k

Table 1: Characteristics of the main classes of fountain codes..

At ISIT 2009, *Lu and al.* [11] have introduced *black-box linear algebra* and more specifically the use of the Wiedemann algorithm for the decoding of fountain codes. This result offers great perspectives for both efficient decoding and a low ϵ when k is small. However, this research area is still open and many works are still to be done in order to find optimal solutions.

Amongst the established fountain codes, we have chosen RL codes together with Gaussian decoding since it provides low decoding overhead. The cost of decoding a random linear code is mitigated by the fact that we are targeting an application where k is small.

2.3 Relaying strategies

The problem of relaying a fountain code over a line network was first studied by *Pakzad and al.* [15]. The most simple relaying strategy consists to perform no processing. Throughout the paper, this strategy is referred to as *passive relaying*. It is well-known [19, 13] that if some processing is allowed at relaying nodes, the *min-cut capacity* can be achieved. The amount of processing may vary to render the communication scheme attractive. The following strategies are considered.

In a *decode-and-forward relaying*, each intermediate node fully decodes and then re-encodes the information before forwarding packets to its neighbor. In this case, each transmitter (source or relay) compensates for the losses on the transmission link to its immediate neighbor.

The *forward-and-decode relaying* is an hybrid strategy between passive and decode-and-forward. A node $\mathcal{R}_i$ relays passively the messages and tries to de-

[1] We refer the reader to [16] to get further details on Raptor code complexity as it is related to the precode.

code at the same time. When it successfully decodes the information, it sends an acknowledgement to $\mathcal{R}_{i-1}$. Now, $\mathcal{R}_i$ becomes a new source and encodes information for the following nodes.

The *greedy random relaying*, as defined in [15], is considered as a form of network coding [8] where relays $\mathcal{R}_i$ forward random linear combinations over $\mathbb{F}_2$ of the received data.

3 ACK and flooding overhead

An important obstacle for the roll-out of fountain codes in a WSN is the cost of acknowledgement. The aim of this section is to provide an analysis of the flooding traffic in an idealized communication scheme.

The cost of acknowledgement is related to two types of packets that are transmitted after a successful decoding by $\mathcal{R}_i$. Acknowledgment (ACK) packets are an incompressible cost: they are necessary to end the overall data transmission and for $\mathcal{S}$ to start transmitting other data. Since the source and the relays $\mathcal{R}_i$ are constantly transmitting encoded packets until an ACK message is being received, there is another cost referred to as the *flooding traffic*. This overhead traffic is measured by the number the packets sent by $\mathcal{S}$ and the relaying nodes $\mathcal{R}_i$ after a successful decoding of the input packets by $\mathcal{D}$. They are a side-effect of an hop-by-hop mode of transmission of the ACK. Similarly, we denote in the following by *acknowledgement traffic* the number of ACK packets sent in the network.

To give an intuitive idea of what is the acknowledgement/flooding traffic, we first consider transmission channels for which no errors occur. Moreover, the message scheduling is supposed to be ideal. When the nodes $\mathcal{R}_{i-1}$ and $\mathcal{R}_{i+1}$ send simultaneously data to $\mathcal{R}_i$, $\mathcal{R}_i$ receives at least one message. Again, these assumptions are not made to express a realistic transmission scheme but they are very useful to give a rough model of what happens after a successful decoding. Transmission delays along each channel of the main and feedback paths are assumed constant and identical for all packets.

The progression of the acknowledgement is h times faster than the progression of the data: when a data packet progresses of one hop, the acknowledgement progresses of h hops. This assumption is reasonable and is motivated by the fact that ACK packets are usually smaller than regular data packets, and hence suffer from a lower packet error rate. An example of transmission is given in Fig. 2 for $n = 4$ and $h = 1$.

Over a line network, it is straightforward to see that for any relaying strategy, the acknowledgement traffic is at least linear in the number of hops n since $\mathcal{S}$ and each relay $\mathcal{R}_i$ has to receive the ACK. For passive and greedy random relaying, the ACK is directly forwarded from $\mathcal{D}$ to $\mathcal{S}$. For decode-and-forward and forward-and-decode relaying, the source $\mathcal{S}$ is virtually moving towards $\mathcal{D}$: $\mathcal{S}$ receives the ACK from $\mathcal{R}_1$, $\mathcal{R}_1$ from $\mathcal{R}_2$ and so on.

