Exploiting Symmetry in Computing Polyhedral Bounds on Network Coding Rate Regions

Jayant Apte, John MacLaren Walsh
Drexel University, Dept. of ECE, Philadelphia, PA 19104, USA
jsa46@drexel.edu, jwalsh@coe.drexel.edu

***Abstract*—We propose an algorithm for computing polyhedral bounds on the rate regions of multi-source multi-sink network coding instances given the knowledge of symmetries of the instance as captured by the network symmetry group. We show how the network symmetry group can be interpreted as a group of symmetries of a polyhedron, which in turn enables the use techniques for exploiting symmetry in polyhedral computation to reduce the complexity of calculating the rate region. We apply these techniques to the polyhedral projection algorithm `chm` to list only those facets and extreme points of a polyhedral bound on rate region that are inequivalent under the action of the network symmetry group. Additionally, a generalization of this algorithm that can exploit richer super-groups of polyhedral symmetries, the restricted affine symmetry groups, is discussed.**

***Index Terms*—network coding, symmetry, polyhedral projection**

I. Introduction

An implicit characterization of the rate region of multi-source multi-sink network coding over directed acyclic graphs (MSNC) was provided by Yan et. al. [1] in terms of the entropy function region Γ_n^*. For $n \geq 4$, however, Γ_n^* and its closure $\overline{\Gamma_N^*}$ are, to date, unknown. Explicit outer and inner polyhedral bounds on rate regions can be obtained for moderate sized instances of MSNC problem using polyhedral computation techniques [2]–[4] and isomorph free exhaustive generation techniques [5] respectively. When the explicit inner and outer bounds match, we obtain an explicit expression of the exact rate region of the MSNC instance.

Computing such polyhedral bounds on a MSNC rate region amounts to polyhedral projection from a space whose dimension is exponential in n to a space whose dimension is linear in n. The polyhedral projection algorithm Convex Hull Method (CHM), first proposed in [6], is well suited to such projection problems with a small projection dimension relative to original dimension, including the calculation of non-Shannon information inequalities [7] and rate regions for Multilevel Diversity Coding Schemes [4], [8]. Nevertheless, even with CHM, the process of calculating these bounds is highly complex, and, in order to enable the calculation of rate regions for the largest networks possible, it is imperative to exploit symmetry to reduce its complexity.

Recently, in [9], the notion of symmetry in network coding was formalized through the definition of a network symmetry group (NSG). The NSG for a MSNC instance with a total of n source and edge random variables is a subgroup of S_n capturing the symmetries imparted to its network codes and rate regions by the underlying directed acyclic graph.

In this paper, we show how to reduce the complexity of CHM, as applied to the calculation of outer bounds on MSNC rate regions, by exploiting the knowledge of the network's NSG. We begin in §II with a review of polyhedral symmetries and network symmetry groups, followed by a detailed description in §III of our implementation [10] of the CHM algorithm, which includes some improvements over the original [6]. These ingredients enable us in §IV to provide a symmetry exploiting variant of CHM. §IV-D then demonstrates how to apply the resulting algorithm to compute network coding rate region outer bounds for a series of example networks.

II. Symmetries of Polyhedral Bounds

An instance of a MSNC problem constsis of a directed acyclic graph $\mathcal{G} = (V, E)$, source and sink sets $\mathcal{S}, \mathcal{T} \subseteq V$, and sink demand function $\beta : \mathcal{T} \to 2^{\mathcal{S}} \setminus \emptyset$, with $n = |S| + |E|$. The goal is for every source $s \in \mathcal{S}$ to send, via a blocked message across all time, at every time instant an average of ω_s bits to be successfully decoded (i.e. with diminishing probability of error as the block length grows) at all sink nodes t such that $s \in \beta(t)$. This communication is to be carried out by sending no more than R_e bits per time instant over each edge $e \in E$, with each node $v \in V$ only allowed to receive messages on incoming edges and encode them into messages on outgoing edges. Collecting the source rates $\boldsymbol{\omega} = [\omega_s | s \in \mathcal{S}]$ and edge rates $\boldsymbol{r} = [R_e | e \in E]$, the *network coding rate region* is a convex cone which is the closure of the set of vectors $[\boldsymbol{\omega}^T, \boldsymbol{r}^T]^T$ that are simultaneously possible under some code. It has been shown [1], [9] that the problem of computing an outer or inner bound to a network coding rate region can be expressed as the polyhedral projection

$$\mathcal{R}_a = \mathsf{proj}_{|\mathcal{S}|+|E|}(\Gamma_n^a \cap \mathcal{L}_{123}^{\cap}), \quad a \in \{\mathsf{in}, \mathsf{out}\}. \tag{1}$$

Here, the sets are regarded as subsets of $\mathbb{R}^{2^n - 1 + n}$, with representative vectors $[\boldsymbol{\omega}, \boldsymbol{r}, \boldsymbol{h}]$, where $\boldsymbol{h}$ is indexed by the set $2^{S \cup E} \setminus \emptyset$ in the order of a binary indicator/counter. $\mathcal{L}_{123}^{\cap}$ reflects the linear constraints that $\forall i \in \mathcal{V}, \forall e \in E,\ \forall s \in \mathcal{S}$

$$\sum_{s \in \mathcal{S}} h_s = h_{\mathcal{S}},\ h_{\mathsf{In}(i)} = h_{\mathsf{In}(i) \cup \mathsf{Out}(i)},\ h_e \leq R_e,\ h_s \geq \omega_s, \tag{2}$$

with $\mathsf{In}(i)/\mathsf{Out}(i)$ the incoming/outgoing edges and sources to node i respectively. Γ_n^a is the Cartesian product of $\mathbb{R}_{\geq 0}^{|S|+|E|}$ (for the dimensions $\boldsymbol{\omega}, \boldsymbol{r}$), and an outer/inner bound for the region of entropic vectors $\bar{\Gamma}_N^*$ [1], [8] for $a = \mathsf{out}/\mathsf{in}$, resp.

