Tensor Product Based Subspace Interference Alignment for Network Coding Applications

Viveck R. Cadambe, Syed A. Jafar†

***Abstract*— In this paper, we develop an interference alignment framework for multi-source (non-multicast) network coding applications. The framework developed here is based on using a tensor product structure for the network coding matrices at nodes in a network. The framework is presented in this paper in the specific context of exact repair for distributed storage, which is a multi-source network coding application. Using this framework, we generalize previously known MDS codes with optimal exact repair in distributed storage.**

I. Introduction

Interference Alignment has emerged as a fundamental technique to manage interference in multi-source network capacity problems in information theory. Among the various tools associated with interference alignment, the technique of asymptotic interference alignment [1] has had a particularly profound impact. The technique has been applied to various wireless settings including the interference channel [1]–[3], the X channel [4], and the compound broadcast channel [5]. In each of these networks, the technique of asymptotic interference alignment achieves the optimal number of *degrees of freedom* (a capacity approximation). A key requisite of the technique is the existence of diagonal channel gain matrices over which alignment is performed. Diagonal channel matrices are obtained in wireless communication scenarios by grouping multiple input and output symbols together, treating each of these groups as a vector (super-symbol), and coding over these vectors. As the size of the super-symbols increase, the size of the diagonal matrices increase and the extent of alignment increases. In the limit of arbitrarily large symbol extensions, the extent of alignment is "maximum" and hence achieves degrees of freedom in the given wireless network. (See [6] for a tutorial).

Recently, the technique of interference alignment has been interpreted as a technique for network coding (albeit not random). This interpretation led to the observation that interference alignment plays a significant role in the multi-source non-multicast *wireline* network capacity problems. Multi-source non-multicast wireline network capacity is a classical open problem in network information theory with applications to distributed storage, peer-to-peer networks, and content distribution over the internet (see for instance, [7], [24]). For this class of problems, the technique of *asymptotic* interference alignment has been applied in two contexts. First, it has been applied to multiple unicasts in a class of (wireline) networks in [8], [9]. Second, it has been applied to the exact repair problem in distributed storage [10], which is in turn, a specific non-multicast (wireline) network communication scenario. The approach of these references can be broadly described as follows. In a network, if all the nodes apply vector linear network coding, then outputs at the destinations of the network can be expressed as linear transformations of the input, much like a wireless channel. If the nodes use random diagonal matrices for network coding, then the transformation from the sources to the destinations are indeed random diagonal matrices. This approach thus paves the way for aligning interference using asymptotic interference alignment.

While linear network coding enables us to inherit techniques from wireless networks into wireline networks (and vice-versa, see [11], [12]), there is one fundamental difference between wired and wireline networks. In wireless networks, the linear transformation which represents the channel matrices are given by nature and hence cannot be controlled (to a large extent). In contrast, in wired networks, linear transformations are a function of the coding matrices which are design parameters. This means that the structure of the end to end channel matrices can be controlled (to the extent allowed by the topology of the network graph). The canonical wireline network coding application where this difference is especially highlighted is exact repair in distributed storage systems [10], [13]–[20]. Specifically, in these references, this extra control offered by the network coding application is exploited to create finite non-asymptotic alignment schemes without sacrificing the extent to which interference is aligned. This is in contrast with wireless communications where, in general, asymptotic alignment is necessary; limiting the size of the symbol extension (i.e., super-symbol) in wireless communications strictly limits the extent to which interference can be aligned [1], [21]–[23]. The ability to control the linear transformations in wireline network communication scenarios spawns a need to under-

†Dr. Viveck R. Cadambe is a joint postdoctoral fellow with the Research Lab of Electronics at Massachusetts Institute of Technology, and the Electrical & Computer Engineering Department at Boston University. Prof. Syed A. Jafar is with the Electrical Engineering and Computer Science Department at the University of California, Irvine, CA 92617, USA. The authors can respectively be contacted at viveck@mit.edu and syed@uci.edu

stand end-to-end linear transformations which are suitable for interference alignment. In this paper, we make progress in this context by developing a systematic approach to linear network code design in wireline networks to enable (non-asymptotic) interference alignment using a finite number of dimensions. We will particularly focus on the canonical problem of exact repair in distributed storage for exposition of our techniques and demonstration of its impact. We next explain the problem of exact repair.