The flooding traffic is in $O(n)$ for decode-and-forward. The node R_i acknowledges R_{i-1} and can only receive some flooding traffic from R_{i-1}. The forward-and-decode follows the same model: if $\mathcal{D}$ decodes successfully, then $\forall i \in [1, n-1], \mathcal{R}_i$ has decoded successfully.

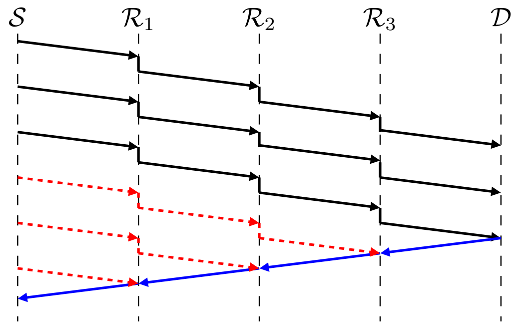

Figure 2: Message timeline. The black plain arrows represent the transmission that carries useful information to $\mathcal{D}$. After and during a successful decoding all the transmissions are useless to $\mathcal{D}$ (dotted arrows). After a successful decoding, $\mathcal{D}$ acknowledges the information reception. The emission of the acknowledgement can collide with some emissions at the node level. It is assumed here for simplicity that collision are handled in favor to the ACK.

For passive and greedy random relaying, the flooding traffic can be approximated by (in number of packets):

$$n\lfloor\frac{n}{h}\rfloor - h\cdot\frac{\lfloor\frac{n}{h}\rfloor(\lfloor\frac{n}{h}\rfloor+1)}{2}. \qquad (1)$$

If the ACK has been received by r nodes, there are still $(n-rh)$ potential nodes relaying information. Equation 1 is obtained by the summation of all the nodes that can potentially emit information before the ACK reaches S.

The model considered here is far from being realistic but it gives us a lower bound on the flooding traffic. Next Section 4 challenges these bounds for a 802.15.4 WSN.

4 Simulation

4.1 Simulation setup

Our simulations are done on WSNet [7], an event-driven network simulator. The MAC protocol used by the sensors follows the IEEE 802.15.4 standard [17] and the physical characteristics correspond to the CC1100 Chipsets from Texas Instruments [10]. The access method of IEEE 802.15.4 considered in this paper is unslotted CSMA/CA. An ACK packet is a regular data packet with a single bit payload. So far, most of the studies on fountain codes have assumed perfect feedback mechanisms. In this paper, we consider a realistic transmission scheme where acknowledgments can also suffer from loss based on various channel statistics. However, even in a realistic configuration, ACK packets have a lower PER because of their smaller size. In our configuration, $\mathcal{D}$ repeats the acknowledgment until it stops receiving data packets.

The transmission channel is characterized by its PER derived as a function of the Signal to Noise Ratio (SNR) γ on a link. The SNR is derived using an isotropic propagation model with a pathloss exponent of $\alpha = 2$. The PER

Layer	Configurations
Application	Network size (in hops): n Distance between each node = 85m
Networking	MAC protocol: Unslotted IEEE 802.15.4 CSMA/CA Transmission period of source = 1s Coding: RL code PDU size = 128bytes Block length = k
Radio	Radio device: Chipset CC1100 Modulation: BPSK Transmitted power = 10dB Transmission rate = 20Kbit/s Frequency = 868MHz
Propagation	Propagation model: AWGN Pathloss exponent $\alpha = 2$ White noise = -111dBm/Hz Fading: none

Table 2: Simulation parameters.

depends on Bit Error Rate (BER) as follows: $PER = 1 - (1 - BER(\gamma))^{\ell}$, with ℓ the length of the packet. The BER depends on the type of modulation and the type of channel considered. It is derived in our case for a BPSK modulation and AWGN channel model using $BER(\gamma) = Q\left(\sqrt{2\gamma}\right) = 0.5 * \mathrm{erfc}(\sqrt{\gamma})$.

The parameters used in our simulations are summarized in Table 2. The simulation results are averaged over 1000 trials. A first simulation set deals with the transmission delay of the different strategies for $k = 10$ in Table 3. The best solution is greedy random relaying: it is fast and scales well. At the other end, decode-and-forward is the slowest and its delay grows linearly with n.