A. Network Symmetry Groups

A *network symmetry*, as defined in [9], is special type of permutation of the set $\mathcal{S} \cup E$, i.e. a special type of bijection $\sigma : \mathcal{S} \cup E \to \mathcal{S} \cup E$. Under the action of this permutation, the vector $[\boldsymbol{\omega}^T, \boldsymbol{r}^T, \boldsymbol{h}^T]^T$ is mapped to a new vector $\sigma([\boldsymbol{\omega}^T, \boldsymbol{r}^T, \boldsymbol{h}^T]^T) = [\tilde{\boldsymbol{\omega}}^T, \tilde{\boldsymbol{r}}^T, \tilde{\boldsymbol{h}}^T]^T$ with $\tilde{\omega}_s = \omega_{\sigma(s)},\ \forall s \in \mathcal{S},\ \tilde{r}_e = r_{\sigma(e)},\ \forall e \in E$ and $\tilde{h}_{\mathcal{A}} = h_{\sigma(\mathcal{A})},\ \forall \mathcal{A} \subseteq \mathcal{S} \cup E$. Any reasonable inner or outer bound on entropy should be symmetric in the labels of the random variables and should satisfy $\sigma(\Gamma_n^a) = \Gamma_n^a$ for any such permutation σ. Any such permutation that also leaves $\mathcal{L}_{123}^{\cap}$ setwise invariant, i.e. with $\sigma(\mathcal{L}_{123}^{\cap}) = \mathcal{L}_{123}^{\cap}$ is called a *network symmetry*, and the collection of all network symmetries form the *network symmetry group* (NSG) with the operator of composition. Because the independence requirement for the sources must be stabilized under this requirement, any such NSG must satisfy $\sigma(\mathcal{S}) = \mathcal{S}$ and $\sigma(E) = E$. [9] shows that the NSG for the network coding problem $(\mathcal{G}, \mathcal{S}, \mathcal{T}, \beta)$ can be calculated as the subgroup of the automorphism group of the line graph of a circulation graph formed from $\mathcal{G}$ and β that setwise stabilizes $\mathcal{S}$ and E. Fig. 1 provides several small examples of NSGs.

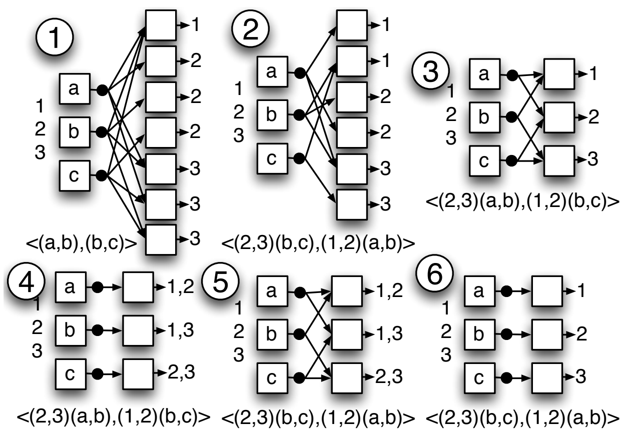

Figure 1: Six instances of 3-source 3-encoder I-DSC problem that have NSG of order 6. The NSGs were computed via a construction given in [9] using graph automorphism algorithms implemented in SageMath [11]. The generators of respective NSGs are specified below each figure, written in the form of permutations of subscripts of random variable associated with sources ($\{1, 2\}$) and encoders ($\{a, b, c\}$)

B. Polyhedral Symmetries

In what follows, boldface capital and small letters denote matrices and column vectors resp. and $\mathbb{N}_n$ refers to the set $\{1, \dots, n\}$. A polyhedron $\mathcal{P} \subseteq \mathbb{R}^d$ [12] can be represented either as the intersection of a finite number of closed halfspaces, $\mathcal{P} = \mathcal{P}(\mathbf{H}, \mathbf{z}) = \{\mathbf{x} \in \mathbb{R}^d | \mathbf{H}\mathbf{x} + \mathbf{z} \geq \mathbf{0}\}$, referred to as its inequality representation, or, equivalently, as the sum of the convex hull of a finite set of extreme points and the conic hull of a finite set of extreme rays, $\mathcal{P} = \mathrm{conv}(\mathbf{V}) + \mathrm{cone}(\mathbf{Y}), \mathbf{V} \in \mathbb{R}^{d \times t}, \mathbf{Y} \in \mathbb{R}^{d \times t'}$, referred to as its extremal representation. Here $\mathrm{conv}(\cdot)$ and $\mathrm{cone}(\cdot)$ refer to the convex and conic hull of the column vectors of $\mathbf{V}$ and $\mathbf{Y}$, respectively. If $t' = 0$, we call the polyhedron a *polytope* while if $t = 0$, we call it a *polyhedral cone*. We denote homogenization of a polyhedron (see [12]) as $\mathsf{homog}(\mathcal{P})$ and the polar of a polyhedral cone as $\mathcal{P}^\circ$ ([13], §14). For a polyhedral cone $\mathcal{P} = \mathcal{P}(\mathbf{A}, \mathbf{0}) = \mathrm{cone}(\mathbf{Y})$, the polar $\mathcal{P}^\circ = \mathcal{P}(\mathbf{Y}^T, \mathbf{0}) = \mathrm{cone}(\mathbf{A}^T)$. For such a cone, $\mathbf{A}, \mathbf{Y}$ are said to form a *double descriptions (DD) pair* [14].