II. Exact Repair in Distributed Storage

Consider k sources to be stored in a distributed storage system having n equal-capacity storage nodes. All k sources are assumed to be of equal size $L = M/k$ over a field $\mathbb{F}_q$ of size q. Source $i \in \{1, 2, \ldots, k\}$ is represented by the $L \times 1$ vector $\mathbf{a}_i \in \mathbb{F}_q^L$. Note here that M denotes the size of the total information stored in the distributed storage system, in terms of the number of elements over the field[1]. There are n nodes, each with a storage capacity of size L. The data stored in the n storage nodes can be interpreted as an (n, k) code of the k source symbols. Each node stores data of size L, i.e., each coded symbol of the (n, k) code is a $L \times 1$ vector. The data stored in node i is represented by $L \times 1$ vector $\mathbf{d}_i$, where $i = 1, 2, \ldots, n$. We assume that our code is linear so that $\mathbf{d}_i$ can be represented as

$$\mathbf{d}_i = \sum_{j=1}^{k} \mathbf{C}_{i,j} \mathbf{a}_j,$$

where $\mathbf{C}_{i,j}$ are $L \times L$ square matrices. Further, we restrict our codes to have a systematic structure, so that, for $i \in \{1, 2, \ldots, k\}$,

$$\mathbf{C}_{i,j} = \left\{ \begin{array}{cc} \mathbf{I} & j = i \\ \mathbf{0} & j \neq i \end{array} \right\}.$$

If the code storing the data is maximum-distance-separable (MDS), then, any k storage nodes can reconstruct all the k sources $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k$. Thus, an MDS code can protect against a failure of up to $(n - k)$ nodes. In this paper, we restrict our study to MDS codes, though the reader may note that most techniques developed here do not critically rely on the MDS property (though any claims of optimality rely on this property.) Therefore, we also do not focus on proving the MDS property of our codes. Such a proof may be found in the extended version of this paper [26].

The problem of interest here is the efficiency of repair when storage nodes fail. In this paper, we restrict ourselves to a single node failure. Further, we restrict ourselves to the case of failed systematic nodes. Now, when a single (systematic) node, say node 1 fails, the goal is to reconstruct the failed data, i.e., $\mathbf{a}_1$. A trivial strategy is to access any k nodes among $\mathbf{d}_2, \mathbf{d}_3, \ldots, \mathbf{d}_n$ and reconstruct $\mathbf{a}_1$. This is indeed possible because of the MDS property. The repair bandwidth with this repair strategy is equal to kL, since a total of k nodes are downloaded. The exact repair problem aims to improve the efficiency of this trivial repair strategy, and aims to construct codes with the minimum possible repair bandwidth. For the case considered, the solution of the exact repair problem is that the minimum repair bandwidth for an (n, k) MDS code is $\frac{(n-1)L}{n-k}$. Note that this repair bandwidth is smaller than kL implying that for any (n, k) the trivial repair strategy can improved through efficient code construction and repair strategies. The lower bound was shown using the cut set bound in [24] and the upper bound was shown through an asymptotic alignment based code construction in [10]. The latter reference constructed codes which achieve the minimum repair bandwidth in the limit of $L \to \infty$. Following this result, explicit finite (non-asymptotic) codes achieving this minimum bandwidth of $\frac{(n-1)L}{n-k}$ have been found in [16], [18], [19]. In this paper, we unify the approaches of [16], [18], [19] through a tensor product based subspace alignment framework. The tensor product based subspace alignment framework derived here is inspired by its first use for cellular networks in [25]. For ease of exposition, we restrict ourselves for the special case of $n = k + 2$. Due to space constraints, the extension of our codes to arbitrary (n, k) and details of the proofs are omitted. Here, we focus on conveying the overview and the intuition of the approach. The omitted details can be found in the extended version of this paper [26]. We begin by describing an outline of an optimal repair strategies for $(n = k + 2, k)$ MDS codes - the solutions of [16], [18], [19] fall within this outline.

