Network Coding over Satellite: From Theory to Design and Performance

M.A. Vazquez-Castro[1(✉)] and Paresh Saxena[2]

[1] School of Engineering, Autonomous University of Barcelona, Campus de Bellaterra, 08193 Barcelona, Spain
angeles.vazquez@uab.es

[2] AnsuR Solutions Barcelona, Advanced Industry Park, c/Marie Curie 8-14, 08042 Barcelona, Spain
paresh.saxena@ansur.es

Abstract. The concept of network coding has greatly evolved since its inception. Theoretical and achievable performance have been obtained for a wide variety of networking assumptions and performance objectives. Even if powerful, such a broad applicability poses a challenge to a unified design approach over different communication networks and systems.

In this work, we propose a (non-reductionist) unified network coding design architectural framework where an ontology of abstraction domains is introduced rather than layer/system/network-specific assumptions and designs. The framework brings together network and system design and seems compatible with upcoming (more general) design frameworks such as software-defined networking, cognitive networking or network virtualization. We illustrate its applicability showing the case of network coding design over DVB-S2X/RCS.

Keywords: Network coding · Satellite communication system

1 Introduction and Contributions

1.1 Evolution of the Concept of Network Coding

The concept of network coding could be to some extent dated back to Yeung and Zhang in [1]. The work was inspired by low earth satellite (LEO) communication networks as illustrated in Fig. 1. The general problem studied was the admissible code rate region of a given network with an arbitrary set of connections. The problem was tackled as a general multiple-source coding problem and only non explicit results were obtained, for which reason subsequent work has prominently been focused on assuming linearity. In the following, we provide a brief account of the evolution of the concept, leaving aside security enforcement that the concept can also provide.

Error-Free Network Information Flow. Alswhede et al. in [2] fully developed the concept with focus on the single-source coding problem. The inspiration in this case was computer network applications in general. The work revealed the fundamental fact

P. Pillai et al. (Eds.): WiSATS 2015, LNICST 154, pp. 315–327, 2015.
DOI: 10.1007/978-3-319-25479-1_24

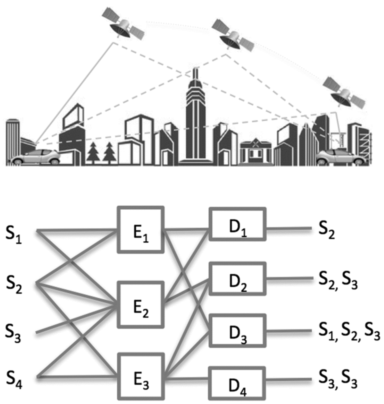

Fig. 1. Illustration of a multisource satellite system along with its logical abstraction as a set of encoders and decoders with arbitrary connections as in [1].

that if coding (rather than mere routing) is applied at the in-network nodes, the source node can multicast the network information flow at the theoretically maximum rate. Such maximum rate, the multicast network capacity, is the smallest minimum cut capacity between the source node and any multicast receiver, what the authors termed as the Max-flow Min-cut for the network information flow. Li et al. [3] further showed that linear network coding and finite alphabet size is sufficient for achievability while Jaggi et al. [4] proposed a deterministic polynomial-time algorithm for network code constructions.

A different thread of work (specifically for networks with non-ergodic error processes, only link failures) was initiated by Koetter and Médard [5]. The authors developed a characterization of linear network coding introducing instrumental tools not only from graph theory but also from control theory and algebra. Specifically, the chosen algebraic framework over finite fields shows how network coding coefficients lead to a matricial network channel transfer whose invertibility drives the solvability of the network (which is equivalent to finding points on algebraic varieties). Alternatively, the same assumption lead to a distributed purely randomized way to find the network coding coefficients in [6], which is capacity achieving.

Given the error-free network assumption in the early work just commented, it can be said that the network coding concept was first discovered aiming at increasing the amount of information transferred across a network (possibly with link-failures). Furthermore, since such error-free networks admit purely logical abstractions, analytically tractable formulations can be obtained as well as analytical closed and/or explicit solutions.

After extensive research assuming error-free (linear) networks, it became apparent that the concept of network coding could be extended to communication networks with links/channels that introduce different types of errors. A number of different proposals exist to tackle this problem each of them illuminating different aspects of it.

