Fences Are Futile: On Relaxations for the Linear Ordering Problem

Alantha Newman[1] and Santosh Vempala[2]

[1] Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139
alantha@theory.lcs.mit.edu

[2] Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA 02139
vempala@math.mit.edu

Abstract. We study polyhedral relaxations for the linear ordering problem. The integrality gap for the standard linear programming relaxation is 2. Our main result is that the integrality gap remains 2 even when the standard relaxations are augmented with *k-fence* constraints for any k, and with *k-Möbius ladder* constraints for k up to 7; when augmented with k-Möbius ladder constraints for general k, the gap is at least $\frac{33}{17} \approx 1.94$. Our proof is non-constructive–we obtain an extremal example via the probabilistic method. Finally, we show that no relaxation that is solvable in polynomial time can have an integrality gap less than $\frac{66}{65}$ unless P=NP.

1 Introduction

Given a complete weighted directed graph, the *linear ordering* problem is to find a linear ordering of the vertices that maximizes the weight of the forward edges (edge (i, j) is a *forward* edge if i precedes j in the ordering). This problem is equivalent to finding a maximum acyclic subgraph of a given graph.

The linear ordering problem is NP-hard [8], motivating the question of polynomial time approximation algorithms. It is in fact easy to find a solution with weight at least half the optimum: take *any* linear ordering of the vertices; partition the edges into two sets, those going forward in the ordering and those going backward. Both sets are acyclic; one of these sets has weight at least half the total weight of all the edges in the graph (and hence at least half the optimum). This simple algorithm gives the best-known polynomial-time computable approximation factor for the problem (namely $\frac{1}{2}$).

In this paper, we study the quality of polyhedral relaxations for this optimization problem [4, 5]. The quality of a relaxation can be measured by the *integrality gap*, the maximum possible ratio between the linear programming optimum and the true integral optimum. A well-known linear programming relaxation for the problem is based on the simple idea of requiring that from every directed cycle C, a solution contains at most $|C|-1$ edges. The corresponding linear constraints are exponential in number (one for each cycle), but can be solved in polynomial time via an efficient separation oracle. This and another well-known relaxation are described in Section 2. How good are the relaxations? The integrality gap for both of these standard relaxations turns out to be at least $2-\epsilon$ for any $\epsilon > 0$. Thus, the estimate they provide on the optimum is no better (in the worst-case) than the trivial upper bound of the total edge weight.

A natural next step is to strengthen these standard relaxations by adding constraints. To this end, a promising set of constraints are the *k-fence* constraints [4, 5]. Although these constraints are NP-complete to separate in general [9], they can be separated in polynomial time for any fixed k. Another set of constraints that have been proposed are the *k-Möbius* ladder constraints [4, 5]. These are known to be separable in polynomial time [1, 12]. In Section 3, we present our main result: the integrality gap is 2 even with k-fence constraints for any k and with k-Möbius ladder constraints for $k \leq 7$; when augmented with k-Möbius ladder constraints for arbitrary k, the gap is at least $\frac{33}{17} \approx 1.94$. Our proofs of the integrality gap start with a probabilistic construction which is molded to have the desired structure (thus we demonstrate the existence of extremal graphs without explicitly describing them).

Finally, we establish a concrete lower bound on approximability: it is NP-hard to approximate the optimum to within a factor better than $\frac{65}{66}$, i.e. no polynomial-time solvable relaxation can have an integrality gap less than $\frac{66}{65}$. The reduction, described in Section 4, is from the problem of finding a maximum satisfiable subset of a given set of linear equations modulo 2.

2 Standard LP Relaxations

In this section, we describe two standard linear programming relaxations, prove that they have the same optimal value for any graph, and show that both relaxations can be arbitrarily close to twice the value of the optimum in the worst case.

2.1 $\mathbf{LP_1}$

The maximum acyclic subgraph problem can be viewed as maximizing the number of edges subject to a constraint for every cycle. Grötschel, Jünger, and Reinelt refer to these constraints as *dicycle inequalities* [4]. We will call them *cycle constraints.* The constraints specify that the sum of the edge variables on any cycle C is at most $|C| - 1$.

$$\begin{array}{ll} \text{maximize} & \sum_{(i,j)\in E} \; w_{ij}x_{ij} \\ \text{subject to:} & \\ & \sum_{ij\in C} \; x_{ij} \leq |C| - 1 \quad \forall C \\ & x_{ij} \in \{0,1\} \quad \forall ij \in E \end{array}$$

The solutions to this integer program are acyclic subgraphs. It is NP-hard to solve this integer program. However, we can relax the requirement that the x_{ij}

are in $\{0, 1\}$ and replace it with the requirement that $0 \leq x_{ij} \leq 1$. We refer to this linear programming relaxation as LP_1. We can solve LP_1 in polynomial time using the Ellipsoid Algorithm [6] via the following polynomial-time separation oracle. Given an assignment for the variables x_{ij}, we consider the graph with each edge (i, j) assigned a weight of $1 - x_{ij}$. In this graph, we find the minimum weight cycle. If there is any cycle with weight less than 1, then the corresponding cycle of length C actually has weight more than $C - 1$, which highlights a violated constraint.

2.2 $\mathbf{LP_2}$

Another integer program is based on the linear ordering problem. It has a variable for every pair of vertices $i, j \in V$. In this program, there are only constraints for 2- and 3-cycles. This set of constraints is discussed by Grötschel, Jünger, and Reinelt [5].

maximize $\sum_{ij} w_{ij} x_{ij}$
subject to:

$$x_{ij} + x_{ji} = 1 \quad \forall i, j \in V$$
$$x_{ij} + x_{jk} + x_{ki} \leq 2 \quad \forall i, j, k \in V$$
$$x_{ij} \in \{0, 1\} \quad \forall i, j \in V$$

Solutions for this integer program correspond to linear orderings. Again, it is NP-hard to solve this integer program. We refer to the corresponding relaxation as LP_2. Although LP_2 only contains constraints for 2- and 3-cycles, we can show that a valid solution for LP_2 does not violate *any* cycle constraints.

