2012 International Symposium on Network Coding (NetCod)

Extended Diversity Coding: Coding Protection and Primary Paths for Network Restoration

Serhat Nazim Avci and Ender Ayanoglu
Center for Pervasive Communications and Computing
Department of Electrical Engineering and Computer Science
University of California, Irvine
Irvine, CA 92697-2625

***Abstract*—The technique of diversity coding offers fast recovery against failures in networks while keeping spare capacity comparable to the alternative state-of-the art network restoration techniques. It provides near-hitless recovery when coding is performed on connections with the same destination node. When the coding structure is extended to cover the primary paths, diversity coding can achieve higher capacity efficiency than its conventional version and other restoration techniques. In this paper, we develop a systematic approach to implement the diversity coding structures for pre-provisioning with coding of the protection and primary paths. In addition, we present an algorithm to map these coding structures into arbitrary topologies. We present simulation results that show capacity efficiency of diversity coding.**

I. Introduction

In recent years, both the data rate and the strategic importance of the data carried on wide area networks have increased substantially. Yet, these networks undergo failures regulary. A study by Gartner group estimated a $500M revenue loss due to network outages for U.S companies in 2004 [1]. According to the data published by FCC, on average, there are 13 cuts per year for every 1K miles of fiber and, specifically 3 cuts per year for every 1K miles of fiber in metro and long-haul networks respectively. In [2], it is reported that 70% of the network failures consist of single link failures, which leads us to focus on recovery from single link failures in this paper.

Many methods have been developed to recover from the link failures in a cost efficient way. Two main metrics that are associated with the solution of this problem are spare capacity percentage and restoration time. Spare capacity percentage represents the cost of required extra spare capacity to guarantee recovery from single link failures, which is a pre-failure cost to the service provider. Restoration time represents the total outage time that users experience, which is an after-failure cost to the customers. It is desirable to minimize these metrics at the same time but usually there is a tradeoff between them. For telephone networks, 50 ms is set as the target restoration time in order to minimize the human perception of the outage. However, for IP networks, it is desirable to achieve “hitless restoration” such that high delays due to the complexities introduced by different layers of the IP protocols are avoided. A study shows that single link failures can take minutes to recover in IP networks even if the underlying network is equipped with high spare capacity and diverse rerouting paths [3].

This work is partially supported by NSF under Grant No. 0917176.

Two early dedicated path protection techniques are known as 1:1 or 1+1 Automatic Protection Switching (APS). In both cases, a primary path is protected by means of a disjoint protection path. In 1:1 protection, this path is inactive; while in 1+1, it carries the same data as the primary path. In the case of a failure, transmission is switched from the primary to the protection path. In 1:1, there is a delay involved due to the transmission of the failure information back to the transmitter, while in 1+1, this delay is not present and the switching can be hitless. These techniques were not implemented widely since they required extremely high spare capacity. Self-healing rings, or SONET rings, were developed as part of the development of the high-speed optical transmission standards, named as Synchronous Optical Network (SONET), in 1990 [1]. SONET rings satisfy the 50 ms restoration time requirement at the expense of much more than 100% spare capacity percentage. On the other hand, shared path protection (SPP) and shared link protection techniques [4] are able to save capacity by allowing sharing the spare capacity among different connections which are not affected by the same link failure. They are more capacity efficient than dedicated path protection and SONET rings. However, the disadvantages of these techniques are longer restoration time and higher signaling complexity.

In [5], a mixture of link-based protection scheme and ring-type protection scheme was presented and named as p-cycle protection. It was developed to achieve high capacity efficiency and high restoration speed at the same time. This technique forms cycles traversing many nodes using the spare capacity allocated on the links. In the case of a failure, the affected traffic is rerouted over the intact parts of the cycle.