Non Error-Free Network Information Flow. Cai and Yeung proposed in [7] to jointly tackle rate maximization and reliability by introducing network error-correcting codes with coherent transmission. The concept of network error correction codes was then to be regarded as a generalization of classical link-by-link error-correcting codes. Subsequent works [8, 9] refined the concept and theoretical coding bounds. Dana et al. in [10] rely on graph-theoretical formulations for the study of a special class of wireless networks, called wireless erasure networks. The multicast capacity under certain assumptions is obtained for such networks revealing a Max-flow Min-cut interpretation. A complementary view is given in Lun et al. in [11], where an (asymptotically) capacity-achieving coding scheme is proposed revealing a fluid-like interpretation of the network flow of "innovative" information.

The algebraic approach to tackle the problem was initiated in [12] by Koetter and Kschischang. They proposed two separated problems (resembling Shannon separation between source and channel coding): the network coding problem for maximizing the network flow (which can be solved assuming error-free links) and the problem of coding for error correction (which can rely on classic coding results and techniques). Accordingly, the proposal is a non coherent transmission model, i.e. neither the source node nor the receiver nodes are assumed to have knowledge of the network coding coefficients. Moreover, it is assumed that the underlying network-coded network implements random linear network coding. Their propose codes as collections of subspaces (for which distance metric driven constructions are developed). The novelty with respect to classic block codes is that while in classical coding theory codewords are vectors, in subspace codes each codework is itself an entire vector space. This can be so due to the fact that the underlying linear network is vector-space preserving. This property ensures that if the source information is represented by a subspace of a fixed vector space and a basis of this subspace is injected into the network, the information can be recovered at the receivers. A worst-case (or adversarial) error model is considered obtaining the maximum achievable rate for a wide range of conditions. Constructions of subspace codes based on rank-metric codes are proposed in [13].

From this brief account of research directions and results when considering network with errors, it is apparent that the concept of network coding become more general with respect to its original purpose. The amount of information transferred across a network needs to be optimized not only in robustness and throughput to achieve capacity but also in reliability.

Physical-Layer Network Information Flow. The concept of network coding further evolved beyond abstracted network flows. Network coding at the physical layer was originally proposed in [14] following the observation that network coding operations are naturally taking place among bearing electromagnetic waves. The well-known additive property of electromagnetic waves can be captured as network coding combinations over the field of complex numbers. For this reason transmission should take place over structured codes as shown in [15–17]. Topologies of interest at the physical layer usually present a higher level of structure than the purely logical ones, the interested reader can check [18] for a survey.

Network Coding for SATCOM. Several works have studied the application of network coding over satellite systems. There are specifically two directions of work on the use of NC in satellite systems.

The first direction focuses on throughput improvement. In [19, 20], network coding is shown to enhance throughput by load balancing and allocating coded packets across different beams in multi-beam satellite systems. Further, works in [21, 22] take advantage of orthogonal transmission available using multi-link reception in multi-beam satellite systems. In this work, network coding is shown to provide benefits in terms of higher throughputs subject to the location of user terminals. Throughput improvement in satcom is then found to depend on enabling extra degrees of freedom in the network, without which network coding alone cannot make much difference.

A different direction looks at the application of network coding to counteract packet losses and guarantee higher reliability and additional performance requirements. It has been shown that network coding together with congestion control algorithms can provide many-fold improvements than existing transport layer protocols [23–25]. In addition, it has been shown in [26], that unequal-protection aware overlapping network coding together with congestion control algorithms can provide improvements in quality-of-experience (QoE) of video streaming. Further, network coding implementation at the link layer has shown to provide several advantages in terms of reliability, complexity, delay, etc. for unicasting and multicasting over satellite networks with re-encoding at intermediate nodes [27, 29]. In [30], it is also shown that it can be used to counteracting prediction failures of the handover procedures in smart gateway diversity satellite systems.

1.2 Discussion and Contributions

From the above brief account, it is apparent the evolution of the concept of network coding for which a rich body of theoretical results exist. Accordingly, it opens up a wide variety of potential applications in general and in satellite communications in particular. Even if powerful, such a broad applicability poses a challenge to a unified design approach over different communication networks and systems.