Given a polyhedral cone $\mathcal{P}$ with extreme ray representatives the columns of $\mathbf{Y}$ labeled by set $\mathbb{N}_K$, the *restricted symmetries* [15], of $\mathcal{P}$ can be defined as those bijections $\sigma : \mathbb{N}_K \to \mathbb{N}_K$ such that there exists a $d \times d$ matrix $\mathbf{T} \in \mathbb{R}^{d \times d}$ for which $\mathbf{y}_{\sigma(i)} = \mathbf{T}\mathbf{y}_i$ for all $i \in [K]$.

The set of all such restricted symmetries of a polyhedral cone $\mathcal{P}$ form a group, with matrix multiplication as its operation, called the *restricted symmetry group* (RSG). The RSG can be computed via the automorphism group of an appropriately constructed edge-colored graph [15], and can be applied to general polyhedra through their homogenization into polyhedral cones. For full dimensional polytopes $\mathcal{P}$ with extreme points the columns of $\mathbf{V}$, the RSG of homog$(\mathcal{P})$, denoted by $G_{rs,\mathrm{homog}(\mathcal{P})}$, is equivalent (isomorphic) to the *affine symmetry group* (ASG) of $\mathcal{P}$, defined as $G_{a,\mathcal{P}} = \{[\mathbf{b}, \mathbf{T}] \in \mathbb{R}^{d \times d+1} | \mathbf{T} \in GL_d(\mathbb{R}),\ \mathbf{T}\mathcal{P} + \mathbf{b} = \mathcal{P}\}$, where $GL_d(\mathbb{R})$ is the general linear group.

C. NSGs and polyhedral symmetries

Since the NSG induces a permutation of the vectors $[\boldsymbol{\omega}^T, \boldsymbol{r}^T, \boldsymbol{h}^T]^T \in \Gamma_n^a \cap \mathcal{L}_{123}^{\cap}$ it forms a subgroup of the RSG of $\Gamma_n^a \cap \mathcal{L}_{123}^{\cap}$ associated with affine transformations $[\mathbf{b}, \mathbf{A}]$ that are linear (i.e. $\mathbf{b} = \mathbf{0}$) and with $\mathbf{A}$ the permutation matrix (whose columns are permuted columns of the identity matrix) such that $\sigma([\boldsymbol{\omega}^T, \boldsymbol{r}^T, \boldsymbol{h}^T]^T) = [\tilde{\boldsymbol{\omega}}^T, \tilde{\boldsymbol{r}}^{,T}, \tilde{\boldsymbol{h}}^T]^T$ as previously defined is equal to $[\tilde{\boldsymbol{\omega}}^T, \tilde{\boldsymbol{r}}^{,T}, \tilde{\boldsymbol{h}}^T]^T = \mathbf{A}[\boldsymbol{\omega}^T, \boldsymbol{r}^T, \boldsymbol{h}^T]^T$. Since the algorithms described in later sections are described for polytopes, we need to bound $\Gamma_n^a \cap \mathcal{L}_{123}^{\cap}$ thereby transforming the problem into a polytope projection problem instead of a polyhedral cone projection problem. Fortunately, an unbounded polyhedron $\mathcal{C}$ can be transformed to create a polytope $\mathcal{B}(\mathcal{C})$ such that the projection of the unbounded polyhedron $\mathcal{C}$ can be obtained from projection of $\mathcal{B}(C)$. While there are several such transformations, including the one in [6], we describe a transformation that is more efficient in terms of dimension of $\mathcal{B}(\mathcal{C})$ (=d) and that also preserves restricted symmetries arising from Network Symmetry Groups.

Let $\mathbf{H}\mathbf{x} \geq \mathbf{0}$ be the inequality description associated with a polyhedral cone $\mathcal{C}$. This cone can be transformed into a polytope $\mathcal{C}' = \{\mathbf{x} \in \mathbb{R}^d_{\geq \mathbf{0}} \mid \mathbf{H}\mathbf{x} \geq \mathbf{0} \ \wedge\ \mathbf{1}^t\mathbf{x} \leq 1\}$. If $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ are vectors in the directions of the extreme rays of $\mathcal{C}$ then the set of extreme points of $\mathcal{C}'$ is $\mathcal{V} = \{\frac{\mathbf{x}_1}{\mathbf{1}^t\mathbf{x}_1}, \dots, \frac{\mathbf{x}_n}{\mathbf{1}^t\mathbf{x}_n}\} \cup \{\mathbf{0}\}$. Futhermore, the set of extreme rays of $\mathsf{proj}_k(\mathcal{C})$ is equal to the conic independent subset of rays in the directions of the non-zero extreme points of $\mathsf{proj}_k(\mathcal{C}')$.