Treating the link failure recovery problem in the context of erasure channel model was introduced by [6], [7] and the technique was called diversity coding. The idea behind this technique is to recover any failed data in any of the link-disjoint N primary paths by extracting the missing data from the erasure code on the parity link. The parity link was formed by applying exclusive or (XOR) operation over the primary paths and carried over a link-disjoint protection path. This technique was actually a simple example of network coding for multiple unicasts before network coding was introduced in 2000 [8]. The fact that one protection path serves for N different primary paths resulted in high capacity efficiency as well as high restoration speed. Diversity coding is faster than all of the mesh-based and ring-based protection schemes, except 1+1 APS, since it eliminates the time consuming rerouting and feedback signaling operations. One inherited advantage of diversity coding is the automatic protection structure which does not require to detect the failure inside the network. A simple failure detection mechanism in the destination node switches to the protection path, which was transmitted at all times without being triggered after the failure. Diversity coding was applied to arbitrary topologies and arbitrary traffic scenarios in [9] and [10]. It was shown that diversity coding achieves much smaller restoration times than SPP and p-cycle protection schemes with a competitive capacity efficiency. When the destination nodes in the diversity coding structure are the same, diversity coding can achieve near-hitless restoration speed [10].

In [11], we introduced a technique called Coded Path Protection (CPP). This technique inputs a solution of mesh-based SPP technique and converts it to the one based on diversity coding. This conversion replaces the time consuming rerouting and signaling operations with faster automatic protection at a small expense in terms of required spare capacity. Since CPP leverages the symmetry in the protection paths, it enables coding and decoding inside the network, which has been pursued inside the network coding framework. The poison-antidote analogy [12] can be used to explain the decoding operations of CPP.

In the conventional diversity coding structure, only protection paths carry the coded data. The idea of incorporating both primary and protection paths into coding operations was discussed in [10]. It was shown with an example that this structure can result in a lower spare capacity requirement than a typical SPP and p-cycle protection technique. We call the new structure extended diversity coding since it has more flexibility than conventional diversity coding in terms of coding. As opposed to the conventional one, an algebraic analysis is required guarantee the decodability of the extended diversity coding structure. In this paper, we present a systematic approach of building decodable extended diversity coding structures and a design algorithm with low complexity for pre-provisioning of the traffic, against single link failures although extension to multiple link failures is relatively straightforward, as in [6], [7]. We investigate the effect of the added coding flexibility on the performance of diversity coding in terms of spare capacity percentage, restoration time, synchronization complexity, and signaling complexity.

II. Extended Diversity Coding

In this paper, we present a systematic approach of building diversity coding structures with higher coding flexibility compared to conventional diversity coding. This approach is based on induction to assure the decodability of the coding structures. Its flexibility of being able to employ more paths comes with a number of performance enhancements over diversity coding in terms of spare capacity percentage, restoration time, and synchronization simplicity. The only downside of this technique is the introduction of a small degree of feedforward signaling. Our second contribution is a capacity placement algorithm which maps extended diversity coding structures into arbitrary topologies with low complexity. This design algorithm stems from the inductive approach that is used in building such codes.

In order to understand the coding structure of extended diversity coding it is helpful to revisit the coding structure of conventional diversity coding. The idea behind conventional diversity coding is protecting N protection paths by employing an $N+1^{st}$ protection path [6], [7]. Assume that the data on the primary paths are defined as d_1,d_2,d_3,...,d_N. Assume, for simplicity, that they share the same destination and source nodes. The coded erasure data c_1 on the protection path is formed by taking modulo-2 sum of these data signals

$$c_1 = d_1 \oplus \ d_2 \oplus \cdots \oplus d_N = \bigoplus_{j=1}^{N} d_j$$

When one of the primary paths fails, the receiver is able to recover the failed data by applying modulo-2 sum to the rest of the received links. If the primary path carrying d_i fails then it is recovered by

$$c_1 \oplus \bigoplus_{\substack{j=1\\ j\neq i}}^{N} d_j = d_i \oplus \bigoplus_{\substack{j=1\\ j\neq i}}^{N} (d_j \oplus d_j) = d_i$$

It is possible to extend this coding structure to topologies where destination nodes and the source nodes are different. However, it is observed in [13] that when the diversity coding is applied to connections only with the same destination node, restoration time, synchronization complexity, and design complexity can be significantly reduced. Therefore, in this paper, we adopt diversity coding and extended diversity coding structures only for the connections with the same destination node.

