The Weight Function Lemma for Graph Pebbling

Glenn Hurlbert[*]
Department of Mathematics and Applied Mathematics
Virginia Commonwealth University
Richmond, VA 23284-2014 USA
ghurlbert@vcu.edu

October 21, 2015

Abstract

Graph pebbling is a network model for studying whether or not a given supply of discrete pebbles can satisfy a given demand via pebbling moves. A pebbling move across an edge of a graph takes two pebbles from one endpoint and places one pebble at the other endpoint; the other pebble is lost in transit as a toll. It has been shown that deciding whether a supply can meet a demand on a graph is NP-complete. The pebbling number of a graph is the smallest t such that every supply of t pebbles can satisfy every demand of one pebble. Deciding if the pebbling number is at most k is Π_2^{P}-complete.

In this paper we develop a tool, called the Weight Function Lemma, for computing upper bounds and sometimes exact values for pebbling numbers with the assistance of linear optimization. With this tool we are able to calculate the pebbling numbers of much larger graphs than in previous algorithms, and much more quickly as well. We also obtain results for many families of graphs, in many cases by hand, with much simpler and remarkably shorter proofs than given in previously existing arguments (certificates typically of size at most the number of vertices times the maximum degree), especially for highly symmetric graphs.

Here we apply the Weight Function Lemma to several specific graphs, including the Petersen, Lemke, 4th weak Bruhat, and Lemke squared, as well as to a number of infinite families of graphs, such as trees, cycles, graph powers of cycles, cubes, and some generalized Petersen and Coxeter graphs. In doing so we partly answer a question of Pachter, et al., by computing the pebbling exponent of cycles to within an asymptotically small range. It is conceivable that this method yields an approximation algorithm for graph pebbling.

[*] Research supported by Simons Foundation grant #246436.

1 Introduction

Graph pebbling is like a number of network models, including network flow, transportation, and supply chain, in that one must move some commodity from a set of sources to a set of sinks optimally according to certain constraints. Network flow constraints restrict flow along edges and conserve flow through vertices, and the goal is to maximize the amount of commodity reaching the sinks. The transportation model includes per unit costs along edges and aims to minimize the total cost of shipments that satisfy the source supplies and sink demands. At its simplest, the supply chain model ignores transportation costs while seeking to satisfy demands with minimum inventory. The graph pebbling model introduced by Chung [8] also tries to meet demands with minimum inventory, but constrains movement across an edge by the loss of the commodity itself, much like an oil tanker using up the fuel it transports, not unlike heat or other energy dissipating during transfer.

Specifically, a *configuration* C of pebbles on the vertices of a connected graph G is a function $C : V(G) \rightarrow \mathbb{N}$ (the nonnegative integers), so that $C(v)$ counts the number of pebbles placed on the vertex v. We write $|C|$ for the *size* $\sum_v C(v)$ of C; i.e. the number of pebbles in the configuration. A *pebbling step* from a vertex u to one of its neighbors v reduces $C(u)$ by two and increases $C(v)$ by one (so that one can think of it as moving one pebble at the *cost* of another as toll). Given two configurations C and D we say that C is *D-solvable* if some sequence of pebbling steps converts C to D. In this paper we study the traditional case in which the target distribution consists of a single pebble at some *root* vertex r (one can peruse [18, 19, 22, 23] for a wide array of variations on this theme). We are concerned with determining $\pi(G, r)$, the minimum number t of pebbles so that every configuration of size t is r-solvable. Then the *pebbling number* of G equals $\pi(G) = \max_r \pi(G, r)$. Alternatively, $\pi(G)$ is one more than the maximum s such that there is some root r and some size s configuration C so that C does not solve r. The primary focus of this paper is to exploit this duality with newly discovered algebraic constraints.

1.1 Calculating Pebbling Numbers

Given a graph G, configuration C, and root r, one can ask how difficult it is to determine if C solves r. In [24] it was determined that this problem is NP-hard. Subsequently, [25, 28] proved that the problem is NP-complete, with [25] showing further that answering the question "is $\pi(G) \leq k$?" is Π_2^{P}-complete (and hence both NP-hard and coNP-hard, and therefore in neither NP nor coNP unless NP = coNP). Finding classes of graphs on which we can answer more quickly is therefore relevent, and there is some evidence that one can be successful in this direction. Besides what we share in this introduction, we show later that many graphs can have very short certificates that $\pi(G) \leq k$.