A downside of utilizing the boundedness transform to solve the cone projection problem is that potentially there are extreme points of $\mathsf{proj}_k(\mathcal{C}')$ that are not conic independent, and thus not extreme rays of $\mathsf{proj}_k(\mathcal{C})$. After CHM completes computing the projection $\text{proj}_k(\mathcal{C}')$, its non-zero extreme points are relabeled as the extreme rays of $\text{proj}_k(\mathcal{C})$, and the set of homogenous inequalities of $\text{proj}_k(\mathcal{C}')$ are relabeled as the inequalities of $\text{proj}_k(\mathcal{C})$.

When this boundedness transformation is applied to $\Gamma_n^a \cap \mathcal{L}_{123}^{\cap}$ we get a polytope $\mathcal{P}$. Each non-zero extreme point of $\mathcal{P} = \Gamma_n^a \cap \mathcal{L}_{123}^{\cap} \cap \{\mathbf{1}_d^T[\boldsymbol{\omega}^T, \boldsymbol{r}^T, \boldsymbol{h}^T]^T \leq 1\}$ is the direction of an extreme rays of $\Gamma_n^a \cap \mathcal{L}_{123}^{\cap}$ scaled to sum to one, and vice versa. Since the sum of an extreme ray direction is invariant under a permutation from the NSG, each NSG yields a permutation of the extreme points of $\mathcal{P}$. Hence, the NSG gives a subgroup of the ASG of $\mathcal{P}$ based on the action of the network symmetries on the vectors $[\boldsymbol{\omega}^T, \boldsymbol{r}^T, \boldsymbol{h}^T]^T$. Furthermore, since each network symmetry is a permutation σ of the set $\mathcal{S} \cup E$, and the projected polytope $\text{proj}_k(\mathcal{P})$ with $k = |\mathcal{S}| + |E|$, the NSG also forms a subgroup of the ASG $\text{proj}_k(\mathcal{P})$, again associated with linear transformations with permutation matrices. In this manner, the NSG gives both a subgroup G of the ASG for $\text{proj}_k(\mathcal{P})$ and a subgroup G_o of the ASG for $\mathcal{P}$, which, as we shall show in upcoming sections, can be utilized together with to calculate the rate region bound (1) with substantially lower complexity than the non-symmetry exploiting CHM.

III. The Convex Hull Method

Convex Hull Method (CHM) [6] is an algorithm to project a polytope $\mathcal{P}$ in $\mathbb{R}^d$ by building successively better inner bounds to the projected polytope $\text{proj}_k(\mathcal{P}) \triangleq \left\{\mathbf{x} \in \mathbb{R}^k \,\middle|\, \exists \mathbf{y} \in \mathbb{R}^{d-k} \text{ such that } (\mathbf{x}, \mathbf{y}) \in \mathcal{P}\right\}$ via the solution of carefully selected linear programs over $\mathcal{P}$. The pseudocode for our implementation of [10] can be found in [16], and the sequence of bounds the algorithm creates when projecting the hypercube is depicted in Fig. 2.

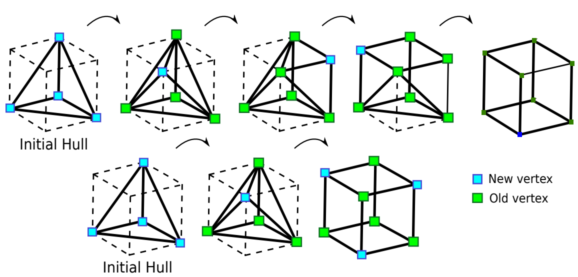

Figure 2: (top) Sequence of inner bounds produced by CHM and (bottom) sequence of inner bounds produced by symCHM while projecting a 4-cube to 3 dimensions given the knowledge all 3! permutations of basis vectors of $\mathbb{R}^3$ form restricted symmetries of projection

Note that we assume that the input polyhedron $\mathcal{P}$ is bounded and full-dimensional, so that the dimension of its affine hull is d. Tests for full-dimensionality and methods for the elimination of any redundant variables and inequalities of the input can be be implemented based on guidelines in [17].

The algorithm relies on the fact that, if $\mathbf{c} \in \mathbb{R}^k$, then an extreme point that attains the solution to the linear program with cost vector $\mathbf{c}$ over $\text{proj}_k(\mathcal{P})$ can be found by projecting the extreme point in $\mathcal{P}$ attaining to the solution of the linear program with cost vector $[\mathbf{c}^T, \mathbf{0}_{d-k}^T]^T$ over $\mathcal{P}$, so that $\min_{\mathbf{x} \in \text{proj}_k \mathcal{P}} \mathbf{c}^T \mathbf{x} = \min_{\mathbf{y} \in \mathcal{P}} [\mathbf{c}^T, \mathbf{0}_{d-k}^T] \mathbf{x}$, and $\arg\min_{\mathbf{x} \in \text{proj}_k \mathcal{P}} \mathbf{c}^T \mathbf{x} = \text{proj}_k \left(\arg\min_{\mathbf{y} \in \mathcal{P}} [\mathbf{c}^T, \mathbf{0}_{d-k}^T] \mathbf{x}\right)$.