A. Coding Structure

The idea of coding both primary and protection paths to recover from single link failures has been a research subject since it is shown to enhance the performance of diversity coding in [10]. In addition to the increased capacity efficiency, this coding flexibility can improve the restoration speed and synchronization complexity of diversity coding. In this paper, we present an algorithm to build such codes while guaranteeing decodability for pre-provisioning of the static traffic by employing dynamic provisioning algorithm of diversity coding [13].

In diversity coding, only one summation is sufficient to recover the failed data. In the extended diversity coding, the decoding can involve both primary and protection paths, and therefore, can be more complex. The novel idea is to provision the connection demands one at a time by adding them to the existing valid coding groups to guarantee the decodability for against single link failure.

Assume that there is an existing coding structure whose received vector looks like

$$\begin{bmatrix} a_{11}x + a_{12}y + a_{13}z + a_{14}v \\ a_{21}x + a_{22}y + a_{23}z + a_{24}v \\ a_{31}x + a_{32}y + a_{33}z + a_{34}v \\ a_{41}x + a_{42}y + a_{43}z + a_{44}v \\ a_{51}x + a_{52}y + a_{53}z + a_{54}v \end{bmatrix} \tag{1}$$

where x, y, z, and v are the coded signals and a_{ij} are the binary coding parameters. The received matrix $\mathbf{A} = [a_{ij}]_{5\times 4}$ has a "full rank + 1" property, which means if we delete one of its rows the remaining matrix would be still full rank. Deleting a row is the analogy of a single link failure whereas full rank property assures the decodability of the signals. Therefore, it is vital to have a systematic approach to find the proper values of the variables in matrix $\mathbf{A}$. As a comparison, the same vector in diversity coding structure would look like

$$\begin{bmatrix} x+0y+0z+0v \\ 0x+y+0z+0v \\ 0x+0y+z+0v \\ 0x+0y+0z+v \\ x+y+z+v \end{bmatrix} \quad (2)$$

We observed that if the connection demands are provisioned one-by-one, it is possible to build these coding structure using induction. In short, we build the coding groups starting with a single connection and enlarge them without compromising their decodability. Assume that we have a new connection demand that shares the same destination node with the connections in matrix $\mathbf{A}$. The signal belonging to this connection is denoted as u. It is possible to enlarge the coding group by adding this new connection. There are three sufficient rules to follow to assure that this addition will not compromise the "full rank + 1" property of the new matrix

1. One of the paths of u must be link-disjoint to any path in the coding group.
2. The other path of u must be coded with only one path in the coding group.
3. No path in the coding group must diverge after any node.

If the rules above are followed, the vector is transformed to

$$\begin{bmatrix} a_{11}x+a_{12}y+a_{13}z+a_{14}v+0u \\ a_{21}x+a_{22}y+a_{23}z+a_{24}v+0u \\ a_{31}x+a_{32}y+a_{33}z+a_{34}v+0u \\ a_{41}x+a_{42}y+a_{43}z+a_{44}v+0u \\ a_{51}x+a_{52}y+a_{53}z+a_{54}v+u \\ 0x+0y+0z+0v+u \end{bmatrix}. \quad (3)$$

The link-disjoint path of the new connection is represented by the sixth row whereas the other path is coded with the path shown in the fifth row. The new matrix $\mathbf{A}^{'} = [a_{ij}]_{6\times5}$ has the "full rank + 1" property given $\mathbf{A}$ has this property. Readers can refer to [13] for a proof of this property. The fact that the sixth row of $\mathbf{A}^{'}$ carries uncoded data does not mean it will not be involved in coding operations. Assume that a new connection demand is to be added to this coding group. The new connection carries signal l. A possible addition of the new connection could transform the vector to

$$\begin{bmatrix} a_{11}x+a_{12}y+a_{13}z+a_{14}v+0u+0l \\ a_{21}x+a_{22}y+a_{23}z+a_{24}v+0u+0l \\ a_{31}x+a_{32}y+a_{33}z+a_{34}v+0u+0l \\ a_{41}x+a_{42}y+a_{43}z+a_{44}v+0u+0l \\ a_{51}x+a_{52}y+a_{53}z+a_{54}v+u+0l \\ 0x+0y+0z+0v+u+l \\ 0x+0y+0z+0v+0u+l \end{bmatrix}. \quad (4)$$