Lemma 1. *A solution for LP_2 does not violate any cycle constraints.*

Proof. We will prove by induction on k that a solution for LP_2 does not violate any k-cycle constraints. Clearly, a valid solution for LP_2 does not violate any 2- or 3-cycle constraints. Assume all k-cycle constraints are satisfied. Then we will show that all $(k+1)$-cycle constraints are satisfied. Consider a cycle C of length $k+1$. Choose two non-adjacent vertices i, j in C. Now consider the following two edge-disjoint cycles: C_1 is composed of edge (i, j) and the path from j to i in C and C_2 is composed of edge (j, i) and the path from i to j in C. By induction, we have $\sum_{e \in C_1} x_e + \sum_{e \in C_2} x_e \leq |C_1| - 1 + |C_2| - 1 \leq |C|$ and $x_{ij} + x_{ji} = 1$. Thus, $\sum_{e \in C} x_e = \sum_{e \in C_1} x_e + \sum_{e \in C_2} x_e - x_{ij} - x_{ji} \leq |C| - 1$. □

A maximum acyclic subgraph has the same weight as a maximum linear ordering, i.e. the optimal integral solutions for the two integer programs above are equal. We now prove that the optimal solutions for the two linear programming relaxations are equal. For some graph $G = (V, E)$ with edge weights $w = \{w_{ij}\}$, let $OPT(LP_1)$ denote an optimal solution for LP_1 and $|OPT(LP_1)|$ denote its objective value. Define $OPT(LP_2)$ similarly with $w_{ij} = 0$ for all $(i, j) \notin E$.

Theorem 1. $|OPT(LP_1)| = |OPT(LP_2)|$.

Proof. First, we will show that $|OPT(LP_1)| \geq |OPT(LP_2)|$, i.e. given an optimal solution for LP_2, we can find a solution for LP_1 with the same value. We simply let the solution for LP_1 be the subset of $\{x_{ij}\}$ such that $(i,j) \in E$. By Lemma 1, this solution does not violate any cycle constraints and is therefore a valid solution for LP_1.

Second, we will show that $|OPT(LP_2)| \geq |OPT(LP_1)|$, i.e. given an optimal solution for LP_1, we can construct a solution for LP_2 with the same objective value. Assign all edges in E value x_{ij} where x_{ij} is taken from the given solution for LP_1. Since this is a valid solution for LP_1, no cycle constraints have been violated thus far. Now consider an arbitrary order for the pairs (j,i) such that $(i,j) \in E$ and $(j,i) \notin E$ and assign $x_{ji} = 1 - x_{ij}$ in that order. Let (j,i) be the first edge causing a violated cycle constraint. Then there is some path p_{ij} of length ℓ from i to j such that the total value of the edges in p_{ij} is more than $\ell - x_{ji}$. Since $(i,j) \in E$, and the solution for the edges in E is optimal, it must be the case that there is some path p_{ji} of length ℓ' such that the value of the edges in p_{ji} equals to $\ell' - x_{ij}$, i.e. a cycle constraint for some cycle containing edge (i,j) must be tight, otherwise we could increase the value of x_{ij}. Thus, p_{ij} and p_{ji} form a cycle of length $\ell + \ell'$ of value more than $\ell + \ell' - (x_{ij} + x_{ji}) = \ell + \ell' - 1$. So it is a contradiction that this edge is the first to cause a violated cycle constraint. Also note that all 2-cycles in E have total value exactly 1. Otherwise, for some $x_{ij} + x_{ji} < 1$, we can find a cycle composed of the paths p_{ji} and p_{ij} which violates a cycle constraint. Simply let p_{ji} be the path in the cycle with x_{ij} for which a cycle constraint is tight, and define p_{ij} similarly.

Let $\bar{G} = (V, \bar{E})$ be the graph with edge set $\bar{E} = \{(i,j)\}$ such that $(i,j) \in E$ or $(j,i) \in E$. By the argument above, all 2-cycles in $\bar{G}$ have value exactly 1 and no cycle constraints are violated. Now we will assign values x_{ij} to all edges (i,j) such that $i,j \in V$ and neither (i,j) nor (j,i) are in E. We define the *shortest path* between i and j to be the path with the least total value, where x_{ij} is the value of an edge. Let α_{ij} be the length of the shortest directed path from i to j in $\bar{G}$. Define α_{ji} similarly. For any i,j the shortest paths from i to j and from j to i form a cycle in the current graph. Therefore, $\alpha_{ij} + \alpha_{ji} \geq 1$. Without loss of generality, assume $\alpha_{ij} \leq \alpha_{ji}$. Then let $x_{ij} = \min\{\frac{1}{2}, \alpha_{ij}\}$ and $x_{ji} = 1 - x_{ij}$. Thus, every cycle that includes edge (i,j) or edge (j,i) will have value at least 1, which implies that every cycle C will have value at most $|C| - 1$. This implies that all 3-cycle constraints are satisfied and all 2-cycles have value exactly 1. □

2.3 Integrality Gap

The *integrality gap* of a linear program is the worst case ratio between the value of an optimal fractional solution and the value of an optimal integral solution over all weight functions $w = \{w_{ij}\}$. Formally, the integrality gap is defined as,

$$\max_{w>0} \frac{|OPT(LP)|}{|OPT(IP)|}$$

In this section, we show that the integrality gap for both LP_1 and LP_2 is $2 - \epsilon$ for any $\epsilon > 0$. As the basis of the construction, we use the fact that there exists a class of undirected graphs with girth g and $\Theta(n^{1+1/g})$ edges. This result is due to Erdös and Sachs [2]. Graphs from this class have been used to prove integrality gaps for the maximum cut problem [11]. Based on these graphs, we define $G(n)$ to be a family of graphs with the following properties. A graph $G \in G(n)$ has n vertices, girth $g = \Theta(\frac{\log n}{\log\log n})$, and $n^{1+1/g}$ edges. Then we have the following lemma.