In order to obtain the initial inner bound for the process (proc. initialhull [16]), we first obtain two extreme points of $\text{proj}_k \mathcal{P}$ by $\text{proj}_k \arg\min_{\mathbf{y} \in \mathcal{P}} [-1, \mathbf{0}_{d-1}^T] \mathbf{y}$ and $\text{proj}_k \arg\min_{\mathbf{y} \in \mathcal{P}} [1, \mathbf{0}_{d-1}^T] \mathbf{y}$. We then select a hyperplane $\left\{\mathbf{x} \in \mathbb{R}^k \,\middle|\, \mathbf{c}^T \mathbf{x} = b\right\}$ containing these points (proc. hyperplane [16]), and an additional new extreme points is obtained by either $\text{proj}_k \arg\min_{\mathbf{y} \in \mathcal{P}} [\mathbf{c}, \mathbf{0}_{d-k}] \mathbf{y}$ or $\text{proj}_k \arg\min_{\mathbf{y} \in \mathcal{P}} [-\mathbf{c}, \mathbf{0}_{d-k}] \mathbf{y}$. A new hyperplane is found containing all of the extreme points and the process is repeated a total of k times, yielding $k+1$ extreme points giving a full dimensional initial inner bound. Given set V containing $k+1$ convex-independent points in $\mathbb{R}^k$, computing the inequality description of the initial hull $\text{conv}(V)$ corresponds to a $k+1 \times k+1$ matrix inversion (proc. facets [16]).

At each stage of the algorithm, a DD pair is maintained for the current inner bound to $\text{proj}_k \mathcal{P}$. Each inequality in the inequality description of the inner bound carries with it a label, indicating whether or not it is terminal or non-terminal. The initial inner bound has all of its inequalities labelled as non-terminal. A non-terminal inequality $\mathbf{c}^T \mathbf{x} \geq b$ is then selected (proc. isterminal [16]), and the associated linear program $\min_{\mathbf{x} \in \mathcal{P}} [\mathbf{c}^T \mathbf{0}_{d-k}^T] \mathbf{x}$ is solved over the high dimensional polyhedron. If the solution obtained is b, the inequality is marked as terminal. Otherwise, the extreme point of $\mathcal{P}$, say $\mathbf{v}$, attaining the solution is projected to get a new extreme point $\text{proj}_k \mathbf{v}$ of $\text{proj}_k \mathcal{P}$. The DD pair of the inner bound is then updated (proc. updatehull [16]) by adding the new extreme point, viewing it as a new inequality in the polar cone a single DD algorithm update step [14] (proc. DDiteration [16]), and any new inequalities thus introduced are marked as non-terminal. Then a new non-terminal facet is selected and the process is repeated until all of the facets are labelled as terminal, at which point the inner bound has been proven equal to $\text{proj}_k \mathcal{P}$.

IV. Symmetry Exploiting CHM

In some polyhedral projection problems, including the motivating problem of outer bounding network coding rate regions, groups of symmetries of both the polyhedron to be projected and of its projection are known prior to the projection calculation. Knowledge of these symmetry groups can enable important complexity and memory reduction enhancements to polyhedral projection algorithms such as CHM. In this section we will detail how to achieve these improvements by describing a symmetry exploiting CHM. In keeping with the ordinary CHM algorithm, we will begin by assuming that we must project a polytope $\mathcal{P}$ to $\text{proj}_k \mathcal{P}$. Additionally, we also have groups G_o, G of known affine symmetries of $\mathcal{P}, \text{proj}_k(\mathcal{P})$, respectively, that are possibly subgroups of the full ASGs of these sets.

Before we describe symmetry exploiting CHM and detail the enhancements the symmetry knowledge enables, we define some terminology related to the action of ASGs on sets of vertices and facets of a polytope. The terminology that follows is described for an arbitrary polytope $\mathcal{P}$ whose ASG (or a subgroup of it G) is known. Let V and H be the set of vertices and facets respectively of $\mathcal{P}$. For an affine symmetry $g \in G$ and a vertex $\mathbf{v}$ of $\mathcal{P}$ denote by $\mathbf{v}^g$ to be the vertex to which $\mathbf{v}$ maps to under action of g and let $\mathbf{v}^G$, the *orbit* of $\mathbf{v}$ under G, be the set of all vertices to which $\mathbf{v}$ can map to under action of G. $\mathbf{v}^G$ contains all vertices that are *G-equivalent* to $\mathbf{v}$. The set of all orbits in $\mathcal{V}$ under action of G forms a partition of $\mathcal{V}$ and is denoted as $\mathcal{O}_V$. Since each facet is simply the convex hull of a collection of vertices, the action of the ASG can be extended to H i.e. we define the orbit of a facet $h \in H$ denoted as h^G and $\mathcal{O}_H$ to be set of all orbits of facets. The transversal $\mathcal{T}$ of a set of orbits $\mathcal{O}$ is a set containing one representative per orbit in $\mathcal{O}$, and transversals of $\mathcal{O}_V$ of $\mathcal{O}_H$ are denoted as $\mathcal{T}_V$ and $\mathcal{T}_H$ respectively. By storing transversals of the inequality and vertex orbit sets, we can substantially reduce the space required for working with highly symmetric polyhedra.