That means both paths of the signal u are incorporated into the coding operation. This operation continues until the coding group reaches its topological limit of link-disjointness criterion. In that case, a new connection group can be formed by the new connection demand itself. The $\mathbf{A}$ matrix of the new group would look like

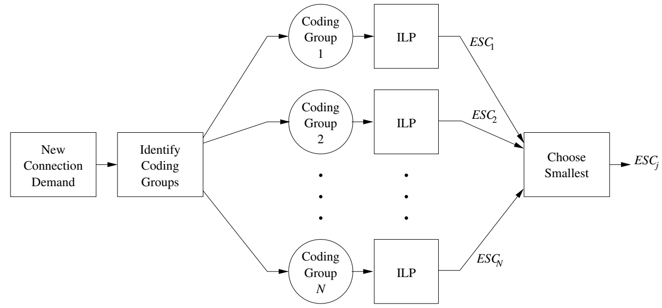


Fig. 1. Extra spare capacity is calculated for each coding scenario and the minimum is chosen.

$$\begin{bmatrix} k \\ k \end{bmatrix} \quad (5)$$

where k is the signal carried by the new connection. The fact that the initial state of a new coding group satisfies the "full rank + 1" property completes the proof of decodability by induction.

B. Proposed Algorithm

We have developed an Integer Linear Programming (ILP) based capacity placement and coding group formation algorithm. This algorithm maps the extended diversity coding structures to arbitrary topologies in order to protect the connections against single link failures. The idea behind this algorithm is derived from the building process of the extended diversity coding in Section II-A and can be extended to multiple link failures with some slight extra complexity. The algorithm is very similar to the optimal algorithm developed for dynamic provisioning of dynamic traffic [13], except it is used for pre-provisioning of static traffic. The algorithm provisions the connection demands of a static traffic matrix one-by-one in a random manner. It attempts to add connection demands to the existing coding groups to save capacity required to route and protect that connection. A new connection from the traffic matrix is added to the group with the same destination which incurs lowest extra spare capacity. This operation is illustrated in Fig. 1. In that figure, ESC means extra spare capacity required to add the connection in each coding group. Note that, ESC depends on the topology of each coding group so it is calculated for each coding group scenario. Then the algorithm selects the one which incurs lowest extra spare capacity. One of the coding groups is an empty coding group to let a connection demand to form its own group. That may be required when it requires the lowest ESC or there is no suitable coding group to add the connection. These newly formed coding groups await for new connection demands as the algorithm proceeds to the rest of the traffic matrix. We have employed an ILP formulation to find the link-disjoint primary and secondary (protection) paths for each scenario. The cost vector of the links is adjusted depending on the topology of each coding group to calculate the required ESC accordingly. Once the best available scenario is found, coding groups are updated and the algorithm passes to the next connection demand in the matrix.

The parameters of the ILP formulation to find a pair of link disjoint primary and secondary paths are as follows.

- $G(V, E)$: Network graph,
- N : Enumerated list of all connections,
- ${a_e}^1$: The cost associated with link e for primary path,
- ${a_e}^2$: The cost associated with link e for secondary path,
- $\Gamma_i(v)$: The set of incoming links of each node v,
- $\Gamma_o(v)$: The set of outgoing links of node v.

The binary ILP variables which take the value of 0 or 1 are

- x_e : Equals 1 iff the primary path of the connection passes through link e,
- y_e : Equals 1 iff the secondary path of the connection passes through link e.

