Survivable Network Coding

Cédric Bentz[(1)], Sourour Elloumi[(1)], Éric Gourdin[(2)], Thibaut Lefebvre[(1,2)]

[(1)]CEDRIC,
CNAM,
Paris, France.

[(2)]Orange Labs,
Orange,
Issy-les-Moulineaux, France.

Abstract

Given a telecommunication network, modeled by a graph with capacities, we are interested in comparing the behavior and usefulness of two information propagation schemes, namely multicast and network coding, when the aforementioned network is subject to single arc failure. We consider the case with a single source node and a set of terminal nodes. The problem of studying the maximum quantity of information that can be routed from the source to each terminal, using either multicast replication alone or combined with network coding, has been extensively studied. Multicast protocols allow an intermediate node to replicate its input data towards several output interfaces, and network coding refers to the ability for an intermediate node to perform coding operations on its inputs, for example linear combinations, releasing a coded information flow on its outputs. We consider the survivability extension of the throughput maximization problem where any single arc can fail. We model such a failure by removing this arc from our graph thus losing a part of the information flow of our static routing. Our aim is to design models and algorithms to compute the survivable maximum throughput in multicast network and compare results obtained with and without network coding. The survivable coding advantage is defined as the quotient of the optimal throughput obtained using survivable network coding over the survivable multicast optimal throughput. We provide theoretical and experimental results on this last quantity.

Multicast and network coding

Unlike classical unicast transmission technique where a node is merely a relay, in the multicast framework we allow an intermediate node to duplicate its input data before passing them to its outputs.

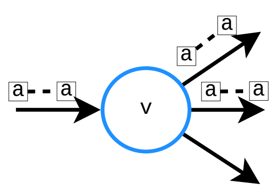


Figure 1: Input data duplication in the multicast framework.

Network coding refers to the ability for an intermediate node to perform coding operations on its inputs, for example linear combinations, releasing a coded information flow on its outputs.

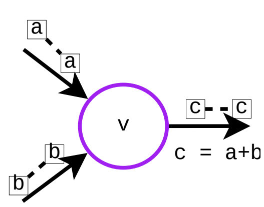


Figure 2: Coding operations at an intermediate node.

Although they seem to be very rare, some networks can benefit from network coding. The so-called butterfly network is a good example.

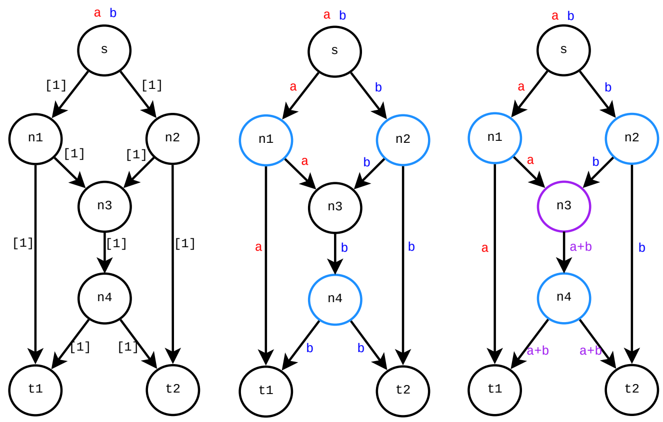


Figure 3: Network coding allows a 33% maximum throughput increase in the butterfly network.

Notations

We consider a network $N = (V, A, s, T, C)$ where (V, A) is a graph, s is a vertex called the source, T is a subset of vertices distinct from the source (each element t of T is called a terminal), C is a function from the set of arcs A to the set of positive integers $\mathbb{Z}_+$ and we note C_a the capacity of an arc a. We suppose that there are two arc-disjoint paths from the source to each terminal and we denote $\mathcal{T}(s, T)$, or simply $\mathcal{T}$ when it is clear from context, the set of Steiner arborescences rooted at s and spanning T. Given such an arborescence τ and a terminal t we note p_τ^{st} the unique path from s to t in τ. We introduce $\mathcal{P}^{st}$ the set of paths from s to terminal t and $\mathcal{P}$ the set of paths from the source to any terminal. We want to compare the behavior of a multicast network without and with network coding.