The next few subsections outline how symmetry exploiting CHM works, while additionally describing the each of the complexity improvements that the modifications enable.

A. Reducing the Number of LPs Solved

Just as in CHM, symmetry exploiting CHM (symCHM) builds a sequence of progressively better inner bounds to $\text{proj}_k(\mathcal{P})$. However, in symCHM, each inner bound obtained is selected to be symmetric under the action of G, and the set of inequalities and extreme rays in its DD pair is represented exclusively by transversals $\mathcal{T}_H, \mathcal{T}_V$. The transversal $\mathcal{T}_H$ of the facets of the inner bound carries with it an indicator variable indicating if it is terminal or non-terminal. At an intermediate step in the algorithm, a non-terminal facet $\mathbf{c}^T\mathbf{x} \geq b$ is selected from the current inner bound's facet transversal $\mathcal{T}_H$. Just as in CHM, the linear program $\min_{\mathbf{y}\in\mathcal{P}}[\mathbf{c}^T, \mathbf{0}_{d-k}^T]\mathbf{y}$ is solved, and if the result is b, the facet is marked as terminal. If the result is not b, the projection of the extreme point attaining the minimum, $\text{proj}_k \arg\min_{\mathbf{y}\in\mathcal{P}}[\mathbf{c}^T, \mathbf{0}_{d-k}^T]\mathbf{y}$, is added to the transversal $\mathcal{T}_V$. This act of adding this single extreme point $\mathbf{v}$ to the inner bound's vertex transversal has the same effect as having added the entire orbit $\mathbf{v}^G$ to the full list of extreme points in CHM. In ordinary CHM, to add each of these extreme points, $|\mathbf{v}^G|$ linear programs would have had to be solved to obtain these extreme points, but in symCHM, only one LP is required to obtain all of them.

B. Reducing the Number & Size of Double Descriptions Steps

When the new extreme point $\mathbf{v}$ is added to the transversal $\mathcal{T}_V$, the transversal of the inequalities $\mathcal{T}_H$ must be updated (proc. symupdatehull [16]) to reflect the new inequalities that the addition of the extreme points $\mathbf{v}^G$ to the symmetric inner bound creates (we call this new polytope the *symmetric improvement*). In ordinary CHM, this would have been done through of $|\mathbf{v}^G|$ steps of the DD method applied to the complete inequality description of the symmetric inner bound. However, based on Lemma 1, which is the same insight from which the incidence decomposition method [15] for representation conversion of symmetric polyhedra is derived, we can often utilize far fewer DD steps (proc. symDD [16]) to obtain the new facets that must be added to the transversal.

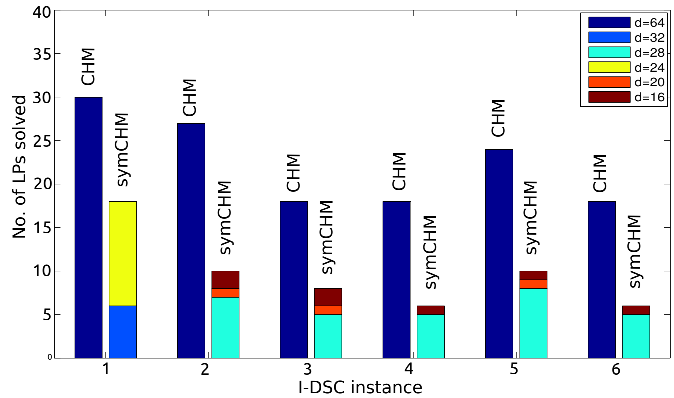

Figure 3: Comparison of number of LPs solved in chm vs symchm for computing Γ_n^{out} for non-isomorphic I-DSC instances with NSG of order 6. d is the dimension of LP solved. In this instance, every LP solved with symchm is of smaller dimension than those solved in ordinary chm.

Lemma 1. *Let $\mathcal{P}_k^{(\ell)}$ be an inner bound on $\text{proj}_k(\mathcal{P})$ whose ASG has G_p as a subgroup, and let $\mathbf{v}$ be a new vertex of a symmetric improvement $\mathcal{P}_k^{(\ell+1)}$. If, in $\mathcal{P}_k^{(\ell+1)}$, $\{f_1, \ldots, f_t\}$ is the set of facets incident to $\mathbf{v}$ then, $\{f_1^g, \ldots, f_t^g\}$ is the set of facets incident to $\mathbf{v}^g$.*

Lemma 1 ensures that as long as we calculate the facets of $\mathcal{P}_k^{(\ell+1)}$ incident to $\mathbf{v}$ correctly, and include any of these facets that are G-inequivalent into the new transversal $\mathcal{T}_H$ after removing those non-terminal inequalities that the new extreme points violate, the new facet transversal will reflect all of the G-inequivalent facets of $\mathcal{P}_k^{(\ell+1)}$. The key issue in calculating the facets incident to $\mathbf{v}$ in $\mathcal{P}_k^{(\ell+1)}$ correctly is that there may be some vertices in $\mathbf{v}^G \setminus \{\mathbf{v}\}$ that are adjacent to $\mathbf{v}$. To check to see if this is the case, and if so, which ones, passing to the homogenized polar $\mathcal{C} = \text{homog}(\mathcal{P}_k^{\ell})^{\circ}$, we can determine the set $\mathcal{A} = \left\{\mathbf{z} \in \mathbf{v}^G \setminus \{\mathbf{v}\} \,\middle|\, \min_{\mathbf{x}\in\mathcal{C}_{\mathbf{v}=,\setminus\mathbf{z}}}[1\ \mathbf{z}^T]\mathbf{x} < 0\right\}$ where

$$\mathcal{C}_{\mathbf{v}=,\setminus\mathbf{z}} = \mathcal{C} \cap \{[1\ \mathbf{v}^T]\mathbf{x} = 0\} \bigcap_{\mathbf{w}\in\mathbf{v}^G\setminus\{\mathbf{z},\mathbf{v}\}} \{[1\ \mathbf{w}^T]\mathbf{x} \geq 0\} \quad (3)$$