The objective function is

$$\min \sum_{e \in E} x_e \cdot {a_e}^1 + y_e \cdot {a_e}^2. \quad (6)$$

The origination, flow, and termination of the primary path (x_e) and the secondary path (y_e) are determined by

$$\sum_{e \in \Gamma_i(v)} x_e - \sum_{e \in \Gamma_o(v)} x_e = \begin{cases} \text{-1} & \text{if } v = s, \\ 1 & \text{if } v = d, \\ 0 & \text{otherwise.} \end{cases} \quad \forall v. \quad (7)$$

$$\sum_{e \in \Gamma_i(v)} y_e - \sum_{e \in \Gamma_o(v)} y_e = \begin{cases} \text{-1} & \text{if } v = s, \\ 1 & \text{if } v = d, \\ 0 & \text{otherwise.} \end{cases} \quad \forall v. \quad (8)$$

The link disjointness between the primary and secondary paths is satisfied by

$$x_e + y_e \leq 1 \quad \forall e. \quad (9)$$

A detailed explanation of how the cost vector is adjusted for each coding group scenario and an example is given in [13] for the dynamic provisioning problem. Those procedures are directly applicable to the pre-provisioning problem of the static traffic.

III. Analysis

Evaluation of the performance of extended diversity coding compared to conventional diversity coding relies on many different criteria. These are spare capacity percentage, restoration time, signaling complexity, and synchronization complexity. The increased coding flexibility usually improves the performance of diversity coding in terms of spare capacity percentage, restoration time, and synchronization complexity. In the worst case, extended diversity coding performs as well as conventional diversity coding in terms of the aforementioned criteria since it can duplicate diversity coding. On the other hand, the increased complexity can cause small signaling overhead. In the following sections, performance evaluation of these two techniques are presented with examples.

A. Capacity Efficiency Example

Coding flexibility of extended diversity coding enables some paths to be coded with the ones that are closer to them instead of the common protection path. This results in lower fiber miles used by the coding group which means higher capacity efficiency. In this section, an example is provided to show how coding flexibility is useful to save more capacity. Consider Fig. 2(a) and assume that there are 4 unit connection demands

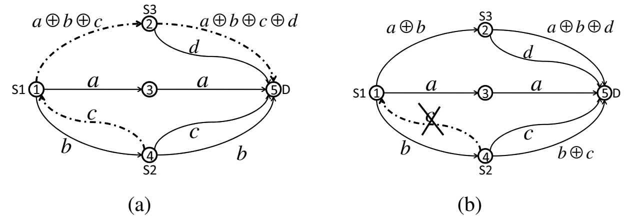


Fig. 2. Capacity efficiency comparison, (a)Diversity coding solution, (b) Extended diversity coding solution

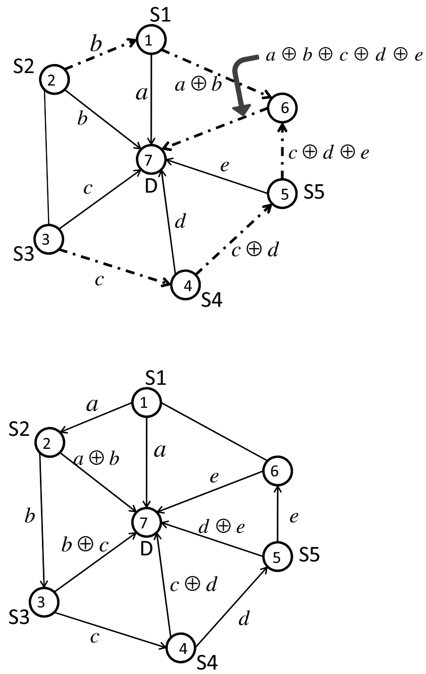
from source nodes to the common destination node D. Two units of these demands originate from node $S1$ carrying signals a and b. The signals c and d are transmitted from nodes $S2$ and $S3$ to node D, respectively. The solution of conventional diversity coding is shown in Fig. 2(a). Only one protection path is involved in coding operations. Therefore, there is only one path that serves as the parity link which is the dashed link $4-1-2-5$. In Fig. 2(b), the solution of extended diversity coding is shown. Instead of employing one single path for protection purposes, both $1-2-5$ and $4-5$ carry the coded data, which leads to further capacity savings over conventional diversity coding. In Fig. 2(b), the crossed link is the part that is not required in extended diversity coding solution with the help of coding flexibility. This example shows that extended diversity coding can be more capacity efficient than classical diversity coding.