Lemma 2. *For any $\epsilon > 0$, there exists an $n \geq f(\epsilon)$ such that at least one directed orientation of $G = (V, E) \in G(n)$ has the following property: for any ordering of the vertices, the number of forward edges is at most $(1+\epsilon)|E|/2$.*

A proof of Lemma 2 can be found in [10]. We now define $G(\epsilon)$ to be the family of *directed* graphs on $n \geq f(\epsilon)$ vertices whose underlying undirected graphs belong to $G(n)$ and which have maximum acyclic subgraphs of size at most $(1+\epsilon)|E|/2$. All edges in $G \in G(\epsilon)$ have weight $w_{ij} = 1$. We use $G(\epsilon)$ to prove the following theorem.

Theorem 2. *The integrality gap of LP_1 is at least $2 - \epsilon$ for any $\epsilon > 0$.*

Proof. For a graph $G = (V, E) \in G(\epsilon)$, we assign $x_{ij} = 1 - 1/g$ for every edge in G, where g is the girth of G. This is a feasible solution for LP_1 since there are no cycles of length less than g. Thus, the optimal solution of LP_1 has size at least $|E|(1 - 1/g)$. The ratio of the optimal fractional solution to the optimal integral solution is at least $(1 - 1/g)/(\frac{1}{2}(1+\epsilon))$. Since $g = \Theta(\frac{\log\log n}{\log n})$, then for any $\epsilon' > 0$, we can choose ϵ and n so that $2(1-1/g)/(1+\epsilon) \geq 2 - \epsilon'$. □

Theorem 3 follows from Theorem 2 and Theorem 1.

Theorem 3. *The integrality gap of LP_2 is at least $2 - \epsilon$ for any $\epsilon > 0$.*

3 Augmented LP Relaxations

In the previous section, we saw that a rather non-trivial LP has an integrality gap arbitrarily close to 2, thus providing an upper bound that is no better than the total weight of the edges in the worst case. How can we get a better upper bound? One way would be to add new constraints to this LP. Some well-known constraints for this problem are the so-called *fence* constraints and *Möbius ladder* constraints presented by Grötschel, Jünger, and Reinelt [4, 5]. In this section, we will show that that if we augment LP_2 with k-fence constraints for any k and with k-Möbius ladder constraints for $k = 3$, then the integrality gap remains 2. This is also true for $5, 7$-ladders, but the proofs are omitted here. For $k \geq 9$, the integrality gap of LP_2 augmented with k-Möbius ladder constraints is at least $\frac{33}{17}$. (Note that the integrality gap of LP_1 augmented with these constraints trivially remains 2: a graph belonging to $G(\epsilon)$ has girth greater than 4 for sufficiently small ϵ and therefore does not contain any k-fences or k-Möbius ladders.)

Throughout this section, it will be convenient to have the following definitions. An edge (i, j) is the *complementary* edge to edge (j, i). The *value* of a set S of edges is defined to be $x(S) = \sum_{(i,j)\in S} x_{ij}$. We define the *shortest* path from i to j in a graph G as a path from i to j with the least total value.

3.1 The Bad Example Graph

We will now describe the *bad example* graph–the graph which we use to prove our lower bound on the integrality gap of the augmented LP relaxation. We use the family of graphs $G(\epsilon)$ defined in Section 2.3. We begin with a graph $G = (V, E) \in G(\epsilon)$. (We used this graph to prove Theorem 2. However, now we need to assign a value to every edge in the complete graph.) For every $(i, j) \notin E$, assign $w_{ij} = 0$. For every edge $(i, j) \in E$, we assign x_{ij} value $1 - 1/g$ and x_{ji} value $1/g$, where g is the girth of G. For all i, j such that neither (i, j) nor (j, i) are in E, we assign a value to x_{ij} using the rule given in the proof of Theorem 1. We will restate this rule here for the sake of convenience. Define $\bar{G}$ to be the graph consisting of the edges in E and their complementary edges. Let α_{ij} be the shortest path from i to j in $\bar{G}$. Define α_{ji} similarly. Without loss of generality, assume $\alpha_{ij} \leq \alpha_{ji}$. Then assign x_{ij} and x_{ji} using the following rule.

Edge Assignment Rule: $x_{ij} = \min\{\frac{1}{2}, \alpha_{ij}\}, \quad x_{ji} = 1 - x_{ij}$

The following corollary holds for the complete directed graph $\tilde{G}$ in which every edge has been assigned a value.

Corollary 1. *If the shortest path from i to j in $\bar{G}$ is $\alpha < \frac{1}{2}$, then the value of the shortest path between i and j in $\tilde{G}$ is α. If the value of the shortest path from i to j in $\bar{G}$ is at least $\frac{1}{2}$, then the value of the shortest path from i to j in $\tilde{G}$ is at least $\frac{1}{2}$.*

Recall that the optimal objective value of LP_2 for $\tilde{G}$ is at least $|E|(1 - 1/g)$. Therefore, if we can show that the edges of $\tilde{G}$ also satisfy other specified constraints, then we can show that the integrality gap of LP_2 augmented with these constraints remains the same as LP_2.

3.2 Möbius Ladders

A Möbius ladder for an odd integer k (a k-ladder) is defined to be a set of $2k$ vertices $\{a_1, b_1, \cdots, a_k, b_k\}$ and $3k$ edges such that each vertex a_i has a directed edge to b_{i+1} and $b_{i-1}, 1 \leq i \leq k$. (We define $b_0 = b_k$ and $b_{k+1} = b_1$.) There is also an edge from b_i to a_i. A 5-ladder is shown in Figure 1. (A 3-ladder is isomorphic to a 3-fence, which will be defined later on.)