Consider the case where a single systematic node, node $i \in \{1, 2, \ldots, k\}$, fails. The goal of repair is to construct $\mathbf{d}_i$ using the data in the surviving nodes $\{\mathbf{d}_j : j \neq i\}$. For repair, $\mathbf{V}_i \mathbf{d}_j$ is downloaded by the new node from node $j \in \{1, 2 \ldots, n\} - \{i\}$ where $\mathbf{V}_i$ is $\frac{L}{2} \times L$ matrix. Note that this strategy downloads a fraction of $\frac{1}{2}$ from each surviving node so that the total repair bandwidth is $\frac{L(n-1)}{2}$ as required. The linear combinations downloaded are of two types listed below.

1) The data downloaded from the surviving systematic nodes $j \in \{1, 2, \ldots, k\} - \{i\}$ contain no information of the failed node $\mathbf{a}_i$, i.e.,

$$\mathbf{V}_i \mathbf{d}_j = \mathbf{V}_i \mathbf{a}_j, j \in \{1, 2, \ldots, k\} - \{i\}.$$

Note that there $\frac{L}{2}$ such linear combinations of each interfering component $\mathbf{a}_j, j \in \{1, 2, \ldots, k\} - \{i\}$.

2) From each of the 2 parity nodes, $\frac{L}{2}$ linear combinations are downloaded. Therefore, a total of L linear combinations are downloaded from all the parity nodes. The L

[1]Note that we assume that $\mathcal{L}, \mathcal{M}$ are parameters of choice. This is a valid assumption for large amounts of information, since a large source can be split into several blocks, each of size $\mathcal{L}$. Each block can be coded separately using our constructions.

components of $\mathbf{a}_i$ have to be reconstructed using these L linear combinations of the form $\mathbf{V}_i\mathbf{d}_j, j = k+1, k+2$. For successful reconstruction of $\mathbf{a}_i$, the *interference* terms associated with $\mathbf{a}_j, j \in \{1, 2, \ldots, k\} - \{i\}$ contained in these linear combinations need to be cancelled completely (See Fig.1)

The goal of our solution will be to completely cancel the interference from the second set of L linear combinations, using the former set of linear combinations, and then to regenerate $\mathbf{a}_1$ using the latter L interference-free linear combinations (See Fig. 1).

In our solution the coding sub-matrices associated with the first parity node are all (scaled) identity matrices, i.e., $\mathbf{C}_{k+1,i} = \lambda_i \mathbf{I}_L$ for $i = 1, 2, \ldots, k$ where λ_i is a scalar over the field $\mathbb{F}_q$, so that

$$\mathbf{d}_{k+1} = \sum_{i=1}^{k} \lambda_{k+1,i}\mathbf{a}_i$$

Further, we denote $\mathbf{C}_{k+2,i} = \mathbf{H}_i$ so that

$$\mathbf{d}_{k+2} = \sum_{i=1}^{k} \mathbf{H}_i\mathbf{a}_i$$

While the above simplifications are restrictive, it turns out that they suffice to construct MDS codes for optimal repair. The simplifications help by revealing an interesting structure to the alignment constraints which are then solved by a tensor product based framework. With the above simplification, the L linear combinations downloaded from the two parity nodes can be expressed as

$$\begin{aligned} \mathbf{V}_i\mathbf{d}_{k+1} &= \sum_{j=1}^{k} \lambda_j \mathbf{V}_i\mathbf{a}_j && (1)\\ \mathbf{V}_i\mathbf{d}_{k+2} &= \sum_{j=1}^{k} \mathbf{V}_i\mathbf{H}_j\mathbf{a}_j. && (2) \end{aligned}$$