The r-unsolvable configuration with one pebble on every vertex other than the root r shows that $\pi(G) \geq n$, where $n = n(G)$ denotes the number of vertices of G. In [26] it is proved that graphs of diameter two satisfy $\pi(G) \leq n + 1$, with a characterization separating the two classes (*Class 0* means $\pi(G) = n$ and *Class 1* means $\pi(G) = n + 1$) given in [6, 9]. One of the consequences of this is that 3-connected diameter two graphs are Class 0. As an extension it is proved in [14] that 2^{2d+3}-connected diameter d graphs are also Class 0, and they use this result to show that almost every graph with significantly more than $n(n \lg n)^{1/d}$ (for any fixed d) edges is Class 0. Consequently, it is a very (asymptotically) small collection of graphs that cause all the problems.

Knowing the pebbling number of a graph and actually solving a particular configuration are two different things, as even a configuration that is known to be solvable (say, one of size equal to the pebbling number; called *large*, as opposed to *small*) can be difficult to solve. Evidence that most configurations are not so difficult, though, comes in the following form. The work of [3] shows that every infinite graph sequence $\mathcal{G} = (G_1, G_2, \ldots, G_N, \ldots)$ has a *pebbling threshold* $\tau_{\mathcal{G}} : \mathbb{N} \rightarrow \mathbb{N}$, which yields the property that almost every configuration C_N on G_N of size $|C_N| \gg \tau(N)$ is solvable (and almost every configuration of size $|C_N| \ll \tau(N)$ is not). In papers such as [4, 12, 13] we find that $\tau_{\mathcal{G}}(N)$ is significantly smaller than $\pi(G_N)$ — for example, $\sqrt{N}$ as opposed to N for the complete graph K_N, and roughly $N2^{\sqrt{\lg N}}$ as opposed to 2^{N-1} for the path P_N. Moreover, the proof techniques reveal that almost all of these solvable configurations can be solved *greedily*, meaning that every pebbling step reduces the distance of the pebble to the root. So the hardness of the problem stems from a rare collection of configurations.

With these results as backdrop, [2] presents a polynomial algorithm for determining the solvability of a large configuration on diameter two graphs of connectivity some fixed κ (whereas the problem was shown in [11] to be NP-complete for small configurations). In fact, [17] present an algorithm to k-fold r-solve any configuration of size at least $\pi(G, r)+4k-4$ on a diameter two graph G in at most $6n+\min\{3k, m\}$ steps, where $m = |E(G)|$. Interestingly, [10] shows that r-solvability is NP-complete over planar graphs also, while it is polynomial over diameter two planar graphs. Also, the proof in [8] that the d-dimensional cube is of Class 0 is a polynomial algorithm (actually bounded by its number of edges $n \lg n$) for large configurations (while no algorithm is known for small ones). Furthermore, [27] contains an algorithm that calculates pebbling numbers, and is able to complete the task for every graph on at most 9 vertices. An order n^4 algorithm for calculating $\pi(G)$ when G has diameter two is given in [17], and [1] present such an algorithm for split graph that runs in $O(n^{\beta})$ time, where $\omega \cong 2.37$ is the exponent of matrix multiplication and $\beta = 2\omega/(\omega+1) \cong 1.41$. Along these lines, our main objective is to develop algorithmic tools that will in a reasonable amount of time yield good upper bounds on $\pi(G)$ for much larger graphs, and in particular decide in some cases whether or not a graph is of Class 0.

This latter determination is motivated most by the following conjecture of Graham in [8]. For graphs G and H, let $G\square H$ denote the *Cartesian product* whose vertices are $V(G\square H) = V(G) \times V(H)$, with edges $(u, x) \sim (v, x)$ whenever $u \sim v$ in G and $(u, x) \sim (u, y)$ whenever $x \sim y$ in H.