B. Restoration Time Example

In some topologies, coding flexibility helps to shorten the paths which leads to significant savings in restoration time. In conventional diversity coding, the protection path has to traverse a long distance to code every signal in the coding group. That increases the synchronization delay occurring between the shortest and longest paths in the same coding group. The extended diversity coding solves this problem by enabling the primary paths to be involved in coding operations. As an example, consider Fig. 3(a) and assume that there are 5 unit connection demands one from each source node $S1$, $S2$, $S3$, $S4$, and $S5$ to destination node D. The links are bidirectional and have unit capacity. That is a typical scenario in metropolitan area networks. The routing and protection of these demands are solved by classical diversity coding in Fig. 3(a). The highest difference between propagation delays of the protection path and the primary paths are represented as PD in the following restoration time formula of the diversity coding

$$RT_{dc} = T + 2.X + PD.$$

Parameter T represents the detection time of a link failure and X is the processing time of a node. If we assume each link has unit length then the longest path in conventional diversity coding solution is 4 units. That means 3 units of propagation delay difference between the protection path and any primary path in the same coding group. Coding flexibility of extended diversity coding can significantly reduce the restoration time by shortening the longest path in the the solution. For the same problem, the solution of extended diversity coding is depicted in Fig. 3(b). The paths are only coded with the neighboring paths which decreases the maximum distance traversed by any path in the coding group. The longest path in the solution is



(b)

Fig. 3. Restoration time comparison, (a) Diversity coding solution, (b) Extended diversity coding solution

reduced to 2 unit lengths, which means the propagation delay difference is at most 1 unit.

C. Synchronization and Signaling Complexity

In conventional diversity coding, the protection path is supposed to code every signal in the coding group. Therefore, it needs to synchronize all of the data signals over the same channel. However, in extended diversity coding, the signals can be coded with a fewer number of signals, which decreases the number of synchronized data signals per each path. Referring to the example in Section III-B, in conventional diversity coding, the protection path needs to synchronize with all of the signals in the same coding group, namely a, b, c, d, e. However, the total number of coded signals in each path is limited to 2 when the extended diversity coding is employed. Therefore, coding flexibility can simplify the synchronization process.

One slight disadvantage of the extra coding flexibility in extended diversity coding is the requirement of a simple feedforward error signaling. Referring to the Fig. 3(b), if link $1-2$ fails, the receiver needs a notification message from node 2 in order to arrange the decoding structure properly. Otherwise, it will treat b as $a \oplus b$, which gives a wrong result.

IV. Simulation Results

In this section, we evaluate the effect of coding flexibility on spare capacity percentage (SCP) and restoration time (RT) numerically. We will present and compare the simulation results for both conventional and extended diversity coding schemes. For a fair comparison, the ILP-based joint capacity placement algorithm proposed in Section II-B is used for both of the techniques. Nevertheless, that algorithm is modified to be applicable to conventional diversity coding. Even if this

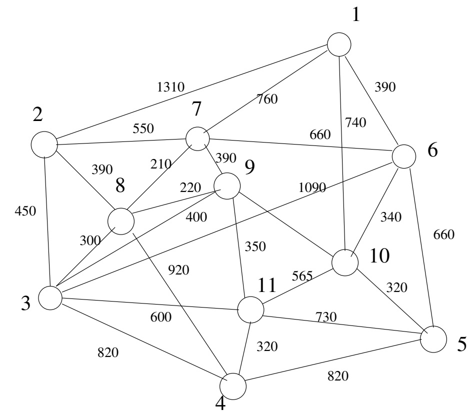


Fig. 4. European COST 239 network.

algorithm is suboptimal, it is useful to carry on a comparative study.