Results from literature

We are looking for a routing which maximizes the throughput from the source to any terminal under capacity constraints. We first consider the case where only multicast is allowed which is a fractional Steiner arborescences packing problem, [JMS2003]

$$MC \begin{cases} \lambda^{mc} = \max & \sum_{\tau \in \mathcal{T}} \varphi_\tau \\ s.t. & \sum_{\substack{\tau \in \mathcal{T} \\ a \in \tau}} \varphi_\tau \leq C_a \quad \forall a \in A \\ & \varphi_\tau \geq 0 \quad \forall \tau \in \mathcal{T} \end{cases}$$

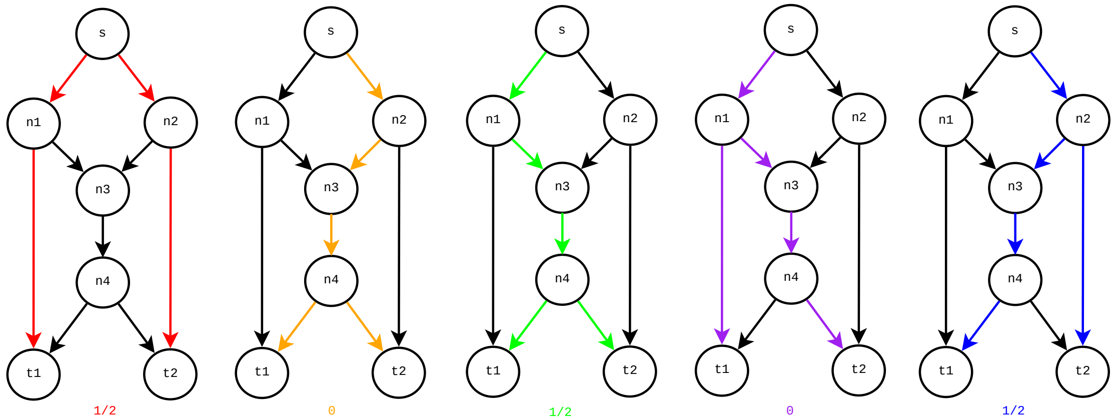


Figure 4: An optimal solution to the fractional Steiner arborescences packing problem in the butterfly network.

When multicast and network coding are both allowed the problem of maximizing the throughput is equivalent to computing the maximum flow between the source and each terminal, and then taking the minimum, see [ACLY2000].

$$NC \begin{cases} \lambda^{nc} = \max \quad \lambda \\ s.t. \qquad\qquad \lambda \le \sum_{p \in \mathcal{P}^{st}} \phi_p \quad \forall t \in T \\ \qquad\qquad\qquad \sum_{\substack{p \in \mathcal{P}^{st} \\ a \in p}} \phi_p \le C_a \quad \forall a \in A, t \in T \\ \qquad\qquad\qquad \lambda, \phi_p \ge 0 \quad \forall p \in \mathcal{P} \end{cases}$$

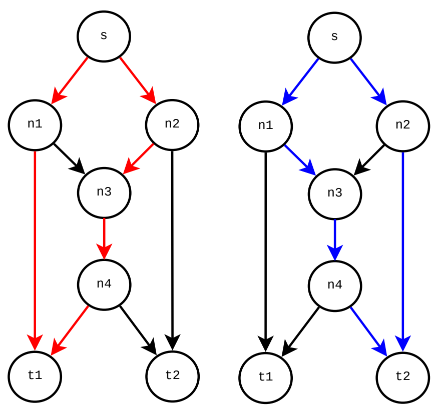


Figure 5: A superposition of two maximum flows gives us an optimal solution to the network coding maximum throughput in the butterfly network. Each terminal receives two units of flow.

We can define the coding advantage θ as follows

$$\theta = \frac{\lambda^{nc}}{\lambda^{mc}}$$

which is always no less than one since a multicast routing is a network coding routing without coding.

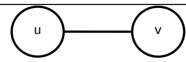
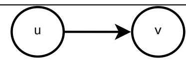
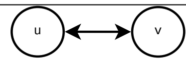
Network	Schema	Flow and capacity	Feature of θ
Undirected		$f_{uv} + f_{vu} \leq C_{uv}$	$\theta \leq 2$ [AC2004], [LL2004]
Directed		$f_{uv} \leq C_{uv}$	θ unbounded [SET2003]
Bidirected		$f_{uv} \leq C_{uv} \quad C_{uv} = C_{vu}$	$\theta = 1$ [YWZXL2012]

Table 1: Feature of θ as a function of the type of network.

In randomly generated digraphs it is very common to find $\theta = 1$, see [GW2012].

Network prone to single arc failure

In the following we will consider that the network is prone to single arc failures. We model such a failure by removing the broken arc from the graph. We are looking for a survivable routing which maximizes the remaining throughput when any failure occurs.