Note that we intend to completely cancel the impact of $\mathbf{a}_j$ using $\mathbf{V}_i\mathbf{a}_j$. To do this, we need

$$\text{rowspan}(\mathbf{V}_i\mathbf{H}_j) = \text{rowspan}(\mathbf{V}_i), j \in \{1, 2, \ldots, k\} - \{i\} \quad (3)$$

This ensures the desired interference alignment which hence ensures cancellation. After interference cancellation, we are left with L linear combinations of $\mathbf{a}_i$ of which $L/2$ are of the form $\lambda_i\mathbf{V}_i\mathbf{a}_i$ and the remaining $L/2$ are of the form $\mathbf{V}_i\mathbf{H}_i\mathbf{a}_i$. To ensure that repair is successful, we need to ensure that $\mathbf{a}_i$ is linearly resolvable from these L linear combinations. To do so, we need to ensure that

$$\begin{aligned} \text{rank}\begin{bmatrix} \mathbf{V}_i \\ \mathbf{H}_i\mathbf{V}_i \end{bmatrix} &= L, i = 1, 2, \ldots, k && (4)\\ \Rightarrow \text{rowspan}(\mathbf{V}_i) \cap \text{rowspan}(\mathbf{V}_i\mathbf{H}_i) &= \{0\} && (5) \end{aligned}$$

for $i = 1, 2, \ldots, k$

Thus, the goal is to find matrices $\mathbf{H}_i$ such that (3) and (5) are satisfied. Before proceeding, it is worth noting that as k increases, the problem becomes more and more constrained. In other words, if we have a solution to (3) and (5) for, say $k = 3$, then this automatically implies a solution for $k = 2$ but not vice-versa. For this reason, the problem was solved for $k = 2$ in [13] and only recently [10], [16], [18], [19] for an arbitrary value of k. Also note the similarities of the required constraints to the interference channel in [1]. This similarity was indeed exploited in [10] which used random diagonal matrices for $\mathbf{H}_i$ mimicking the wireless context. The reference then chose $\mathbf{V}_i$ using the asymptotic alignment technique. Here, we provide a tensor product based framework to solve this problem using a finite number of dimensions (i.e., for a finite value of L). For ease of exposition, we only focus on the case where $k = 3, n = 5$ here. The solution for arbitrary values of k and $n = k+2$ is very similar to the case where $k = 3$.

III. Tensor Product Based Subspace Alignment

The following lemma is a key observation in our development. In the lemma, the notation $\otimes$ is used to denote the Tensor (Kronecker) product between matrices.

Lemma 1: Let $\mathbf{H} = \mathbf{G}_1 \otimes \mathbf{G}_2 \otimes \ldots \otimes \mathbf{G}_r$. Also let $\mathbf{V} = \mathbf{U}_1 \otimes \mathbf{U}_2 \otimes \ldots \otimes \mathbf{U}_r$, where the matrix product $\mathbf{U}_i\mathbf{G}_i$ is defined. Then,

- $$\text{rowspan}(\mathbf{VH}) = \text{rowspan}(\mathbf{V})$$
  if and only if
  $$\text{rowspan}(\mathbf{U}_i\mathbf{G}_i) = \text{rowspan}(\mathbf{U}_i), \forall i = 1, 2, \ldots, r.$$
- $$\text{rowspan}(\mathbf{VH}) \cap \text{rowspan}(\mathbf{V}) = \{0\}$$
  if $\exists i \in \{1, 2, \ldots, r\}$ s.t.
  $$\text{rowspan}(\mathbf{U}_i\mathbf{G}_i) \cap \text{rowspan}(\mathbf{U}_i) = \{0\}$$
  where 0 is used to denote the row vector of zeroes (of the appropriate dimension).