In this work we propose a network coding design architectural framework, which can be mapped to whatever software and/or hardware resources are provided by the underlying abstracted systems and networks. In particular, our contributions are:

- Proposal of a unified network coding design architectural framework with the following distinguishing features:
  - *Non-reductionist*: Differently to current standard practice, rather than layer/system/network-specific assumptions and designs, an ontology of abstraction domains is introduced. Specific network codes or performance targets can be freely chosen by the designer.
  - *Backward-Forward Compatible:* The proposed framework is backward compatible with existing networks and system. Most importantly, it seems compatible with upcoming more general design frameworks such as: software defined networking, cognitive networking or network virtualization.

  - *Inter-disciplinary*: Network, system and coding experts/designers can easily collaborate in the design based on a common design framework with well defined ontological domains.
- We illustrate its applicability for a use case of network coding over Digital Video Broadcasting by Satellite – Second Generation (Extension)/Return Channel Satellite (DVB-S2X/RCS) [31].

2 Proposal of Network Coding Design Framework

2.1 Ontology of Abstraction Domains

Theoretical results in network coding literature usually make use of networking terms such as packets to refer to information units or flows to refer to sequences of them. However, such terms actually refer to mathematical objects, such as vectors or symbols living in some algebraic object. Moreover, while for a networking designer a packet usually refers to the network layer, for a satellite system designer a packet may refer to the link layer data unit of some system data plane.

In order to solve this problem of ambiguity, we build upon the fact that the human brain naturally disambiguates linguistic terms by relating to the *references* they apply to [32]. Such references can be physical or not. If they are physical, the brain makes the unambiguous description of the physical object (this is the way humans learn language and start communicating). In case they are not, which is our case, the brain should be given a reference/description of the non-physical (abstract) object (this is the way e.g. programmers or system designers work) together with the identification of the domain the term belongs to. The challenge in network coding is that we need a terminology for the basic data unit that can also serve for system-level design of a networking technology, which moreover need clear maps to certain mathematical objects.

We propose a clear distinction between the following two abstraction domains for network coding design[1]:

- *Mathematical-Abstract Domain*: This is the domain of network coding theoretical design and the *reference* of the terms to be used for this domain can be taken from the original sources of the corresponding network coding concepts, e.g. network flow in [2].
- *Logical-Abstract Domain*: This is the domain of our proposed network coding design architectural framework. It includes the necessary mapping between mathematical and logical objects so that theoretical network codes can be incorporated into this design domain. We propose as *reference* of the terms for this domain already existing networking terms.

Consequently, in the rest of this paper we will not use mathematical-abstract domain terminology. Note that our proposed logical-abstract domain abstracts away both system-level domain (functionalities of physical system elements) and networking-level

[1] Note that this distinction is simpler and well established in classic coding theory.

domain. The latter, will only be specified for a specific network coding design, which will have a clear reference in the standard layered communication model.

2.2 Logical-Abstract Domain: Proposed Design Architectural Framework

This is the domain of our proposed architectural framework. This means the designer (or team of designers) working in this domain do not need to work with mathematical objects. This can be so, because the designer should have access to the necessary mappings between mathematical and logical objects. This way, mathematical models and codes can be incorporated into the abstract-logic framework. Note that the framework is not meant to verify whether or not network coding provides an advantage w.r.t. routing, this should be known in advance (see e.g.) [34]. The framework allows a team of designers to collaborate in order to meet a given set of requirements within a unified abstraction domain jointly considering coding, networking and system level/design considerations. In order to do so, a **reference abstraction level** shall be chosen according to the requirements. The natural choice is to define a sub/intra-layer in the standard IP/TCP protocol stack at which network coding will operate. Such a choice inherently sets the appropriate space/time scales and resolution for the design. The framework then is not reductionist, as the networking domain is abstracted away, but uses the TCP/IP protocol stack for setting the reference level of abstraction and interactions for every specific design. Once settled the abstraction level of the logical-abstract design domain, a **NC functionalities design toolbox** allows the designer to choose functional design blocks *according to the technical requirements and design objectives.* The designer should also identify which external functionalities (w.r.t the reference layer) the architecture should interact with (e.g. other protocols that can make network coding related functionalities more effective and efficient).