An acyclic subgraph of a 5-Möbius ladder includes at most 12 of the 15 edges. However, there is a fractional solution of $12\frac{1}{2}$ that satisfies LP_2: each edge from b_i to a_i is assigned a value of $\frac{1}{2}$ and all other edges are assigned value 1. In general, an acyclic subgraph of a k-ladder includes at most $3k - (\frac{k+1}{2})$ of the edges. However, we can always find a fractional solution with value $2\frac{1}{2}k$ that satisfies every cycle constraint. So we add the following constraint to LP_2 for every subset of edges that forms a k-ladder.

$$\sum_{(i,j) \in k-ladder} x_{ij} \leq 3k - (\frac{k+1}{2}) \tag{1}$$

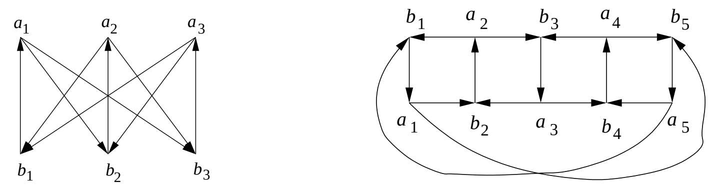

Fig. 1. A 3-ladder (or 3-fence) and a 5-ladder.

Recall that LP_2 yields an assignment for the complete graph. Therefore, in order to show that the total value of the edges in any k-ladder is at most $3k - \frac{k+1}{2}$, it suffices to show that the total value of the edges in any k-ladder is at least $\frac{k+1}{2}$. This is because the set of edges complementary to a k-ladder also form a k-ladder and the sum of the values of both k-ladders is exactly $3k$.

Now we will show that constraint (1) is satisfied for all 3-ladders in $\tilde{G}$, which we defined in Section 3.1. Let M be the set of $3k$ edges in a k-ladder and let C be the subset of edges (a_i, b_{i+1}) and (a_i, b_{i-1}) for $i \in \{1, \ldots k\}$. The edges in C make up an undirected cycle of value $2k$. For example, in the 5-ladder shown in Figure 1, C is the set of all edges expect for the 5 vertical edges. Furthermore, define T to be the set of edges in $\bar{G}$ (each graph $\tilde{G}$ has a corresponding $\bar{G}$, which is also defined in Section 3.1) that belong to the shortest paths for each pair of vertices i, j such that edge $(i, j) \in C$. If there are multiple shortest paths for some pair, only one of these paths is included in T. The edges in T will play a key role in our proof.

We say that T contains an *undirected* cycle if the underlying undirected graph contains a cycle. There are two cases to consider: T can either be a tree or contain an undirected or directed cycle. In the case that T contains a cycle, we will show that constraint (1) is satisfied, and when T is a tree, we will show that constraint (1) is satisfied for $k = 3$. Constraint (1) is also satisfied for $k = 5, 7$, but the proofs are omitted. However, this constraint is not necessarily satisfied when $k = 9$. In other words, our edge assignment rules could violate some k-ladder constraints for $k \geq 9$.

Case 1: *T* Contains a Cycle. We consider two subcases based on the values of the $2k$ paths in T from a_i to b_{i+1} and from a_i to b_{i-1} for $i \in \{1, \ldots k\}$. For each of these cases, we will show that the total value of the edges in M is at least $\frac{k+1}{2}$. In the proof, we will use the following lemma.

Lemma 3. *If the value of C is at least 1, then the total value of the edges in M is at least $\frac{k+1}{2}$.*

Proof. If the total value of C is $1 + \epsilon$ for some $\epsilon \geq 0$, then we will show that the total value of the other k edges in M is at least $\frac{k-1-\epsilon}{2}$. This would imply that the total value of the edges in M is $(1+\epsilon)+(\frac{k-1-\epsilon}{2})=\frac{k+1+\epsilon}{2}$ as required. We just need to show that $x(M\setminus C)$ is at least $\frac{k-1-\epsilon}{2}$. Consider the equations for each of the directed 4-cycles in M. There are k such 4-cycles, which by Lemma 1 each have value at least 1. Note that each edge in C appears in exactly one equation, and each edge of $M\setminus C$ appears in exactly two equations. Adding these k equations, we have: $x(C)+2x(M\setminus C)\geq k$. By assumption, $x(C)=1+\epsilon$. Therefore, we have: $x(M\setminus C)\geq\frac{k-1-\epsilon}{2}$. □

Lemma 4. *If T contains a cycle, then the total value of M is at least $\frac{k+1}{2}$.*

Proof. Note that if two or more of the $2k$ paths in C have value at least $\frac{1}{2}$, then by Corollary 1 and Lemma 3, the total value of M is at least $\frac{k+1}{2}$. Now we will consider two remaining cases.

(i) *All $2k$ paths in T have value strictly less than $\frac{1}{2}$.* By Claim 1, all edges in C have value equal to the value of their respective paths in T. Since some subset of the edges in T form a cycle, the total value of the edges in T is at least 1. Since every edge in T, by definition, belongs to at least one of the $2k$ shortest paths, the total value of C is at least the total value of T, which is at least 1. By Lemma 3, the total value of M is at least 2.

(ii) *Only one of the $2k$ paths in T has value at least $\frac{1}{2}$.* We can assume the total value of the other $2k-1$ edges in C is less than $\frac{1}{2}$. In this case, note that T does not contain a directed cycle. If it did, the total value of the edges in C would be at least 1. Thus, if we removed the edge with value at least $\frac{1}{2}$, the remaining edges would sum to at least $\frac{1}{2}$.