An overview of the proof of the lemma is placed in the appendix. A complete proof can be found in the extended version of this paper [26]. The first property of the above lemma is useful to ensure alignment, i.e., ensure relations of the form (3). The second property listed in the lemma is useful to ensure relations of the form (5).

To see this, consider the special case where $k = 3$. We fix $L = 2^3 = 8$. Let us focus on the constraints associated with $\mathbf{V}_1$ alone. We need

$$\text{rowspan}(\mathbf{V}_1\mathbf{H}_2) = \text{rowspan}(\mathbf{V}_1\mathbf{H}_3) = \text{rowspan}(\mathbf{V}_1)$$

and

$$\text{rowspan}(\mathbf{V}_1\mathbf{H}_1) \cap \text{rowspan}(\mathbf{V}_1) = \{0\}$$

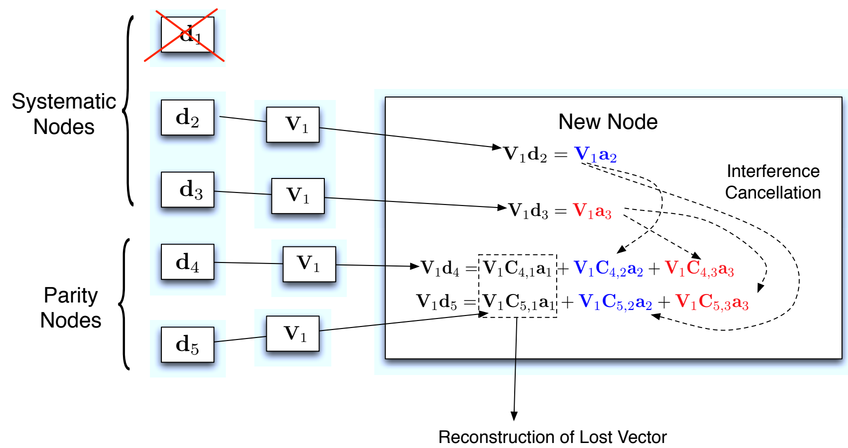

Fig. 1. Repair of first node for $n = 5, k = 3$. Equation (3) ensures that interference cancellation is possible. Equation (5) ensures that $\mathbf{a}_1$ can be reconstructed.

Suppose that $\mathbf{V}_1 = \mathbf{U}_1 \otimes \mathbf{I} \otimes \mathbf{I}$ where $\mathbf{U}_1$ is a 1×2 matrix and $\mathbf{I}$ is a 2×2 identity matrix. Also, suppose that

$$\mathbf{H}_i = \mathbf{H}_{i,1} \otimes \mathbf{H}_{i,2} \otimes \mathbf{H}_{i,3}$$

where $\mathbf{H}_{i,j}$ are all 2×2 matrices. We now need $\text{rowspan}(\mathbf{V}_1\mathbf{H}_2)$ to be equal to the rowspan of $\mathbf{V}_1$. To ensure this, the lemma stated above implies that we need $\text{rowspan}(\mathbf{U}_1) = \text{rowspan}(\mathbf{U}_1\mathbf{H}_{2,1})$. Similarly, we need $\text{rowspan}(\mathbf{U}_1) = \text{rowspan}(\mathbf{U}_1\mathbf{H}_{3,1})$. To ensure that $\text{rowspan}(\mathbf{V}_1) \cap \text{rowspan}(\mathbf{V}_1\mathbf{H}_1) = \{0\}$, it is sufficient to ensure that $\text{rowspan}(\mathbf{U}_1) \cap \text{rowspan}(\mathbf{U}_1\mathbf{H}_{1,1}) = \{0\}$. Note that all these constraints can be satisfied, for example, by setting