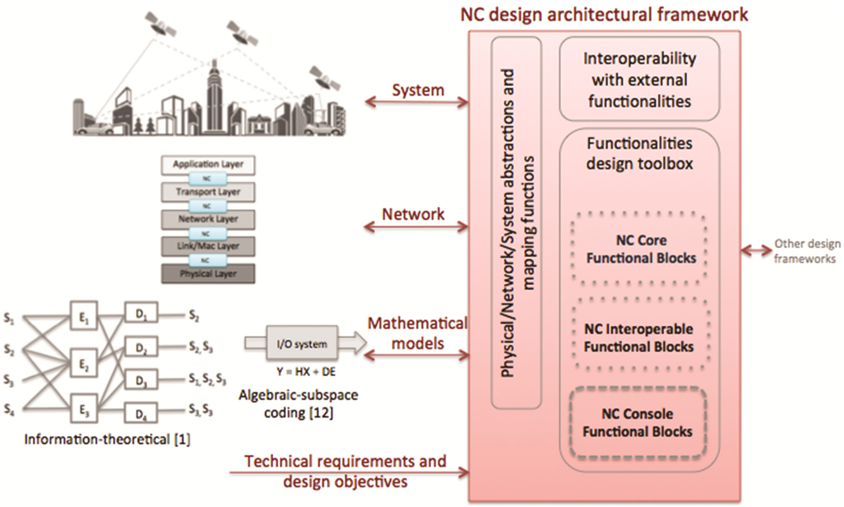

Fig. 2. Proposed NC design architectural framework.

The resulting architectural framework is schematically illustrated in Fig. 2. The output should be a complete network coding functional architecture at the reference communication layer meeting the design requirements and corresponding trade-offs and can be directly implemented as a protocol and/or provided to other more general design frameworks. This way, the framework seems compatible with more general design frameworks such as: software defined networking, cognitive networking or network virtualization. This is so because the design framework can incorporate any underlying software and/or hardware resources and constraints in the same way as any of these other frameworks. In addition, in-network encoding nodes define network coded virtual routes, which can cognitively adapt to network/system dynamics.

2.3 Network Coding Design Toolbox

The proposed architectural framework design provides elementary functionality design blocks (FBs), which can be related to technical requirements (see Fig. 2). The designer may choose existing coding schemes as inputs to the framework (e.g., Greedy and Sparse random codes [35], BATS codes [36], FUN codes [37], DARE [33] and Fulcrum Network Codes [38]) based on the requirements and design target. Such schemes may or may not be compatible with existing forward error correction (FEC) schemes. We propose a preliminary set of NC FBs distributed into three hierarchical levels based on their significance, universality and availability.

NC Core Blocks: These blocks provide the basic core architectural functionalities, elementary coding/re-encoding/decoding operations and logical interpretation for NC usage. In general, they are the main FBs that are foreseen to be present in any NC architecture:

- **NC Logic FB:** It drives the overall network coding solution based on some network coding design, for which a mapping mathematical-logical is given. The scheme can be intra-session/inter-session, coherent/incoherent, large files/streaming, systematic/ non-systematic, etc. Per-node functionalities interact to decide on the use of coding coefficients (random/deterministic), generate and supply them to NC coding FB.
- **NC Coding FB:** It takes care of all the data processing steps including encoding, re-encoding, decoding operations, encapsulation processes, adding/removing headers, etc. It is the central FB of the NC architecture interacting with several others building blocks to allow data processing and flowing.

NC Interoperable Blocks: These blocks provide the interoperability of NC core architecture with external functional blocks. They interact with other external system/protocol level protocols and provide a comprehensive solution for congestion control, cross-layer optimizations, etc. that drive the functionalities of NC core blocks. Examples are:

- **NC Resource-Allocation:** It is responsible for adaptive network coding and allocates optimal coding parameters taking into account different tradeoffs.
- **NC Congestion Control:** It is responsible for managing congestion. It is primarily used when NC is implemented in the upper layers of the protocol stack. It interacts with other cross-layer congestion-control algorithms.