Without loss of generality, assume that edge (a_1,b_2) in C is the edge with value at least $\frac{1}{2}$. Consider the 4-cycle $\{a_1,b_2,a_2,b_1\}$ in M that contains this edge. Since T contains a path from a_2 to b_1 and a path from a_1 to b_2, it cannot contain a path from b_1 to a_1 and from b_2 to a_2, since then it would contain a directed cycle. Without loss of generality, assume T does not contain a path from b_1 to a_1. Let T' be the set of edges in G that correspond to the shortest paths for all edges in C except (a_1,b_2). Then T' contains a directed or undirected path from a_1 to b_1. Since the total value of the edges in T' is less than $\frac{1}{2}$, there is a directed or undirected path from a_1 to b_1 in G with value less than $\frac{1}{2}$. Thus the shortest path in G from b_1 to a_1 must have value at least $\frac{1}{2}$. Consider the $\frac{k-1}{2}$ edge-disjoint 4-cycles in M that remain when we remove edges with either endpoint in the set $\{a_1,b_1\}$. Then, since edges (b_1,a_1) and (a_1,b_2) each have value at least $\frac{1}{2}$ and the 4-cycle has value at least $\frac{k-1}{2}$, then the total value of the edges in M is at least $\frac{k+1}{2}$. □

Corollary 2. *For a 3-fence in which the corresponding T contains a cycle, either $x(C)$ is at least 1, or C contains an edge of value at least $\frac{1}{2}$.*

Case 2: *T* is a Tree. We will consider two subcases based on the total value of the edges in T. For these subcases, we will use the following two lemmas.

Lemma 5. *If T is a tree, then every edge in T is included in the shortest paths corresponding to at least two edges in C.*

Proof. Assume there is an edge e in T that belongs to only one shortest path corresponding to an edge in C. Consider the set of edges that corresponds to the $2k-1$ shortest paths corresponding to the other $2k-1$ edges in C. By assumption, this set of edges does not include edge e. This set forms a connected graph, since the $2k-1$ corresponding edges form a connected graph. It also contains all $2k$ vertices in M. Thus, if we add edge e to this graph, it will contain a cycle, implying that T contains a cycle, which is a contradiction. □

Lemma 6. *If T is a tree corresponding to a 3-ladder, then for some $i \in \{1, 2, 3\}$, T contains a directed path from a_i to b_i and from a_{i+1} to b_{i+1}.*

Proof. First we will show there is a path from a_i to b_i for some i. Let $p(j, k)$ denote the set of edges in the directed path from j to k in T. Since T is a tree, if $p(j, k)$ exists, it is unique. Assume there is not a path from a_i to b_i for any i. We will show that this leads to a contradiction. Consider the paths $p(a_1, b_2)$ and $p(a_1, b_3)$. The first case is that, without loss of generality, b_2 is in $p(a_1, b_3)$. Since there must be a directed path from a_3 to b_2 and there is a directed path from b_2 to b_3, there is a directed path from a_3 to b_3. The second case is that b_2 is not in $p(a_1, b_3)$ and b_3 is not in $p(a_1, b_2)$. Let v be the vertex that belongs to both $p(a_1, b_2)$ and $p(a_1, b_3)$ and is farthest from a_1, as shown in Figure 2.

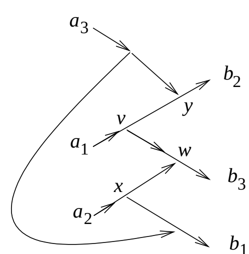

Fig. 2.

There must also be a path from a_2 to b_3 but not from a_2 to b_2. Thus, there must be a path from a_2 to some vertex $w \neq v$ in the path $p(v, b_3)$. Similarly, since there is no path from a_1 to b_1, there must be a path from some vertex $x \neq w$ in $p(a_2, w)$ to b_1. There is also a path from a_3 to a vertex $y \neq v$ in $p(v, b_2)$, since there is a path from a_3 to b_2 but not from a_3 to b_3. Thus, T contains a simple undirected path from a_3 to b_1. But T also contains a directed path from a_3 to b_1. So T contains a cycle, which is a contradiction.

Now, without loss of generality, assume there is path from a_1 to b_1 in T and assume all the vertices on this path are numbered in increasing order. Consider the vertex v_1 and v_2 where paths from a_2 and to b_2, respectively, intersect the path from a_1 to b_1. If v_1 is less than v_2, then $p(a_1, b_1)$ and $p(a_2, b_2)$ intersect and we are done. So assume v_1 is greater than v_2. But then the path from a_3 must intersect $p(a_1, b_1)$ at a point before v_2 and the path to b_3 must intersect $p(a_1, b_1)$ at a point after v_1, so the paths $p(a_1, b_1)$ and $p(a_3, b_3)$ intersect. □

Lemma 7. *The total value of a 3-ladder is at least 2.*

Proof. By Lemma 4, this is true for the case when T contains a cycle. If T is a tree such that the total value of the edges in T is at least $\frac{1}{2}$, then by Lemma 5 every edge in T belongs to the corresponding path for at least two edges in C. Thus, the total value of C is at least 1. By Lemma 3 the total value of M is at least $\frac{k+1}{2}$.

If the total value of the edges in T is less than $\frac{1}{2}$, then by Lemma 6 and without loss of generality, assume T contains a directed path from a_1 to b_1 and from a_2 to b_2. Let s be the value of the directed path from a_1 to b_1 in T, i.e. $s = \sum_{ij \in p(a_1,b_1)} x_{ij}$. Note that $s < \frac{1}{2}$ since the total value of the edges in T is less than $\frac{1}{2}$. The value of $x_{b_1 a_1}$ is at least $1 - s$ and each edge in the path from a_1 to b_1 is on a path from a_1 to b_2 or from a_2 to b_1, so the sum of $x_{a_1 b_2}$ and $x_{a_2 b_1}$ is at least s. Therefore, the total value of the edges $(b_1, a_1), (a_1, b_2)$, and (a_2, b_1) is at least 1. The 4 edges in M that have neither endpoint in the set $\{a_1, b_1\}$ make up a directed 4-cycle, so they have total value at least 1. Thus any 3-ladder has value at least 2.