	$n = 5$	$n = 10$	$n = 15$
Decode-and-forward	69.9	138.2	206.5
Forward-and-decode	66.8	134.1	202.7
Passive relaying	24.4	53.0	111.3
Greedy random relaying	20.2	29.0	57.9

Table 3: Transmission delay in seconds for $k = 10$.

We now turn our attention to characterizing flooding traffic.

4.2 Assessing the flooding traffic model

The flooding traffic has been computed as a function of $n \in [2, 15]$, for $k = 10$. The flooding traffic observed for decode-and-forward and forward-and-decode is marginal. For passive and greedy random relaying, the analytical estimation of Equation 1 is confronted with our simulations in Fig. 3 on average results. Parameter h for each strategy has been obtained through a linear regression method. Its appears that the model matches relatively well the simulation. Passive relaying is more impacted by the flooding traffic than greedy random relaying is.

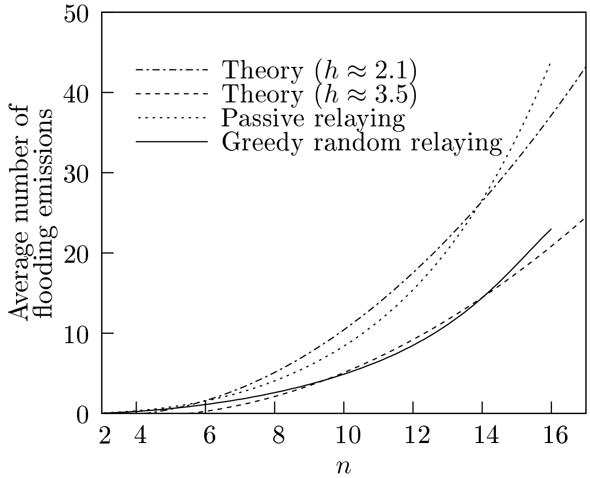

Figure 3: Validation of the flooding traffic f for passive and greedy random relaying (average results).

We observe a high variability in our results. The standard deviation in our previously mentioned simulation results grows quickly with n (see Fig. 4). We observe with nearly the same probability trials in which $h \approx n$ or $h \approx 1$. This is entirely due to random backoff of 802.15.4 that induces high variability in the medium access. We ensure that no congestion occurs by setting the transmission period of the source to one second. The conclusion is that the choice of the MAC layer may greatly influence the flooding traffic.

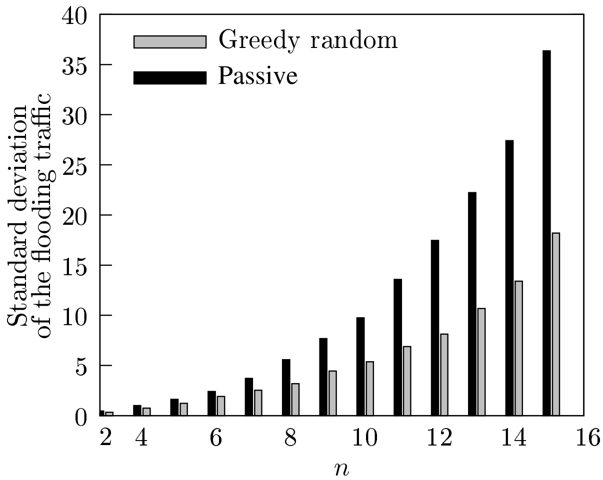

Figure 4: Standard deviation of the flooding traffic.

4.3 Comparing the strategies

Figure 5 provides a detailed traffic analysis of the different strategies for $n = \{5, 10, 15, 20\}$ and $k = 5$. In this dimension, decode-and-forward relaying offers the best results and a high scalability. Passive and greedy random relaying have an exponential growth and for $n = 20$, passive relaying is almost untractable. For $n = 15$, the flooding traffic represents 14% (resp. 8%) of the overall traffic for passive relaying (resp. greedy random relaying).