$$\mathbf{U}_1 = (0 \quad 1),$$

$$\mathbf{H}_{2,1} = \mathbf{H}_{3,1} = \mathbf{I}$$

$$\mathbf{H}_{1,1} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

In other words, with a tensor product structure imposed on all encoding and repair matrices, *all the constraints necessary for recovery of the first node* $\mathbf{a}_1$ *can be satisfied by choosing the first factor of the tensor products carefully*. Similarly, by careful manipulation of the second factor of the tensor products, the repair of $\mathbf{a}_2$ can be ensured. The third factor can be used to ensure repair of $\mathbf{a}_3$. Thus, one set of solutions for the required problem is

$$\mathbf{H}_1 = \mathbf{G} \otimes \mathbf{I} \otimes \mathbf{I}, \mathbf{V}_1 = \mathbf{U} \otimes \mathbf{I} \otimes \mathbf{I}$$

$$\mathbf{H}_2 = \mathbf{I} \otimes \mathbf{G} \otimes \mathbf{I}, \mathbf{V}_2 = \mathbf{I} \otimes \mathbf{U} \otimes \mathbf{I}$$

$$\mathbf{H}_3 = \mathbf{I} \otimes \mathbf{I} \otimes \mathbf{G}, \mathbf{V}_3 = \mathbf{I} \otimes \mathbf{I} \otimes \mathbf{U}$$

where, $\mathbf{G}$ is a 2×2 matrix and $\mathbf{U}$ is a 1×2 matrix chosen such that $\text{rowspan}(\mathbf{UG}) \cap \text{rowspan}(\mathbf{U}) = \{0\}$. Indeed, the solutions of [16], [18], [19] all have the above form! Further, a careful understanding of this procedure can lead to more general solutions for both the case of $n = k + 2$ and for arbitrary (n, k) as well (See [26].).

IV. Discussion

In this paper, we described a tensor product based alignment framework to achieve interference alignment using a finite number of dimensions. The framework provides a systematic approach to satisfy a set of alignment conditions, if they have a certain form. In particular, we use a tensor product of k matrices as the network coding matrices. The k_0th factor of the tensor product suffices to ensure the necessary alignment constraints for the k_0th message in the application considered. Lemma 1 provides a guideline to aligning interferers using this tensor product framework. The technique developed in this paper motivates several questions.

- A question of interest is whether this framework is as powerful as asymptotic interference alignment, in terms of the extent of alignment. For instance, the framework developed here only deals with repair of systematic nodes. It is not clear whether this approach leads insights into design of codes with optimal parity repair. It is worth noting that asymptotic alignment based codes of [10] perform optimally w.r.t. parity node repair as well. On a similar note, it is not clear whether this framework can be applied to create finite-block length achievable schemes for the communication scenarios studied in [8], [9] in place of asymptotic alignment.
- The reverse of the above question is also not clear. In other words, is not clear whether, from a network capacity perspective, asymptotic alignment is strictly more powerful than the tensor product based framework. While we are not aware of a network communication scenario where the tensor product based alignment approach achieves a better rate than asymptotic alignment,

we currently cannot preclude the existence of such a scenario.
- Finally, a question worth exploring is whether the insights of this work can be used to derive non-trivial achievable schemes for a different (or larger) class of (non-multicast) network communication scenarios.

APPENDIX

The proof of the lemma stems from combining the following three properties of the tensor product. The first following properties is well known. The second and the third properties can be derived from the multi-linearity and associative property of the tensor product

1) *Mixed Product Property:*

$$(\mathbf{P}_1 \otimes \mathbf{P}_2 \ldots \otimes \mathbf{P}_m)(\mathbf{Q}_1 \otimes \mathbf{Q}_2 \ldots \otimes \mathbf{Q}_m)$$
$$= (\mathbf{P}_1\mathbf{Q}_1) \otimes (\mathbf{P}_2\mathbf{Q}_2) \ldots \otimes (\mathbf{P}_m\mathbf{Q}_m)$$