□

For $k = 5, 7$, it is also the case that the total value of the edges in M is at least $\frac{k+1}{2}$. These proofs are omitted here. For $k = 9$, it is not necessarily the case that the total sum of the edges in a Möbius ladder is at least 5. We conclude this section with a bound for the general case.

Theorem 4. *The integrality gap of LP$_2$ augmented by Möbius ladder constraints is at least $\frac{33}{17} - \epsilon$ for any $\epsilon > 0$.*

Proof. When k is at least 9, then T is a tree with at least 18 unique vertices and at least 17 edges. If every edge original edge in G is assigned value $\frac{33}{34}$ and every complementary edge is assigned value $\frac{1}{34}$, then the total value of the edges in T is at least $\frac{1}{2}$. Thus, the objective value of the LP relaxation augmented with constraint (1) for all k will be at least $33|E|/34$. Since the optimal integral value is arbitrarily close to $|E|/2$ for sufficiently large n, the integrality gap of LP$_2$ extended by Möbius ladder constraints is at least $\frac{33}{17} - \epsilon$ for any $\epsilon > 0$. □

3.3 Fence Constraints

A *k-fence* is obtained by directing the edges of a complete undirected bipartite graph on $2k$ vertices as follows: the $2k$ vertices are divided into two sets, A and B, of k vertices each. Each vertex from set A is paired with a vertex from set B and each of these pairs is connected with a *up* edge. All other edges are directed *down*. A 3-fence is shown in Figure 1.

An acyclic subgraph of a 3-fence includes at most 7 of the 9 edges. However, there is a fractional solution of $7\frac{1}{2}$ that satisfies LP$_2$: each down edge is assigned a value of 1 and each up edge is assigned $\frac{1}{2}$. In general, an acyclic subgraph of a k-fence includes at most $k^2 - k + 1$ of the edges. However, we can always find a fractional solution with value $k^2 - \frac{k}{2}$ that satisfies every cycle constraint. Thus, if we add fence constraints to the LP relaxation, we may get a fractional solution that is a better approximation of the integral solution. Fence constraints state that the total value of the edges in any k-fence cannot exceed $k^2 - k + 1$.

$$\sum_{(i,j) \in k-fence} x_{ij} \leq k^2 - k + 1 \tag{2}$$

We will show that despite the fact that LP_2 is strengthened by adding constraint (2), the integrality gap of this augmented LP relaxation remains 2. To show this, we will again use the graph $\tilde{G}$ discussed in Section 3.1. Since a 3-ladder is also a 3-fence, we know that the total value of any 3-fence is at most 7 in this solution. Thus, we can show that the total value of any k-fence is at most $k^2 - k + 1$.

Lemma 8. *The total value of any k-fence is at most $k^2 - k + 1$.*

Proof. The set of complementary edges of a k-fence also form a k-fence and hence (2) is equivalent to the condition that the total value of edges in a k-fence is *at least* $k - 1$. In Lemma 7, we showed that the lemma is true for $k = 3$, which will be the base case for our inductive proof. We will assume that the total value of any $(k-1)$-fence is at least $k - 2$ and show that the total value of any k-fence is at least $k - 1$.

A k-fence contains $\binom{k}{3}$ distinct 3-ladders (or 3-fences) as subgraphs. For some 3-ladder contained in a k-fence, if the corresponding T is a tree (T and C corresponding to a 3-ladder are defined in Section 3.2) with value less than $\frac{1}{2}$, then by Lemma 6 for some $i \in \{1, 2, 3\}$, there is a directed path from a_i to b_i and from a_{i+1} to b_{i+1}. Thus, the total value of the edges $(b_i, a_i), (a_i, b_{i+1})$, and (a_{i+1}, b_i) is at least 1. When we remove all edges that have one endpoint in $\{a_i, b_i\}$ from the k-fence, we are left with a $(k-1)$-fence. By induction, this fence has value at least $k - 2$. Thus, the total value of the k-fence is at least $k - 1$.

If it is the case that for none of the 3-ladders contained in the k-fence, the corresponding T is a tree with value less than $\frac{1}{2}$, then we will show that the total value of the edges directed from A to B in the k-fence is at least 1. Consider a particular 3-ladder that is a subgraph of the k-fence. If the corresponding T is a tree with value at least $\frac{1}{2}$, then this is true by Lemma 5. If T contains a cycle, then by Corollary 2, the set of edges C corresponding to this 3-ladder has value at least 1 or contains an edge e with at least $\frac{1}{2}$. In the latter case, consider another 3-ladder subgraph that does not contain edge e. Using Corollary 2 again, one of the edges in C corresponding to *this* 3-ladder must also contain an edge with value at least $\frac{1}{2}$. Thus, the total value of the edges directed from A to B is at least 1.

Let Y be the set of edges from b_i to a_i and let X be the set of edges from a_i to b_j. If we consider the k possible $(k-1)$-fences that are subgraphs of the k-fence, we see that each edge in Y is used in $k - 1$ of these $(k-1)$-fences, and each edge in X is used in $k-2$ of these $(k-1)$-fences. By the induction hypothesis, each $(k-1)$-fence has value at least $k-2$ by induction. Combining these k equations, we have: $(k-1)x(Y)+(k-2)x(X) \geq k(k-2)$. We know that $x(X)$ is at least 1. The minimum value of $x(X)+x(Y)$ that satisfies the equation is $x(X)+x(Y)=k-1$, i.e. $x(X)=1, x(Y)=k-2$. Thus, the total value of the edges in a k-fence is at least $k-1$, which implies that it is also at most k^2-k+1. □

Theorem 5. *The integrality gap of LP_2 augmented with fence constraints is $2-\epsilon$ for any $\epsilon > 0$.*

4 Lower Bounds on Approximation

In this section, we describe a reduction from the problem of finding a maximum satisfiable subset of a given set of linear equations modulo 2 with three variables per equation to the maximum acyclic subgraph problem. For this problem, Håstad proved the following tight bound.