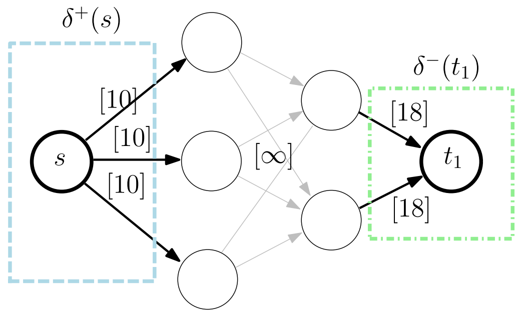


Figure 6: A maximum survivable flow differs from a maximum flow: the value of the maximum flow is 30 (minimum cut $\delta^+(s)$). The value of survivable maximum flow is 15. This value is obtained when considering the two arcs of $\delta^-(t_1)$. To preserve the maximum amount of flow from the failure of one of the two arcs, one would consider sending as much flow as possible (18 units) on the other. But if this arc fails, there is only 12 units of flow on the first arc. So to have as much flow as possible in the case of any single arc failure, the solution consists in sending 15 units on both arcs. Note that this solution is hence determined by both cuts and in the cut $\delta^-(t_1)$, no arc is saturated.

When only multicast is allowed we can formulate the survivable maximum throughput problem as follows

$$MC_A \begin{cases} \lambda_A^{mc} = \max & \lambda \\ s.t. & \lambda + \sum_{\substack{\tau \in \mathcal{T} \\ a \in p_\tau^{st}}} \varphi_\tau \leq \sum_{\tau \in \mathcal{T}} \varphi_\tau \quad \forall a \in A, t \in T \\ & \sum_{\substack{\tau \in \mathcal{T} \\ a \in \tau}} \varphi_\tau \leq C_a \quad \forall a \in A \\ & \lambda, \varphi \geq 0 \end{cases}$$

Observe that if arc a fails, for each arborescence τ we remove the subtree rooted at the tail of a.

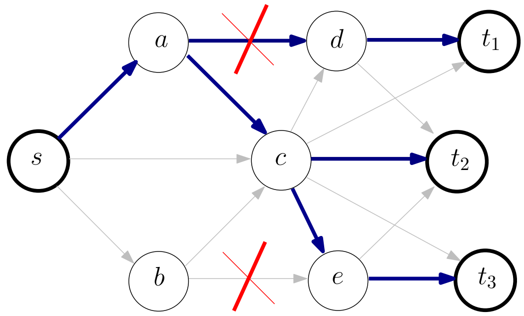


Figure 7: Impact of failures on a multicast tree. The arc (a,d) belongs to the arborescence. The subtree $\{(a,d),(d,t_1)\}$ is discarded but data can still be sent from the source node s towards the terminals t_2 and t_3. The failure of arc (b,e) has clearly no impact on that tree.

If we allow both multicast and network coding we get the following problem

$$NC_A \begin{cases} \lambda_A^{nc} = \max \quad \lambda \\ s.t. \qquad\qquad \lambda + \sum_{\substack{p\in\mathcal{P}^{st} \\ a\in p}} \phi_p \le \sum_{p\in\mathcal{P}^{st}} \phi_p \quad \forall a\in A, t\in T \\ \qquad\qquad\qquad \sum_{\substack{p\in\mathcal{P}^{st} \\ a\in p}} \phi_p \le C_a \quad \forall a\in A, t\in T \\ \qquad\qquad\qquad \lambda, \phi \ge 0 \end{cases}$$

We can thus define the survivable coding advantage θ_A and study it.

$$\theta_A = \frac{\lambda_A^{nc}}{\lambda_A^{mc}}$$

Bibliography

[ACLY2000] R. Ahlswede, N. Cai, S.-Y. R. Li and R.W. Yeung, *Network Information Flow* , IEEE Transactions on Information Theory, vol. 46(4), pp. 1204-1216, 2000.

[JMS2003] K. Jain, M. Mahdian and M. R. Salavatipour, *Packing Steiner trees*, Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 2003.

[SET2003] P. Sanders, S. Egner and L. Tolhuizen, *Polynomial time algorithms for network information flow,* In Proceedings of the fifteenth annual ACM symposium on Parallel algorithms and architectures (pp. 286-294). ACM. 2003.

[AC2004] A. Agarwal and M. Charikar, *On the advantage of network coding for improving network throughput*, In Information Theory Workshop, 2004. IEEE (pp. 247-249). IEEE.

[LL2004] Z. Li and B. Li, *Network coding in undirected networks*, CISS, 2004.

[YWZXL2012] X. Yin, X. Wang, J. Zhao, X. Xue and Z. Li, *On Benefits of Network Coding in Bidirected Networks and Hyper-networks*, INFOCOM, 2012 Proceedings IEEE, pp. 325-333, 2012.

[GW2012] E. Gourdin and Y. Wang, *Some Further Investigation on Maximum Throughput: Does Network Coding Really Help?*, in Proc. of International Teletrac Congress (ITC) 2012.