NC Console Blocks: These blocks provide the basic support functionalities (of storage, feedback, etc.) to NC core blocks and NC interoperable blocks, examples are NC storage, NC feedback manager and NC signaling.

3 Design Illustration

In this section, we illustrate the applicability of the proposed framework for the design of network coding for satellite systems over DVB-S2X/RCS. Specifically, two different use cases are presented: kernel-aware network coding unicast [28] and kernel-agnostic (overlay) network coding multicast [26, 29]. The main relevance of these use cases here is to show that the proposed design framework allows the underlying DVB-S2X/RCS standard to be properly abstracted if needed resulting in different operative solutions.

In general, satellite systems suffer from packet losses due to time-variant fading. Hence, we assume encoding at the source to counteract packet losses (erasures) and one in-network encoding node (not necessarily at the satellite) between the source and the sink for the unicast case and between the source and the two sinks for the multicast case. The chosen design objectives of network coding application here is to guarantee quality-of-service (QoS) in terms of reliability and transmission rates for both cases and also QoE for multicast (see [26] for details). Note that given the specific characteristics of the underlying system, the transmission is not assumed interactive.

The architecture design is shown in Fig. 3. Due to the limited space in this article, functions for the different induced time/space scales/resolution for each case are not explained here. The design for the in-kernel unicast has been chosen to be compliant with the existing FEC in the standard at the link-layer (LL) and hence the NC reference layer is identified between the IP layer and the link layer, which we term as (systematic) network coding (LL-SNC). The NC storage FB interacts directly with the physical storage and interacts frequently with NC core blocks to support coding functionalities. The NC feedback manager analyses the feedback and provide support to NC resource allocation FB to evaluate optimal coding parameters. The interactions may or may not be frequent depending on the dynamics coherence time. Finally, the NC signalling FB is responsible to process all the signalling parameters within the NC packets. These parameters (like packet size, frame number, coding scheme ID, systematic/non-systematic packet ID, etc.) are transferred across the network within the NC packets and NC signalling block is responsible for its management. The NC coding scheme (e.g. mapped from some mathematical design) is then responsible for all the coding/re-encoding/decoding operations and NC packetization/de-packetization. Also note that the middle level of FBs or the proposed design serves two main purposes. First, it is interoperable with external functions and performs cross-layer interactions with other protocols. Based on these cross-layer interactions and feedback on the network statistics (packet loss report), optimal coding parameters (code rate, frame length, etc.) can be chosen in order to guarantee QoS and QoE. Such optimal coding parameters are then passed to top-level NC core functions for NC data processing. The bottom level hierarchy consists of NC support blocks that include NC feedback manager FB, NC storage FB and NC Signalling FB.

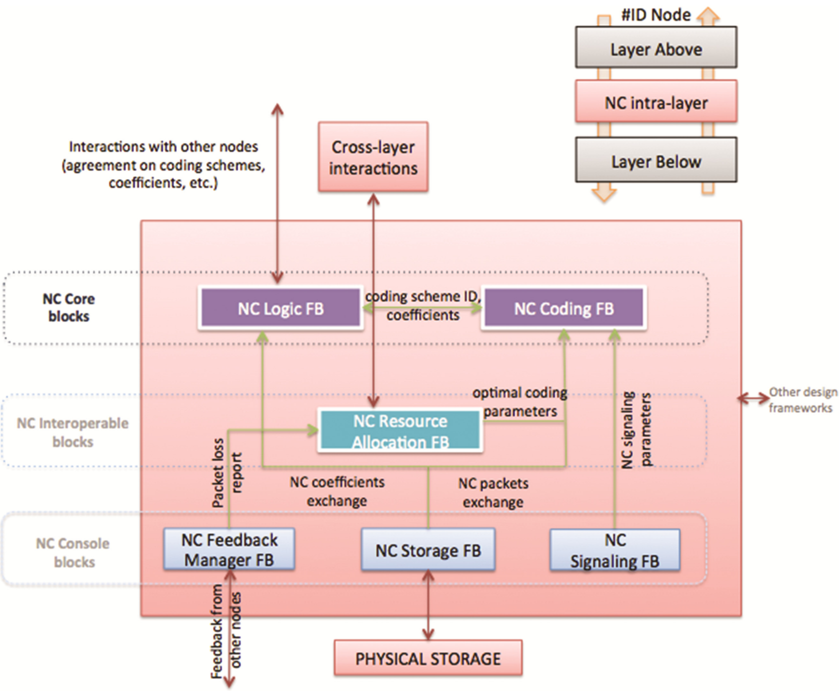

Fig. 3. Illustration of functional architecture network coding design over DVB-S2X/RCS.