Theorem 6 (Håstad [7]). *For every $\epsilon > 0$, it is NP-hard to tell if a given set of linear equations modulo 2 with three variables is satisfiable or at most $m(\frac{1}{2}+\epsilon)$ of its clauses are satisfiable.*

We use Theorem 6 and the reduction described below to obtain the following lower bound on the maximum acyclic subgraph problem (and hence the linear ordering problem).

Theorem 7. *It is NP-hard to approximate the maximum acyclic subgraph to within $\frac{65}{66}+\epsilon$ for any $\epsilon > 0$.*

Given a set of m linear equations on n variables, we construct a graph G using the following rules: (we assume all equations have the right hand side zero by negating one literal if necessary.)

1. For each variable $x \in F$, we create two vertices and two edges. The vertices are x_0 and x_1 and the edges are (x_0, x_1) and (x_1, x_0). These vertices and edges will form the *variable* gadget for x.
2. For each clause $C_j \in F$, we construct a *clause* gadget. Each clause has the form $x+y+z \equiv 0$ where x, y and z are literals. For a literal x in the clause we create a 4-cycle $\{x_2, x_3, x_4, x_5\}$. We label edge (x_5, x_2) as $x=1$ and edge (x_3, x_4) as $x=0$. We also do this for the literals y and z. Then we add the following 12 edges as shown in Figure 3: (z_2, x_5), (z_2, y_3), (z_4, y_3), (z_4, x_3), (x_2, z_3), (x_2, y_5), (x_4, z_5), (x_4, y_5), (y_2, z_3), (y_2, z_5), (y_4, x_3), (y_4, x_5).
3. Each clause gadget is linked to the appropriate variable gadgets as follows.
   - For a literal x, we connect the corresponding 4-cycle in the clause gadget to the variable gadget by adding edges $(x_2, x_1), (x_1, x_3), (x_0, x_5), (x_4, x_0)$.
   - For a literal $\overline{x}$, we connect the corresponding 4-cycle in the clause gadget to the variable gadget by adding edges $(x_2, x_0), (x_0, x_3), (x_1, x_5), (x_4, x_1)$.

The resulting graph G has $36m + 2n$ edges, 36 edges for each clause gadget and two edges for each variable gadget. In order to relate variable assignments to acyclic subgraphs of G, we say that removing edge (x_1, x_0) (labeled $x = 1$ in Figure 3) corresponds to setting the variable x to true, and removing edge (x_0, x_1) (labeled $x = 0$ in Figure 3) corresponds to setting variable x to false. Throughout the proof, we will refer to edges labeled $x = 0$ or $x = 1$ in a clause gadget for a literal x (i.e edges (x_5, x_2) and (x_3, x_4)) and the edges in the variable gadgets as *labeled* edges. The proof of Theorem 7 uses the lemmas below. A feedback arc set of a graph is defined as a set of edges whose removal results in an acyclic graph.

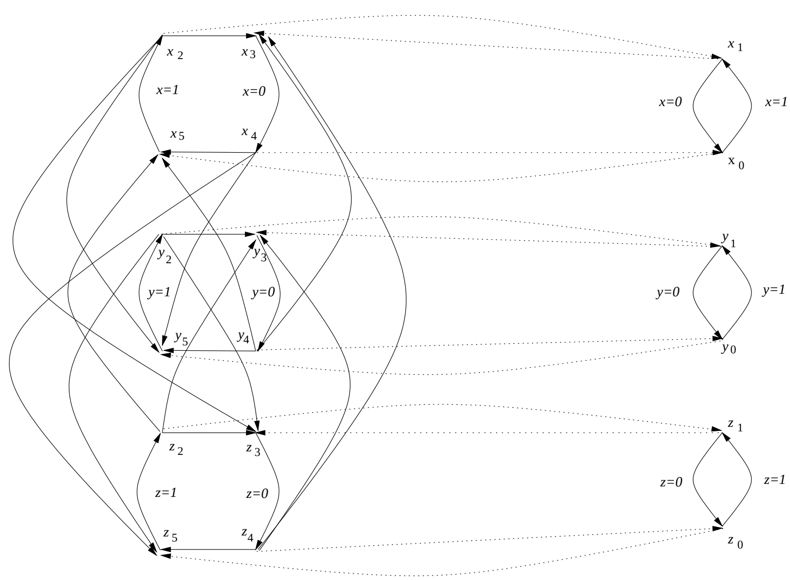

Fig. 3. The clause and variable gadgets for $x + y + z \equiv 0$.

Lemma 9. *A minimal feedback arc set is acyclic.*

Proof. For any acyclic graph, there is an ordering of the vertices such that all edges in the graph are forward edges, i.e. for an edge (i, j), i precedes j in the ordering. Given a feedback arc set, consider such an ordering for the acyclic graph that is obtained by deleting the feedback arc set. If any edges in the feedback arc set are forward edges, then the feedback arc set is not minimal, since such edges can be added to the acyclic graph without creating any cycles. Thus, the feedback arc set must consist only of backward edges and hence is itself acyclic. □

Lemma 10. *There is a minimum feedback arc set of G consisting of only labeled edges.*

Proof. Note that every cycle in G contains labeled edges. This is because for every non-labeled edge (i, j) in G, i is a vertex such that the only incoming edge is a labeled edge, and j is a vertex such that the only outgoing edges is a labeled edge. Thus, given a feedback arc set F containing a non-labeled edge in a clause gadget, we can find another feedback arc set F' with $|F'| \leq |F|$ by replacing each non-labeled edge with either of its adjacent labeled edges. □

