Consider a linear program $LP(A, b, c)$ in the following form:

$$\max\ c^t x \tag{1}$$
$$\text{s.t. } Ax \leq b \tag{2}$$
$$x \geq 0, \tag{3}$$

with x and c are in $\mathbb{R}^n$, b is in $\mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$.

Let $\mathcal{P} = \{P_1,\ P_2,\ \ldots,\ P_k\}$ be partition of the decision variables. We define the restricted linear program, $RLP(A, B, c, \mathcal{P})$ to be:

$$\max\ c^t x \tag{4}$$
$$\text{s.t. } Ax \leq b \tag{5}$$
$$x_i = x_i \ \ \forall\ i, j \ \in P_l,\ l = 1,\ \ldots,\ k \tag{6}$$
$$x \geq 0, \tag{7}$$

Obviously can be written in such a way as to aggregate variables that belong in the same partition. Such an aggregation will yield a formulation with only k variables and (potentially) fewer constraints. We will refer to the aggregated version as $ARLP(A, b, c, \mathcal{P})$.

It is easier, however, to discuss the proposed methods using the above formulation as both, as stated, $LP(A, b, c)$ and $RLP(A, B, c, \mathcal{P})$ exist in the same dimension.

For this work we are primarily focused on partitions $\mathcal{P}$ such that the optimal solution to $LP(A, b, c)$ is equal to that of $RLP(A, B, c, \mathcal{P})$. We note that techniques have been developed that attempt find such $\mathcal{P}$, most notably those that seek to exploit symmetry and its generalization, equitable partitions. In such cases, the benefit of $RLP(A, B, c, \mathcal{P})$ is clear, preprocessing will result in a smaller formulation that will likely be easier to solve. However, the solutions returned by solving the restricted problem will necessarily be in the interior of the optimal face (unless $\mathcal{P}$ consists only of singletons). In many cases it is desirable to find an optimal vertex to $LP(A, b, c)$. Various crossover methods have been developed to map an interior solution to an optimal vertex, but such methods tend to be very computationally taxing. We show here that mapping a solution from restricted problems generated by symmetry and/or equitable partitions to optimal vertices are considerably easier than the more general crossover methods.

1 Background: Symmetry and Equitable Partitions

Symmetries and Fractional Symmetries:

The symmetry group of $LP(A, b, c)$ is defined to be permutations that map feasible solutions to the LP to feasible solutions with the same objective value. Let $\mathcal{G}$ represent the symmetry group. A problem's symmetry group can be used to define equivalence classes on both the sets of variables as well as the set of feasible solutions. As permutations act on vectors, it is most natural to think of how symmetries act on solution vectors. We say two solutions are equivalent if they have the same objective values and there exists a permutation in $\mathcal{G}$ that maps one solution to the other. Computing $\mathcal{G}$ is not practical, as it requires the knowledge of all feasible solutions. In practice, the *formulation group* $\mathcal{F}$, is used to approximate $\mathcal{G}$. Note that $\mathcal{F} \subset \mathcal{G}$. The formulation group of (1) contains the set of permutations $(P_\pi,\ P_\sigma)$ such that:

$$\begin{aligned} cP_\pi &= c, \\ P_\sigma b &= b, \\ AP_\pi &= P_\sigma A, \\ P_\pi \in S^n,\ P_\sigma &\in S^m, \end{aligned} \tag{8}$$

where S_π and S_σ are the sets of all permutation matrices of appropriate size. This ensures that integer solutions are mapped to other integer solutions, something that is important in integer programming but not in linear programming. Dropping the integrality restriction gives the set of *fractional symmetries*, $\mathcal{F}$, that contain the set of doubly stochastic matrices that satisfy:

$$\begin{aligned} cP_\pi &= c, \\ P_\sigma b &= b, \\ AP_\pi &= P_\sigma A, \\ P_\pi \in D^n,\ P_\sigma &\in D^m, \end{aligned} \tag{9}$$

where D_π and D_σ are the sets of all doubly stochastic matrices of appropriate size. Note that the set of all fractional symmetries can be thought of as a polyhedron (the no longer form a group).

Orbits and Generalizations:

We call the set of all solutions equivalent to a solution x an *orbit* of x with respect to $\mathcal{G}$. We use $\text{Orb}(x,\ \mathcal{G})$ to represent the orbit of x with respect to $\mathcal{G}$. We overload the Orb notation to act on variables as well as vectors. The orbit of variable x_i with respect to $\mathcal{G}$ are those variables that are equivalent to x_i. We have that $j \in \text{Orb}(i, \mathcal{G})$ if and only if $e_j \in \text{Orb}(e_i, \mathcal{G})$. The variable orbits can be used to define a natural partition of the variables as $i,\ j \in \text{Orb}(h, \mathcal{G})$ implies that $h, j \in \text{Orb}(i, \mathcal{G})$. We let $\mathcal{O} = \{O_1, \ldots,\ O_k\}$ represent the *orbital partition* of the variables.