In Figs. 4 and 5 we show illustrative performance results for the QoS design objectives. Figure 4 shows the kernel-aware design performance as compared to existing LL-FEC in DVB-S2X/RCS. The simulation assumes realistic satellite setups with light rainfall (20 % packet erasures) at each link and available frame lengths ($N = \{50,256\}$) in the standard. The overall reliability is plotted against the code rate. Our results show that up to 53.85 % higher reliability is guaranteed for a code rate of 0.7. This particular code rate is relevant here as LL-SNC achieves 100 % reliability. Similarly with the frame length ($N = 50$, hence, lower delay), up to 16.28 % higher reliability is guaranteed using LL-SNC. A more detailed description can be found in [28].

For the kernel-agnostic case, the reference layer is identified between the application layer and transport layer. In Fig. 5., we show illustrative performance results with one intermediate *logical* node and two sinks (users) with realistic channel conditions (light rainfall, 20 % for the common link and 20 % and heavy rainfall, 60 % packet erasures for the multicast links) and realistic transmission rates.

Achievable rates are shown compared to the theoretical Max-flow Min-cut. In addition, a region with lower reliability but with a 43.17 % rate increase is also shown in which the sinks with worst channel conditions will recover fewer packets.

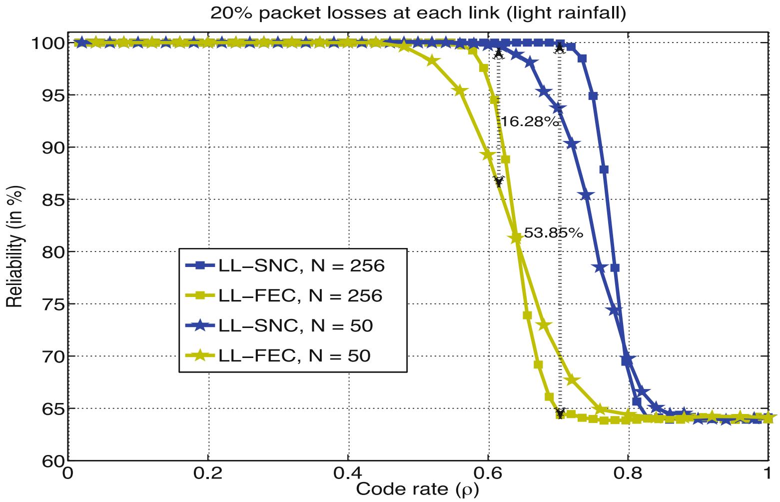

Fig. 4. Illustrative performance results of network coded unicast design between IP and link layer (and comparison with current LL-FEC).

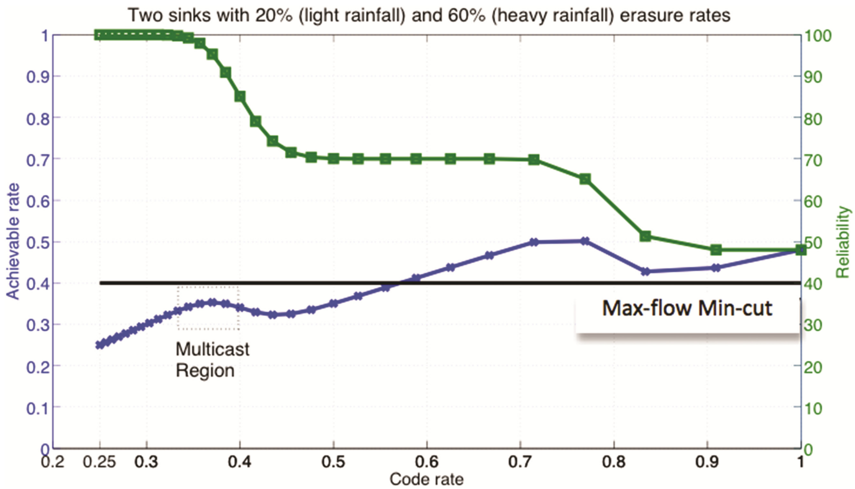

Fig. 5. Illustrative performance results of network coded multicast between application and transport layer.