Lemma 11. *The minimum feedback arc set for the graph G contains $n+3m+u$ edges, where u is the minimum number of unsatisfied equations.*

Proof. By Lemma 9, exactly one edge from every variable gadget is in the minimum feedback arc set. In addition, we show the following things:

(i) For a clause, $x+y+z=0$, and an assignment that satisfies this clause, we need to remove only three edges from the corresponding clause gadget so that the subgraph consisting of the clause gadget and its three corresponding variable gadgets is acyclic. By Lemma 10, we only need to remove labeled edges from the clause gadget. There are four satisfying assignments for the variables x, y, z in this clause. They are: $\{\{0,0,0\}, \{0,1,1\}, \{1,1,0\}, \{1,0,1\}\}$. For each of these four assignments, we remove the three edges with the opposite assignment from the clause gadget. For example, for the assignment $x = y = z = 0$, we remove the edges labeled $x = 1, y = 1, z = 1$.

(ii) For a clause, $x + y + z = 0$, and any assignment that does not satisfy this clause, we need to remove four edges from the corresponding clause gadget so that the subgraph consisting of the clause gadget and its three corresponding variable gadgets is acyclic. There are four assignments to the variables x, y, z that do not satisfy this clause. They are: $\{\{1,1,1\}, \{0,0,1\}, \{1,0,0\}, \{0,1,0\}\}$. For each of these four assignments, if we remove the labeled edges corresponding to the opposite assignment, then the clause gadget will still contain a cycle. For example, for the assignment $x = y = z = 1$, we remove edges labeled $x = 0, y = 0, z = 0$. However, the edges labeled $x = 1, y = 1, z = 1$ remain and form a cycle. So we must remove one more of these edges for the resulting subgraph of the clause gadget to be acyclic.

(iii) For each variable gadget, if we remove one of the edges in the corresponding 2-cycle and the corresponding edge from the clause gadgets representing clauses that contain this variable, then the resulting graph does not contain any cycle composed of edges from multiple clause gadgets. For the clause $x+y+z=0$, consider the edge (x_2, x_1). If a cycle contains this edge, it must also contain the only incoming edge to vertex x_2, which is the edge labeled $x = 1$. If these edges are contained in a cycle with edges from another clause gadget, then at vertex x_1, we can move to another gadget. However, we arrive at a vertex such that the only out edge corresponds to the edge that remains iff x has been set to 0, which is not the case if the edge labeled $x = 1$ was present. So there cannot be any cycles that use edges from more than one clause gadget.

It follows from (i),(ii), and (iii) that the minimum feedback arc set has size $n + 3m + u$. □

Corollary 3. *The Maximum Acyclic Subgraph for G is of size $n + 33s + 32u$ where s and u represent the number of satisfied and unsatisfied clauses, respectively, for an assignment that satisfies the maximum number of clauses.*

Proof of Theorem 7. By Corollary 3 and by Theorem 6 it is NP-hard to distinguish between a graph that has a maximum acyclic subgraph of size $n + 33(\frac{1}{2} + \epsilon)m + 32(\frac{1}{2} - \epsilon)m$ and a graph that has a maximum acyclic subgraph of size $n + 33m$. If we could approximate the maximum acyclic subgraph to within $\frac{2n+65}{2n+66} + \epsilon$, then we could distinguish between these two cases. Therefore it is NP-hard to approximate to approximate the maximum acyclic subgraph to within $\frac{2n+65}{2n+66} + \epsilon$. We can make n arbitrarily small compared to m by creating another set of linear equations in which each original equation appears k times for some k so that we have km clauses and only n variables. The ratio $\frac{2n+65}{2n+66}$ is arbitrarily close to $\frac{65}{66}$ as k becomes large. Therefore, it is NP-hard to approximate the maximum acyclic subgraph to within $\frac{65}{66} + \epsilon$ for any$\epsilon > 0$. □

References

1. Alberto Caprara and Matteo Fischetti, 1/2-Chvatal-Gomory cuts, *Mathematical Programming*, 74A, 1996, 221–235.
2. P. Erdös and H. Sachs, Regulare Graphe gegebener Taillenweite mit minimaler Knotenzahl, Wiss. Z. Univ. Halle-Wittenberg, *Math. Nat.* 12, 1963, 251–258.
3. M. X. Goemans and L. A. Hall, The Strongest Facets of the Acyclic Subgraph Polytope are Unknown, *Proceedings of IPCO 1996*, 415–429.
4. M. Grötschel, M. Jünger, and G. Reinelt, On the Maximum Acyclic Subgraph Polytope, *Mathematical Programming*, 33, 1985, 28–42.
5. M. Grötschel, M. Jünger, and G. Reinelt, Facets of the Linear Ordering Polytope, *Mathematical Programming*, 33, 1985, 43–60.
6. M. Grötschel, L. Lovász and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer-Verlag, New York, 1988.
7. Johan Håstad, Some Optimal Inapproximability Results, *Proceedings of STOC*, 1997.
8. Richard M. Karp, Reducibility Among Combinatorial Problems, *Complexity of Computer Computations*, Plenum Press, 1972.
9. Rudolf Mueller, On the Partial Order Polytope of a Digraph, *Mathematical Programming*, 73, 1996, 31–49.
10. Alantha Newman, Approximating the Maximum Acyclic Subgraph, M.S. Thesis, MIT, June 2000.
11. S. Poljak, Polyhedral and Eigenvalue Approximations of the Max-Cut Problem, *Sets, Graphs and Numbers, Colloq. Math. Soc. Janos Bolyai* 60, 1992, 569–581.
12. Andreas Schulz and Rudolf Mueller, The Interval Order Polytope of a Digraph, *Proceedings of IPCO 1995*, 50–64.