We have three test networks to analyze the comparative performance of these two techniques. These networks are COST 239 network [1], NSFNET network [14], and Smallnet network [15]. Their topologies are given in Fig. 4, Fig. 5, and Fig. 6, respectively. In those figures, numbers next to nodes are node indices, whereas the numbers next to the links are costs (lengths) of using that link. In the Smallnet network, link lengths are set to 100 kilometers. The traffic scenario for COST 239 network is taken from [15]. The traffic matrix of NSFNET network consists of 200 random connection demands, which are chosen using a realistic gravity-based model [16]. Each node in NSFNET network represents a U.S. state and their population is proportional to the weight of each node in the connection demand selection process. The traffic matrix of the Smallnet network consists of uniformly selected 200 random demands. The simulation results for three networks are given in Table I. The symbol F refers to the total number of connection demands. In these simulations, the objective was to minimize the total capacity. A summary of these results in terms of comparative analysis can be found in Table II where ACGS means average coding group size.

It is observed that in all three networks, extended diversity coding is more capacity efficient than conventional diversity coding, as expected. When the sparsity of the network increases, the improvement in the capacity efficiency decreases because coding flexibility is useful when there are more than two connections in the coding group. In a sparse network, most

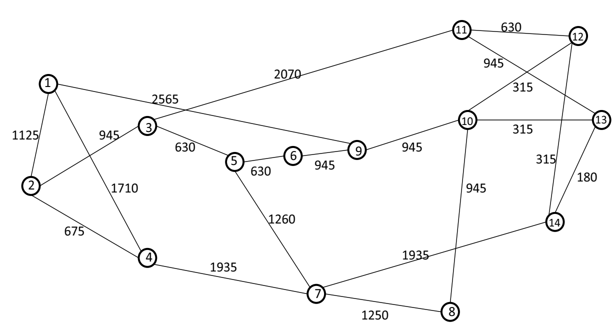


Fig. 5. NSFNET network.

TABLE I
COMPREHENSIVE SIMULATION RESULTS

Network	Method	Total Capacity	Shortest Capacity	SCP	RT	F	No. of coding groups
COST 239	Div. Coding	255178	125580	103.2%	5.19 ms	150	58
	Extended Div. Cod.	246149	125580	96.01%	4.25 ms	150	50
NSFNET	Div. Coding	933636	410570	127.4%	11.6 ms	200	105
	Extended Div. Cod.	911875	410570	122.1%	11.74 ms	200	99
Smallnet	Div. Coding	59430	28300	110.1%	1.27 ms	200	62
	Extended Div. Cod.	57392	28300	102.8%	1.17 ms	200	61

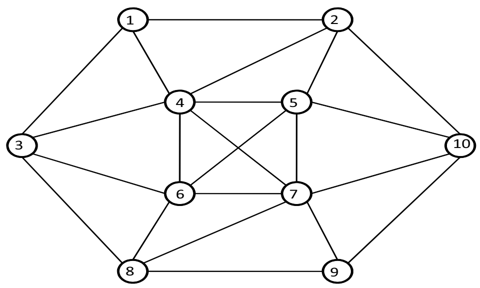


Fig. 6. Smallnet network.

TABLE II
THE EFFECT OF CODING FLEXIBILITY ON PERFORMANCE METRICS

Network	Nodal Degree	Improvement in		
		ACGS	SCP	RT
COST 239	4.72	16.72%	7.19%	18.2%
NSFNET	3	6.3%	5.3%	-1.2%
Smallnet	4.4	1.55%	7.3%	7.8%

of the coding groups only consist of two connections due to the link-disjointness criterion. In terms of restoration speed, there is an improvement in COST 239 and Smallnet networks but there is a small decline in the NSFNET network. Ideally, extended diversity coding can result in lower RT values in any network. However, the objective of the simulations was to minimize the total capacity. Therefore, extended diversity coding forms bigger coding groups with the help of coding flexibility. As a result, these enlarged coding groups usually have higher variance between the shortest and longest paths, which can increase the synchronization delay. As it is seen in Table II, extended diversity coding increases the coding group size up to 16.72%.