While not immediately obvious, equitable partitions are the fractional symmetry analog of orbits. Formally, an equitable partition $(\mathcal{V}, \mathcal{C})$ with respect to $\mathcal{F}$, which we denote as $EQ(\mathcal{P} = (\mathcal{V}, \mathcal{C}), \mathcal{F})$, is a partition of the variables $(\mathcal{V} = (V_1,\ \ldots,\ V_k))$ and constraints $(\mathcal{C} = (C_1,\ \ldots,\ C_k))$ that satisfies the following constraints:

$$\sum_{p \in C} A_{i,p} - A_{j,p} = 0\ \forall i, j \in V \in \mathcal{V},\ C \in \mathcal{C} \tag{10}$$

$$\sum_{i \in V} A_{i,p} - A_{i,q} = 0\ \forall p, q \in C \in \mathcal{C}, V \in \mathcal{V} \tag{11}$$

We say that partition $\mathcal{P}^2$ is a *refinement* of partition $\mathcal{P}^1$ if for all $P_i^2 \in \mathcal{P}^2$ there exists a $P_j^1 \in \mathcal{P}^1$ with $P_i^2 \subseteq P_j^1$. It can be shown that any orbital partition is a refinement of an equitable partition. Conversely, P^1 is *coarser* than P_2. The coarsest equitable partition is easily computed and is often used as the first step in computing the formulation group of an instance.

In both orbits and equitable partitions, we choose one element from each partition and refer to it as the *representative*. For the sake of this paper, we will say that the representative of each partition is the variable/constraint with the smallest index.

Stabilizers and Generalizations:

The *stabilizer* of a group $\mathcal{G}$ with respect to an element v, $stab(v, \mathcal{G})$ is defined to be:

$$stab(v, \mathcal{G}) := \{\pi \in \mathcal{G} \mid \pi(v) = v\}.$$

Note that this definition allows for v to be either a vector or a variable. Using the constraints (8), this can be thought of adding the constraint $P_\pi v = v$, or, in the case where v represents a variable, say x_i, by fixing $P_\pi(i,i)$ to one.

Similar to the stabilizer, we wish to *isolate* elements of an equitable partition by creating a refinement where that element is in a singleton partition. Formally, we have $iso(v, \mathcal{F})$ is define to be:

$$iso(v, \mathcal{F}) := \{\pi \in \mathcal{F} \mid \pi(v) = v\}.$$

Again, this can be thought of adding the constraint $P_\pi v = v$, or, in the case where v represents a variable, say x_i, by fixing $P_\pi(i,i)$ to one.

2 Stuff

A result from Grohe...

Theorem 1. *Let $\mathcal{F}$ be the set of fractional symmetries acting on a linear program and let $\mathcal{P} = (\mathcal{V}, \mathcal{C})$ be the orbital partition with respect to $\mathcal{F}$. The optimal solution value of $LP(A, b, c)$ is equal to the optimal solution value of $RLP(A, b, c, \mathcal{P})$*

Corollary 1. *As a result of these fixings, an aggregated LP with the same optimal objective value can be written using $|\mathcal{V}|$ many variables and $|\mathcal{C}|$ many constraints.*

In general, if $\mathcal{P}^2$ is a refinement of $\mathcal{P}^1$, then the LP solution to $RLP(A, B, c, \mathcal{P}_1)$ is no smaller than $RLP(A, B, c, \mathcal{P}_2)$. Thus, for any partition that is a refinement of the problem's equitable partition $\mathcal{P}$, Theorem 1 ensures that $RLP(A, b, c, \mathcal{P}_1)$ has the same objective value of $LP(A, b, c)$.

Dimensions of RLP polyhedra:

Note that the polyhedra defined by the feasible region of both $LP(A, b, c)$ and $RLP(A, b, c, \mathcal{P})$ are both in $\mathbb{R}^n$. However, while the feasible region of $LP(A, b, c)$

can be fully dimensional, that of $RLP(A, b, c, \mathcal{P})$ is not (except for trivial partitions). Indeed, it is easy to see that the set of equalities defined in (6) have a rank of $n - |\mathcal{P}|$.

Hierarchy of optimal restrictions:

We new define a hierarchy of restrictions that are all guaranteed to share the same optimal objective value (that of the original LP). We define iso^0 to just be $\mathcal{F}$ and recursively define $iso^k(\mathcal{F})$ to be $iso(k, iso^{k-1}(\mathcal{F}))$. As every iso^k is a set of fractional symmetries, it can be used to generate the equitable partition $\mathcal{P}^k$ (with respect to iso^k). As iso^{k+1} is a subset of iso^k, we have that $\mathcal{P}^{k+1}$ is a refinement of $\mathcal{P}^k$. Thus, $\{dim(RLP(A, b, c, \mathcal{P}^k)\}_k$ is a nondecreasing sequence and that $\mathcal{P}^n$ consists of only singletons.