(see proc. repDD [16]). Next the rays of $\mathcal{C}_{\mathbf{v}=} = \mathcal{C} \cap \{[1\ \mathbf{v}^T]\mathbf{x} = 0\}$ are determined through an ordinary DD step adding $\{[1\ \mathbf{v}^T]\mathbf{x} \geq 0\}$ to $\mathcal{C}$. These rays are further refined by adding the inequalities $\{[1\ \mathbf{w}^T]\mathbf{x} \geq 0\}$ for each $\mathbf{w} \in \mathcal{A}$ if any, through $|\mathcal{A}|$ further DD steps. The new inequality transversal of $\mathcal{P}_k^{(\ell+1)}$ is created by removing any G-equivalent inequalities from $\mathcal{P}_k^{\ell}$'s removed (as rays in the homogenized polar) in these $|\mathcal{A}|+1$ DD steps, and by adding the representatives of the new rays introduced at the end of these $|\mathcal{A}|+1$ DD steps.

The consideration of symmetry in this step of updating the inequality description of the inner bound just described both reduced the number of DD steps required for CHM and their size. Indeed, only $|\mathcal{A}|+1$ DD steps must be performed to find the result of adding $|\mathbf{v}^G|$ new extreme points. Also, the size of each these DD steps is substantially smaller, since the cone $\mathcal{C}_{\mathbf{v}=}$ is being dealt with in the $|\mathcal{A}|$ latter DD rather than $\mathcal{C}$.

C. Reducing the Dimension of the LPs over $\mathcal{P}$

As explained previously, symmetry exploiting CHM creates its successively improving symmetric inner bounds to $\text{proj}_k(\mathcal{P})$ by solving linear programs over $\mathcal{P}$ of the form $\min_{\mathbf{y}\in\mathcal{P}}[\mathbf{c}^T, \mathbf{0}_{d-k}^T]\mathbf{y}$. It is thus of interest to learn how to exploit the ASG of $\mathcal{P}$ to reduce the complexity of solving these linear programs. It is known that symmetry can substantially simplify linear programs through dimensional reduction [18]. A key consideration in this complexity reduction is whether or not the symmetries of the constraint set $\mathcal{P}$ are shared with the cost vector, which in the instance of CHM & symCHM is always of the form $[\mathbf{c}, \mathbf{0}_{d-k}]$. As such, let $G_{o,\mathbf{c}}$ be the subgroup of the known ASG subgroup G_o that leaves the cost invariant, i.e. $G_{o,\mathbf{c}} = \left\{[\mathbf{b}, \mathbf{A}] \in G_o \,\middle|\, [\mathbf{c}^T\mathbf{0}_{d-k}^T](\mathbf{A} - \mathbb{I}_k) = \mathbf{0}_k, [\mathbf{c}^T\mathbf{0}_{d-k}^T]\mathbf{b} = 0\right\}$. The dimensionality reduction is achieved by observing that it suffices to consider in the optimization only the part of $\mathcal{P}$ that is fixed under the action of $G_{o,\mathbf{c}}$. Bearing this in mind, define the fixed space (which is a subspace of $\mathbb{R}^d$),

$$\mathsf{Fix}(G_{o,\mathbf{c}}) = \left\{\mathbf{y} \in \mathbb{R}^d \,|\mathbf{A}\mathbf{y} + \mathbf{b} = \mathbf{y},\ \forall[\mathbf{b}, \mathbf{A}] \in G_{o,\mathbf{c}}\right\}. \quad (4)$$

The key result is that

$$\min_{\mathbf{y}\in\mathcal{P}}[\mathbf{c}^T\mathbf{0}_{d-k}^T]\mathbf{y} = \min_{\mathbf{y}\in\mathcal{P}\cap\mathsf{Fix}(G_{o,\mathbf{c}})}[\mathbf{c}^T\mathbf{0}_{d-k}^T]\mathbf{y} \quad (5)$$

Whenever the symmetry group is rich enough so that $|G_{o,\mathbf{c}}| > 1$, the LP on the right of (5) is of lower dimension than the original, and hence it is desirable to solve the LP over $\mathcal{P}$ by instead solving the linear program over $\mathcal{P} \cap \mathsf{Fix}(G_{o,\mathbf{c}})$. By exploiting this dimensionality reduction whenever available, symCHM can further reduce complexity relative to CHM.