V. CONCLUSION

In this paper, we introduced a systematic approach to build extended diversity coding structures in which both primary and protection paths are coded. The added coding flexibility does not impair the decodability of the new coding groups. The advantages of this coding structure were shown earlier with some examples [10].

An accompanying ILP-based design algorithm was proposed to map these coding structures to arbitrary topologies. It is a heuristic algorithm with very low complexity because it handles the connection demands one-by-one from a static traffic matrix. The logic behind this algorithm is to protect the connection demands by adding them to the established coding groups. A coding group starts with only one connection and enlarged up to its topological limit without losing the decodability property.

We investigated the effect of the introduced coding flexibility over conventional diversity coding structures in terms of capacity efficiency and restoration speed. We ran simulations over 3 different realistic networks and traffic scenarios. Results indicate that extended diversity coding decreases SCP around 5-7% over all networks. It usually increases the restoration speed. The added coding flexibility makes use of topological diversity better and forms bigger coding groups. Therefore, at some cases, the bigger coding groups cancel the fundamental speed advantage of extended diversity coding.

REFERENCES

[1] W. D. Grover, *Mesh-Based Survivable Networks: Options and Strategies for Optical, MPLS, SONET, and ATM Networking*. Prentice-Hall PTR, 2004.

[2] M. Menth, M. Duelli, and J. Milbrandt, "Resilience analysis of packet-switched communication networks," *IEEE/ACM Trans. Netw.*, vol. 17, no. 6, p. 1, December 2009.

[3] J.-P. Vasseur, M. Pickavet, and P. Demeester, *Network Recovery: Protection and Restoration of Optical, SONET-SDH, IP, and MPLS*. Elsevier, 2004.

[4] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, "Survivable WDM mesh networks," *J. Lightwave Technol.*, vol. 21, no. 4, pp. 870–883, April 2003.

[5] W. Grover and D. Stamatelakis, "Cycle-oriented distributed preconfiguration: ring-like speed with mesh-like capacity for self-planning network restoration," in *Proc. ICC '98*, vol. 1, 1998, pp. 537–543.

[6] E. Ayanoglu, C.-L. I, R. D. Gitlin, and J. E. Mazo, "Diversity coding: Using error control for self-healing in communication networks," in *Proc. IEEE INFOCOM '90*, vol. 1, June 1990, pp. 95–104.

[7] ——, "Diversity coding for transparent self-healing and fault-tolerant communication networks," *IEEE Trans. Commun.*, vol. 41, pp. 1677–1686, November 1993.

[8] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, pp. 1204–1216, July 2000.

[9] S. Avci, X. Hu, and E. Ayanoglu, "Hitless recovery from link failures in networks with arbitrary topology," in *Proc. of the Information Theory and Applications Workshop*, February 2011, pp. 1–6.

[10] S. Avci and E. Ayanoglu, "Recovery from link failures in networks with arbitrary topology via diversity coding," in *Proc. IEEE GLOBECOM*, December 2011, pp. 1–6.

[11] ——, "Coded path protection: Efficient conversion of sharing to coding," in *Proc. IEEE ICC (to appear)*, June 2012.

[12] D. Traskov, N. Ratnakar, D. Lun, R. Koetter, and M. Médard, "Network coding for multiple unicasts: An approach based on linear optimization," in *Proc. IEEE ISIT*, July 2006, pp. 1758–1762.

[13] S. Avci and E. Ayanoglu, "Design algorithms for fast network restoration via diversity coding," in *Proc. of the Information Theory and Applications Workshop*, February 2012, pp. 1–7.

[14] M. B, S. Ramamurthy, D. Banerjee, and A. Mukherjee, "Some principles for designing a wide-area wdm optical network," *IEEE/ACM Trans. Netw.*, vol. 4, no. 5, pp. 684–706, October 1996.

[15] B. Wu, K. L. Yeung, and P.-H. Ho, "ILP formulations for p-cycle design without candidate cycle enumeration," *IEEE/ACM Trans. Netw.*, vol. 18, no. 1, pp. 284–295, February 2010.

[16] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," in *Proc. ACM SIGMETRICS*, June 2003.