Crossover

We use the above facts to design a crossover strategy that will take a vertex solution from $ARLP(A, b, c, \mathcal{P}^k\}$ and lift it to a vertex solution to $ARLP(A, b, c, \mathcal{P}^{k+1})$. By iterating through, this will eventually lead to a vertex to $ARLP(A, b, c, \mathcal{P}^n) = LP(A, b, c)$

This will be done iteratively by lifting each vertex solution of $RLP(A, b, c, \mathcal{O}^i\}$ to a vertex solution to $RLP(A, b, c, \mathcal{O}^{i+1})$

3 Required Theory

Things to show(?)

- Active constraints (bounds) using $\mathcal{O}^k$ are active in $\mathcal{O}^{k+1}$: this isn't super obvious in the aggregated model, but is in the restricted version(?)
- When one constraint becomes active, so are all others in its respective orbit, defined by *some* stabilizer. We have to be a bit careful in showing this. Crossing over from $\mathcal{O}^k$ to $\mathcal{O}^{k+1}$, the stabilizer we care about is $stab^{k+1}$

Things to consider in the algorithm:

- Three types of pivots, degenerate, those that hit a singleton constraint orbit, and those that hit a singleton constraint, and those that hit a nontrivial constraint orbit.
  - Degenerate: We can drop the corresponding equality constraint and move on.
  - Singleton: We add one linearly independent active constraint
  - Non trivial: we add possibly more than one linearly independent active constraint. To determine this, we only need to compute a maximal set of linearly independent constraints for that orbit (proof). Can we determine this using the symmetry group instead?

Not entirely sure why I put all of this here... Pick and choose...

Orbits of Vertices

Similar to variables, permutations can act on vectors. We can use symmetry to compute how many equivalent vertices there are in a polyhedron. First, recall the Orbit-Stabilizer Theorem (OST):

Theorem 2. *For $x \in \mathbb{R}^n$ and group $\mathcal{G}$, we have $|Orb(x, \mathcal{G})| = \frac{|\mathcal{G}|}{|Stab(x,\mathcal{G})|}$.*

The intuition behind the OST is as follows. Some permutations in $\mathcal{G}$ map x to itself. These permutations are those that make up $\text{Stab}(x, \mathcal{G})$. By somehow dividing $\mathcal{G}$ by these permutations we will be left with permutations that map x to a unique vector.

Using OST, then, for a vertex x^i, the set of equivalent vertices contains $\frac{|\mathcal{G}|}{|\text{Stab}(x^i,\mathcal{G})|}$. Let B^i and NB^i be the set of basic and nonbasic variables associated with vertex x^i. Because NB^i is enough to uniquely determine x^i, we have that $\text{Stab}(x^i, \mathcal{G}) = \text{Stab}(NB^i, \mathcal{G})$. Also, as $\text{Stab}(S, \mathcal{G}) = \text{Stab}(\overline{S}, \mathcal{G})$ for any set S, we also have that $\text{Stab}(x^i, \mathcal{G}) = \text{Stab}(B^i, \mathcal{G})$. Thus, we can compute the size of the orbit of a given vertex by just using the basis information (the actual solution doesn't have to be known).

In addition the the number of equivalent vertices, symmetry information can be used to identify relationships between active and inactive constraints. Consider the following theorem:

Theorem 3. *For vertex x^i, any constraint orbit generated with respect to $Stab(NB^i, \mathcal{G})$ contains either all binding constraints or all inactive constraints.*

Note that the above theorem relies on the stabilizer of x^i, not the whole group. Indeed, the size of the orbits with respect to the stabilizer will be no larger than those with respect to $\mathcal{G}$. The solution to the aggregated solution can be used to give insight on the set of active constraints for the corresponding vertices.

Theorem 4. *For a vertex x^i, let $\overline{x}$ be the average of all vertices in $Orb(x^i, \mathcal{G})$. Then, for any constraint orbit O^C, either all constraints in that orbit are binding or none of them are.*

Corollary 2. *For a vertex x^i, let $\overline{x}$ be the average of all vertices in $Orb(x^i, \mathcal{G})$. If a constraint is binding for $\overline{x}$, than the constraint is binding for all vertices in $Orb(x^i, \mathcal{G})$.*

Data: Aggregated solution x^0, set of active constraints C, group $\mathcal{G}$
Result: A vertex solution to the original LP
while $rank(C) < n$ **do**
    Let $\mathcal{G} = \text{Stab}(C, \mathcal{G})$, generate orbital partition of variables
    $\mathcal{O} = \{O^1, \ldots, O^k\}$;
    Create LP of type (**??**) using $\mathcal{O}$;
    while $rank(C) < n$ **do**
        Select orbit $O^i = \{x_{i_1}, \ldots, x_{i_p}\}$;
        Compute $\text{Stab}(x_{i_1}, \mathcal{G})$
    end
end

Algorithm 1: How to write algorithms