4 Conclusions and Open Issues

In this work, we propose a (non-reductionistic) unified network coding design architectural framework. The framework brings together coding, network and system design (hence including system-level current standardization practices) within the same ontological level so that interdisciplinary teams can cooperate. The framework is validated by showing its applicability to two different use case scenarios of network coding over satellite. Our proposal contributes to expand satellite communications design beyond purely system-level design over physical elements. Our framework seems compatible with upcoming (more general) design frameworks such as software-defined networking, cognitive networking or network virtualization (see e.g. current activity in Internet research task force, IRTF [39]).

Acknowledgement. The authors acknowledge inter-disciplinary networking support by the COST Action IC 1104.

References

1. Yeung, R.W., Zhang, Z.: Distributed source coding for satellite communications. IEEE Trans. Inf. Theory **45**(4), 1111–1120 (1999)
2. Ahlswede, R., Cai, N., Li, S.-Y.R., Yeung, R.W.: Network information flow. IEEE Trans. Inf. Theory **46**(4), 1204–1216 (2000)
3. Li, S.-Y.R., Yeung, R.W., Cai, N.: Linear network coding. IEEE Trans. Inf. Theory **49**(2), 371–381 (2003)
4. Jaggi, S., Sanders, P., Chou, P.A., Effros, M., Egner, S., Jain, K., Tolhuizen, L.M.G.M.: Polynomial time algorithms for multicast network code construction. IEEE Trans. Inf. Theory **51**(6), 1973–1982 (2005)
5. Koetter, R., Médard, M.: An algebraic approach to network coding. IEEE/ACM Trans. Netw. **11**(5), 782–795 (2003)
6. Ho, T., Koetter, R., Médard M., Karger, D.R., Effros, M.: The benefits of coding over routing in a randomized setting. In: Proceedings of IEEE International Symposium on Information Theory (2003)
7. Cai, N., Yeung, R.W.: Network coding and error correction. In: Proceedings of IEEE Information Theory Workshop 2002, pp. 119–122, Bangalore, India, October 2002
8. Zhang, Z.: Linear network error correction codes in packet networks. IEEE Trans. Inf. Theory **54**(1), 209–218 (2008)
9. Yang, S., Yeung, R.W., Ngai, C.K.: Refined coding bounds and code constructions for coherent network error correction. IEEE Trans. Inf. Theory **57**(3), 1409–1424 (2011)
10. Dana, A.F., Gowaikar, R., Ravi Palanki, R., Hassibi, B., Effros, M.: Capacity of wireless erasure networks. IEEE Trans. Inf. Theory **52**(3), 789–794 (2006)
11. Lun, D.S., Medard, M., Koetter, R., Effros, M.: On coding for reliable communication over packet networks. Phys. Commun. **1**(1), 3–20 (2008)
12. Koetter, R., Kschischang, F.: Coding for errors and erasures in random network coding. IEEE Trans. Inf. Theory **54**(8), 3579–3591 (2008)
13. Silva, D., Kschischang, F., Koetter, R.: A rank-metric approach to error control in random network coding. IEEE Trans. Inf. Theory **54**(9), 3951–3967 (2008)
14. Zhang, S., Liew, S.C., Lam, P.P.: Hot Topic: Physical-layer Network Coding. In: ACM MobiCom, pp. 358–365, September 2006
15. Nazer, B., Gastpar, M.: Compute-and-forward: harnessing interference through structured codes. IEEE Trans. Inf. Theory **57**(10), 6463–6486 (2011)
16. Nazer, B., Gastpar, M.: Reliable Physical Layer Network Coding. Proc. IEEE, Spec. Issue Netw. Coding **99**(3), 438–460 (2011)
17. Vazquez-Castro, M.A.: Arithmetic geometry of compute and forward. In: Proceedings of IEEE Information Theory Workshop (2014)