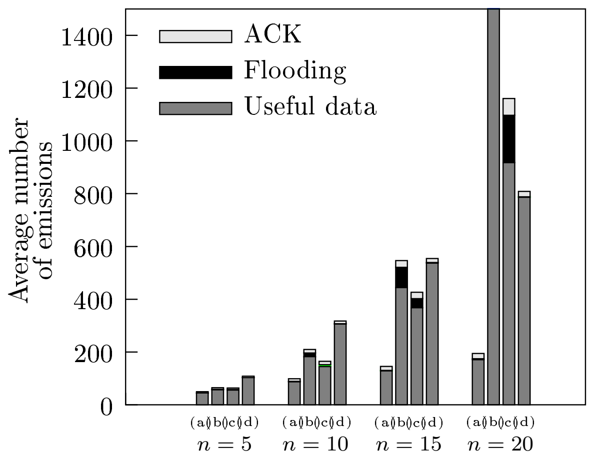

Figure 5: Traffic decomposition of decode-and-forward (a), passive relaying (b), greedy random (c) and forward-and-decode (d) relaying for $n = \{5, 10, 15, 20\}$ and $k = 5$.

The Fig. 6 (a) to (c) provide relay by relay the emitted traffic relative to passive, greedy random and forward-and-decode strategies for $k = 5$ and $n = 10$. In this case, greedy random performs the best. It has also the best fairness (Jain's fairness equals to 0.92). Passive relaying and forward-and-decode have respectively 0.80 and 0.79 for fairness. For forward-and-decode, many packets are redundant which explains the shape of the curve. Removing this redundancy is possible, but at the price of a quadratic encoding cost.

5 Conclusion

Fountain codes are attractive for WSNs because of their rateless property. Our analysis identified the main obstacles to their deployments on IEEE 802.15.4, *i.e.* the data size k and the acknowledgement path (n-hop). Decode-and-forward relaying is to be preferred if the application is delay-tolerant since it guarantees a minimal number of transmissions. It preserves the sensors' energy. For applications in which both delay and energy are critical, the situation is more complicated. If the number of hops on the route is below fifteen, greedy random relaying is well adapted. Otherwise, forward-and-decode is better.

An important question left by the paper is how does it scale with other MAC layers. It is easy to see that our model applies well on TDMA schemes. Many MAC layers have been proposed to preserve the sensors energy through efficient

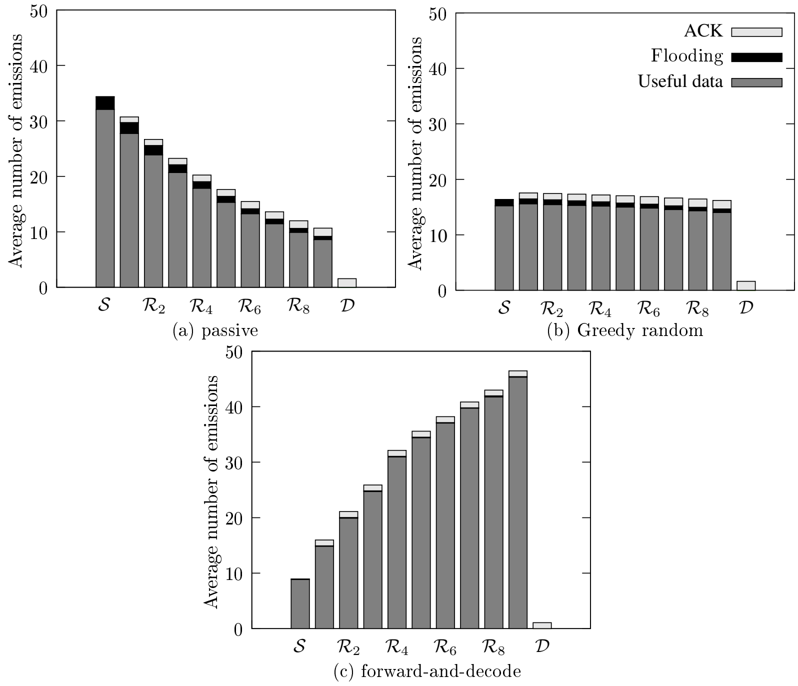

Figure 6: Emitted traffic for each node ($n = 10$). We have $k = 5$ and we consider passive (left), greedy (center) and forward-and-decode (right) relaying.

duty-cycling. The authors will look next into the suitability of such solutions for the deployment of fountain codes.