D. Example: Application to IDSC Rate Regions

Symmetry exploiting CHM can be utilized to calculate the projection of the Shannon or LP outer bound to the rate region of the IDSC instances in Fig. 1. Each of these instances have a NSG of order 6, which, as discussed in §II-C, yields a symmetry group of the same order for both the polyhedral cone to project, and the result of the projection when calculating the LP outer bound. Fig. 3 demonstrates the substantial reduction in computation afforded by exploiting these symmetries in the projection: for each of these networks, substantially fewer LPs are solved by symCHM than CHM, and those LPs that are solved are all of substantially lower dimension.

V. Conclusions and Future Work

This paper showed how to utilize the NSG to reduce the complexity of calculating polyhedral bounds on the network coding rate regions. To do this, a polytope projection algorithm, symCHM that can exploit known symmetry groups G_o, G of both the original polytope and its projection while building progressively better symmetric inner bounds to the projection was presented. This algorithm improved upon ordinary CHM algorithm in several ways. First of all, a substantial reduction in the amount of memory to represent the polyhedra was enabled by only storing one representative from each of the equivalence classes, under symmetry, of the inequalities or extreme rays under the action of the group. Second of all, the number of LPs to calculate the projection was drastically reduced by acting on every new extreme point by G, keeping the inner bound symmetric at every stage, and, when labeling a facet of the projection as terminal (and hence correctly shared with the projection), labeling all of its G-equivalent other facets as terminal. Third of all, when a new extreme point (and hence all of its G equivalent forms) has been added to the inner bound, the action of G was utilized to reduce the complexity of updating the inequality description of the inner bound. Finally, any symmetries in G_o that are shared with the cost vector in any LPs solved over it were utilized to reduce the dimension of the LP to solve. We then showed how to transform the network coding rate region problem into a form that can utilize this new projection algorithm. Since the polyhedral projection algorithm presented can exploit a supergroup to the NSG, the ASG, future work will determine a larger ASG subgroup of MSNC rate regions than the NSG.

Acknowledgments

NSF support under CCF-1016588 and CCF-1421828.

References

[1] X. Yan, R. Yeung, and Z. Zhang, “An implicit characterization of the achievable rate region for acyclic multisource multisink network coding,” *IEEE Trans. on Inform. Theory*, vol. 58, no. 9, pp. 5625–5639, 2012.

[2] C. Li, J. Apte, J. Walsh, and S. Weber, “A new computational approach for determining rate regions and optimal codes for coded networks,” in *IEEE Int. Symp. Network Coding (NetCod)*, 2013, pp. 1–6.

[3] C. Li, J. M. Walsh, and S. Weber, “Computational approaches for determining rate regions and codes using entropic vector bounds,” in *50th Annual Allerton Conference on Communication, Control and Computing, 2012*, Oct 2012, pp. 1 –9.

[4] Congduan Li, Steven Weber, and John MacLaren Walsh, “On Multilevel Diversity Coding Systems,” submitted on July 21, 2014. [Online]. Available: http://arxiv.org/abs/1407.5659

[5] J. Apte, C. Li, and J. Walsh, “Algorithms for computing network coding rate regions via single element extensions of matroids,” in *IEEE Int. Symp. Information Theory (ISIT)*, June 2014, pp. 2306–2310.

[6] C. Lassez and J.-L. Lassez, “Quantifier elimination for conjunctions of linear constraints via a convex hull algorithm,” in *Symbolic and Numerical Computation for Artificial Intelligence*, Donald, Kapur, and Mundy, Eds. Academic Press, 1993.

[7] W. Xu, J. Wang, J. Sun, “A projection method for derivation of non-Shannon-type information inequalities,” in *IEEE International Symposium on Information Theory (ISIT)*, 2008, pp. 2116 – 2120.

[8] R. W. Yeung, *Information Theory and Network Coding*. Springer, 2008.

[9] J. Apte and J. M. Walsh, “Symmetry in Network Coding,” in *Int. Symp. Information Theory (ISIT)*, June 2015, to appear, extended version at, https://sites.google.com/site/jayantapteshomepage/.

[10] J. Apte, “chm-An Implementation of Convex Hull Method,” 2014, http://www.ece.drexel.edu/walsh/aspitrg/software.html.

[11] W. Stein and D. Joyner, “Sage: System for algebra and geometry experimentation,” *ACM SIGSAM Bulletin*, vol. 39, pp. 61 – 64, 2005.

[12] G. Ziegler, *Lectures on Polytopes*. Springer-Verlag, 1995. [Online]. Available: http://books.google.com/books?id=9We8QgAACAAJ

[13] R. Rockefellar, *Convex Analysis*. Princeton Univ. Press, 1970.

[14] K. Fukuda and A. Prodon, “Double description method revisited,” *Combinatorics and computer science*, pp. 91–111, 1996.

[15] D. Bremner, M. D. Sikirić, and A. Schürmann, “Polyhedral representation conversion up to symmetries,” pp. 45–72, 2009.

[16] J. Apte and J. M. Walsh. Exploiting Symmetry in Computing Polyhedral Bounds on Network Coding Rate Regions. extended version at, https://sites.google.com/site/jayantapteshomepage/.

[17] K. Fukuda, “Frequently asked questions in polyhedral computation,” 2004, ftp://ftp.ifor.math.ethz.ch/pub/fukuda/reports/polyfaq040618.pdf.

[18] R. Bödi and K. Herr, “Symmetries in linear and integer programs,” 2009. [Online]. Available: http://arxiv.org/abs/1407.7405