2) *Invariance w.r.t. span:* Let $\mathbf{P}_i, \mathbf{Q}_i, i = 1, 2, \ldots, m$ be matrices such that the dimension of $\mathbf{P}_i$ is equal to the dimension of $\mathbf{Q}_i$. Also, let rowspan$(\mathbf{P}_i) \neq \mathbf{0}$, rowspan$(\mathbf{Q}_i) \neq \mathbf{0}, \forall i = 1, 2, \ldots, m$ where $\mathbf{0}$ represents the row vector whose entries are all equal to 0. Then, rowspan$(\mathbf{P}_i) =$ rowspan$(\mathbf{Q}_i), i = 1, 2, \ldots, m$, if and only if

$$\text{rowspan}(\mathbf{P}_1 \otimes \mathbf{P}_2 \ldots \otimes \mathbf{P}_m) = \text{rowspan}(\mathbf{Q}_1 \otimes \mathbf{Q}_2 \ldots \otimes \mathbf{Q}_m),$$

3) *Inheritance of linear independence:* Let $\mathbf{P}_i, \mathbf{Q}_i, i = 1, 2, \ldots, m$ be matrices such that the dimension of $\mathbf{P}_i$ is equal to the dimension of $\mathbf{Q}_i$. Now, suppose that rowspan$(\mathbf{P}_l) \cap$ rowspan$(\mathbf{Q}_l) = \{\mathbf{0}\}$ for some $l \in \{1, 2, \ldots, m\}$, i.e., each row of $\mathbf{P}_l$ is linearly independent of all the rows of $\mathbf{Q}_l$ for some $l \in \{1, 2, \ldots, m\}$. Then,

$$\text{rowspan}(\mathbf{P}_1 \otimes \mathbf{P}_2 \otimes \ldots \otimes \mathbf{P}_m) \cap \text{rowspan}(\mathbf{Q}_1 \otimes \mathbf{Q}_2 \otimes \ldots \otimes \mathbf{Q}_m)$$
$$= \{\mathbf{0}\}$$

REFERENCES

[1] V. Cadambe and S. Jafar, "Interference alignment and the degrees of freedom of the k user interference channel," *IEEE Trans. on Information Theory*, vol. 54, pp. 3425–3441, Aug. 2008.

[2] T. Gou and S. A. Jafar, "Capacity of a class of symmetric SIMO Gaussian interference channels within O(1)," *arxiv.org*, vol. abs/0905.1745, 2009. http://arxiv.org/abs/0905.1745.

[3] C. Wang, T. Gou, and S. A. Jafar, "Subspace alignment chains and the degrees of freedom of the three-user mimo interference channel," *CoRR*, vol. abs/1109.4350, 2011. http://arxiv.org/abs/1109.4350.

[4] V. R. Cadambe and S. A. Jafar, "Interference alignment and the degrees of freedom of wireless x networks," *IEEE Transactions on Information Theory*, vol. 55, pp. 3893 –3908, sept. 2009.

[5] T. Gou, S. A. Jafar, and C. Wang, "On the degrees of freedom of finite state compound wireless networks - settling a conjecture by weingarten et. al," 2009. http://arxiv.org/abs/0909.4177.

[6] S. A. Jafar, "Interference alignment: A new look at signal dimensions in a communication network," vol. 7, no. 1, pp. 1–136, 2011.

[7] S. Huang, A. Ramamoorthy, and M. Medard, "Minimum cost mirror sites using network coding: Replication versus coding at the source nodes," *IEEE Transactions on Information Theory*, vol. 57, pp. 1080 –1091, feb. 2011.

[8] A. Das, S. Vishwanath, S. Jafar, and A. Markopoulou, "Network coding for multiple unicasts: An interference alignment approach," in *2010 IEEE International Symposium on Information Theory Proceedings (ISIT)*, pp. 1878 –1882, June 2010.