References

[1] Apavatjrut Anya, Claire Goursaud, Katia Jaffrès Runser, Cristina Comaniciu, and Jean-Marie Gorce. Towards increasing diversity for the relaying of LT Fountain Codes in Wireless Sensor Networks. Research Report RR-7231, INRIA, March 2010.

[2] Amos Beimel, Shlomi Dolev, and Noam Singer. RT oblivious erasure correcting. *IEEE/ACM Transactions on Networking*, 15(6):1321–1332, 2007.

[3] John W. Byers, Michael Luby, and Michael Mitzenmacher. Accessing Multiple Mirror Sites in Parallel: Using Tornado Codes to Speed Up Downloads. In *INFOCOM '99*, pages 275–283, New York, NY, USA, 1999. IEEE.

[4] John W. Byers, Michael Luby, Michael Mitzenmacher, and Ashutosh Rege. A Digital Fountain Approach to Reliable Distribution of Bulk Data. In *Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication - SIGCOMM 1998*, pages 56–67, Vancouver, BC, Canada, August 1998.

[5] Paolo Casari, Michele Rossi, and Michele Zorzi. Fountain codes and their application to broadcasting in underwater networks: performance modeling and relevant tradeoffs. In *ACM international workshop on Underwater Networks - WuWNeT '08*, pages 11–18, San Francisco, California, USA, 2008. ACM.

[6] Mary-Luc Champel, Kévin Huguenin, Anne-Marie Kermarrec, and Nicolas Le Scouarnec. LT Network Codes. In *International Conference on Distributed Computing Systems - ICDCS 2010*, page to appear, Genoa, Italy, June 2010. IEEE Computer Society.

[7] Antoine Fraboulet, Guillaume Chelius, and Eric Fleury. Worldsens: development and prototyping tools for application specific wireless sensors networks. In *ACM/IEEE International Conference on Information Processing in Sensor Networks -IPSN 2007*, pages 176–185. ACM, 2007.

[8] Christina Fragouli, Jean-Yves Le Boudec, and Jörg Widmer. Network coding: an instant primer. *Computer Communication Review*, 36(1):63–68, 2006.

[9] Esa Hyytiä, Tuomas Tirronen, and Jorma T. Virtamo. Optimal Degree Distribution for LT Codes with Small Message Length. In *INFOCOM 2007*, pages 2576–2580, Anchorage, AK, USA, May 2007. IEEE.

[10] Texas Instruments. CC1100 Single Chip Low Cost Low Power RF-Transceiver, 2006. http://focus.ti.com/lit/ds/symlink/cc1100.pdf.

[11] Feng Lu, Chuan Heng Foh, Jianfei Cai, and Liang-Tien Chia. LT codes decoding: Design and analysis. In *International Symposium on Information Theory - ISIT 2009*, pages 2492–2496, Seoul, South Korea, 2009. IEEE.

[12] Michael Luby. LT Codes. In *Foundations of Computer Science - FOCS 2002*, pages 271–, Vancouver, BC, Canada, November 2002. IEEE Computer Society.

[13] Urs Niesen, Christina Fragouli, and Daniela Tuninetti. On Capacity of Line Networks. *IEEE Transactions on Information Theory*, 53(11):4039–4058, 2007.

[14] A. Oka and L. Lampe. Data extraction from wireless sensor networks using distributed fountain codes. *IEEE Transactions on Communications*, 57(9):2607–2618, 2009.

[15] Payam Pakzad, Christina Fragouli, and Amin Shokrollahi. Coding schemes for line networks. In *IEEE International Symposium on Information Theory - ISIT 2005*, pages 1853–1857, Adelaide, Australia, 2005. IEEE.

[16] Amin Shokrollahi. Raptor codes. *IEEE Transactions on Information Theory*, 52(6):2551–2567, 2006.

[17] IEEE Computer Society. IEEE Std 802.15.4 - Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). Standard, 2003.

[18] Tuomas Tirronen. *Fountain Codes: Performance Analysis and Optimisation.* PhD thesis, 2010.

[19] Daniela Tuninetti and Christina Fragouli. On the throughput improvement due to limited complexity processing at relay nodes. In *IEEE International Symposium on Information Theory - ISIT 2005*, pages 1081–1085, Adelaide, Australia, 2005. IEEE.

Centre de recherche INRIA Grenoble – Rhône-Alpes
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399