18. Liew, S.C., Zhang, S., Lu, L.: Physical-layer network coding: tutorial, survey, and beyond. Phys. Commun. **6**(1), 4–42 (2013)
19. Vieira, F., Shintre, S., Barros, J.: How feasible is network coding in current satellite systems ?. In: ASMS Conference and SPSC Workshop, pp. 31–37 (2010)
20. Vieira, F., Lucani, D., Alagha, N.: Load-aware soft-handovers for multibeam satellites: a network coding perspective. In: ASMS Conference and SPSC Workshop, pp. 189–196 (2012)
21. Alegre-Godoy, R., Alagha, N., Vazquez-Castro, M.A.: Offered capacity optimization mechanisms for multi-beam satellite systems In: IEEE ICC, pp. 3180–3184 (2012)
22. Vazquez-Castro, M.A.: Graph model and network coding gain of multibeam satellite communications. In: IEEE ICC, pp. 4293–4297 (2013)
23. Gupta, S., Vazquez-Castro, M.A.: Location-adaptive network-coded video transmission for improved quality-of-experience. In: 31st AIAA International Communications Satellite Systems Conference (ICSSC) (2013)
24. Gupta, S., Pimentel-Niño, M.A., Vazquez-Castro, M.A.: Joint network coded-cross layer optimized video streaming over relay satellite channel. In: 3rd International Conference on Wireless Communications and Mobile Computing (MIC-WCMC) (2013)
25. Cloud J., Leith D., Medard M.: Network Coded TCP (CTCP) Performance over Satellite Networks. In: International Conference on Advances in Satellite and Space Communications (SPACOMM), pp. 53–556 (2014)
26. Pimentel-Niño, M.A., Saxena P., Vazquez-Castro M.A.: QoE driven adaptive video with overlapping network coding for best effort erasure satellite links. In: 31st AIAA International Communications Satellite Systems Conference (ICSSC) (2013)
27. Saxena, P., Vázquez-Castro, M.A.: Network coding advantage over MDS codes for multimedia transmission via erasure satellite channels. In: The 5th International Conference on Personal Satellite Services (PSATS), June 2013
28. Saxena, P., Vázquez-Castro, M.A.: Link Layer Systematic Random Network Coding for DVB-S2X/RCS. In: IEEE Communications Letters, May 2015
29. Saxena, P., Vazquez-Castro, M.A.: Network coded multicast and multi-unicast over satellite. In: The 7th International Conference on Advances in Satellite and Space Communications (SPACOMM), April 2015
30. Muhammad, M., Giambene, G., De Cola, T.: Channel prediction and network coding for smart gateway diversity in terabit satellite networks. In: GLOBECOMM, pp. 3549–3554 (2014)
31. ETSI EN 302 307 V1.2.1, Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting (DVB-S2) (2009)
32. Kripke S.: Naming and Necessity, pp. 193–219. Harvard University Press, Cambridge, Chapter 10 (1979)
33. Saxena, P., Vazquez-Castro, M.A.: DARE: DoF-aided random encoding for network coding over lossy line networks. In: IEEE Communications Letters (2015)
34. Vazquez-Castro, M. A.: Subspace coding over Fq-linear erasure satellite channels. In: 7th International Conference on Wireless and Satellite Systems (2015). (Invited paper)
35. Pakzad, P., Fragouli, C., Shokrollahi, A.: Coding Schemes for line networks. In: IEEE ISIT, pp. 1853–1857 (2005)
36. Yang, S., Yeung, R., Coding for a network coded fountain. In: IEEE ISIT, pp. 2647–2651 (2011)
37. Huang, Q., Sun, K., Li, X., Wu, D.: Just FUN: a joint fountain coding and network coding approach to loss tolerant information spreading. In: ACM MobiHoc, pp. 83–92 (2014)

38. Lucani, D.E., Pedersen, M.V., Heide, J., Fitzek, F.H.P.: Fulcrum network codes: a code for fluid allocation of complexity. In: IEEE Journal on Selected Areas in Communications Submitted for Publication
39. IRTF: Network Coding Research Group (NWCRG). https://irtf.org/nwcrg