[9] J. Han, N. Shroff, and C.-C. Wang, "Analysis of precoding-based intersession network coding and the corresponding 3-unicast interference alignment scheme," in *Proc. of 49st Annual Allerton Conf. on Communications, Control and Computing*, 2011.

[10] V. R. Cadambe, S. Jafar, and H. Maleki, "Distributed data storage with minimum storage regenerating codes - exact and functional repair are asymptotically equally efficient," *CoRR*, vol. abs/1004.4299, April 2010. http://arxiv.org/abs/1004.4299.

[11] N. Lee, J.-B. Lim, and J. Chun, "Degrees of freedom of the mimo y channel: Signal space alignment for network coding," *IEEE Transactions on Information Theory*, vol. 56, pp. 3332 –3342, july 2010.

[12] I. Maric, A. Goldsmith, and M. Medard, "Analog network coding in the high-snr regime," in *2010 IEEE Wireless Network Coding Conference (WiNC)*, pp. 1 –6, june 2010.

[13] Y. Wu and A. Dimakis, "Reducing repair traffic for erasure coding-based storage via interference alignment," in *IEEE International Symposium on Information Theory*, pp. 2276 –2280, 28 2009-july 3 2009.

[14] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramachandran, "Explicit codes minimizing repair bandwidth for distributed storage," *CoRR*, vol. abs/0908.2984, 2009. http://arxiv.org/abs/0908.2984.

[15] C. Suh and K. Ramchandran, "Exact regeneration codes for distributed storage repair using interference alignment," *CoRR*, vol. abs/1001.0107, 2010. http://arxiv.org/abs/1001.0107.

[16] I. Tamo, Z. Wang, and J. Bruck, "MDS array codes with optimal rebuilding," *CoRR*, vol. abs/1103.3737, 2011. http://arxiv.org/abs/1103.3737.

[17] Z. Wang, I. Tamo, and J. Bruck, "On codes for optimal rebuilding access," *CoRR*, vol. abs/1107.1627, July 2011. http://arxiv.org/abs/1107.1627.

[18] V. R. Cadambe, C. Huang, and J. Li, "Permutation code: Optimal exact-repair of a single failed node in MDS code based distributed storage systems," *Proceedings of IEEE Symposium on Information Theory (ISIT)*, July 2011.

[19] D. S. Papailiopoulos and A. G. Dimakis, "Distributed storage codes through hadamard designs," *2011 IEEE Symposium on Information Theory (ISIT)*, July 2011.

[20] D. S. Papailiopoulos, A. G. Dimakis, and V. R. Cadambe, "Distributed storage codes through hadamard designs," *To be presented in ISIT 2011*, july 2011.

[21] G. Bresler and D. Tse, "3 user interference channel: Degrees of freedom as a function of channel diversity," in *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, pp. 265 –271, Oct. 2009.

[22] G. Bresler, D. Cartwright, and D. Tse, "Settling the feasibility of interference alignment for the mimo interference channel: the symmetric square case," *CoRR*, vol. abs/1104.0888, 2011.

[23] M. Razaviyayn, G. Lyubeznik, and Z.-Q. Luo, "On the degrees of freedom achievable through interference alignment in a mimo interference channel," *CoRR*, vol. abs/1104.0992, 2011.

[24] A. Dimakis, P. Godfrey, M. Wainwright, and K. Ramchandran, "Network coding for distributed storage systems," in *IEEE INFOCOM*, pp. 2000 –2008, may 2007.

[25] C. Suh and D. Tse, "Interference alignment for cellular networks," in *Communication, Control, and Computing, 2008 46th Annual Allerton Conference on*, pp. 1037 –1044, Sept. 2008.

[26] V. R. Cadambe, C. Huang, S. A. Jafar, and J. Li, "Optimal repair of MDS codes in distributed storage via subspace interference alignment," 2011. Preprint available on author's website, http://newport.eecs.uci.edu/∼vcadambe.