Conjecture 1 (Graham) *Every pair of graphs G and H satisfy $\pi(G\square H) \leq \pi(G)\pi(H)$.*

The conjecture has been verified for many graphs; see [16] for the most recent work. However, as noted in [20], there is good reason to suspect that $L\square L$ might be a counterexample to this conjecture, if one exists, where L is the *Lemke* graph of Figure 1. Since L is Class 0, Graham's conjecture requires that $L\square L$ be also, but it is a formiddable challenge to compute

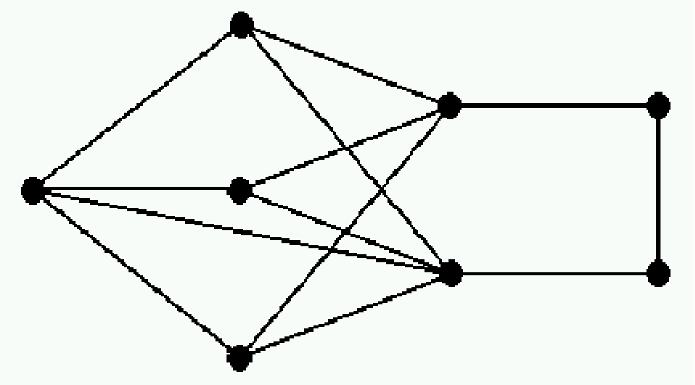

Figure 1: The Lemke graph

the pebbling number of a graph on 64 vertices. One hopes that graph structure and symmetry will be of use, but purely graphical methods have failed to date. The methods of this paper represent the first strides toward the computational resolution of the $64 question[1], "Is $\pi(L\square L) \leq 64$?". Certainly, these methods alone will not suffice[2], but if they produce a decent upper bound then the methods of [27] might be able to finish the job.

1.2 Results

The main tool we develop is the Weight Function Lemma 2. This lemma allows us to define a (very large) integer linear optimization problem that yields an upper bound on the pebbling number. This has several important consequences, including the following.

1. The pebbling numbers of reasonably small graphs often can be computed easily. Moreover, it is frequently the case that the fractional relaxation suffices for the task, allowing the computation for somewhat larger graphs.

2. It is also common that only a small portion of the constraints are required, expanding the pool of computable graphs even more.[3] One can restrict the types of constraints to greedy, bounded depth, and so on, with great success, seemingly because of the comments above. Potentially, this allows one to begin to catalog special classes of graphs such as Class 0, (semi-)greedy, and tree-solvable.

3. The dual solutions often yield very short certificates of the results, in most cases quadratic in the number of vertices, and usually at most the number of vertices times the degree of the root. These certificates are remarkably simple compared to the usual solvability arguments that chase pebbles all over the graph in a barrage of cases. One can sometimes find such certificates for infinite families of graphs by hand, without

[1] Yes, I'll pay if you beat me to it!

[2] We obtain evidence in [21] that $\pi(L\square L) \leq 108$ — in fact, for one root r we show $\pi(L\square L, r) \leq 68$.

[3] We present some findings along these lines in [21], with graphs on 15 and 20 vertices.

resorting to machine for more than the smallest one or two of its members. This is our approach in Section 3.2, for example.

4. Our method gives trivial proofs of

   (a) $\pi(C_{2k}) = 2^k$ and $\pi(C_{2k+1}) = 2\lfloor 2^{k+1}/3\rfloor + 1$, which we write as $\lceil (2^{k+2} - 1)/3\rceil$, and

   (b) $C_n^{(k)}$ is Class 0 for $k \geq n/2(\lg n - \lg\lg n)$, where $G^{(k)}$ denotes the k^{th} *graph power* of G (as opposed to the *Cartesian* power G^k). This answers a question of [26], who defined the *pebbling exponent* of G minimum such $e = e_\pi(G)$ for which $\pi(G^{(e)}) = n(G)$. Thus $e_\pi(G) \leq n/2(\lg n - \lg\lg n)$ (see Theorem 14), which is fairly close to the obvious lower bound of $n/2\lg n$.

In this paper we apply the Weight Function Lemma to several specific graphs, including the Petersen, Lemke, 4th weak Bruhat, and Lemke squared, as well as to a number of infinite families of graphs, such as trees, cycles, graph powers of cycles, cubes, and some generalized Petersen and Coxeter graphs.

2 The Weight Function Lemma

Let P_n be the path $v_1v_2\cdots v_n$ on n vertices. Then $\pi(P_n) = 2^{n-1}$ is easily proved by induction. In particular, any configuration of at least 2^{n-1} pebbles solves v_1. But one can say more about smaller v_1-solvable configurations as well, with the use of a weight function w. Define w on $V(P_n)$ by $w(v_{n-i}) = 2^i$, and extend the weight function to configurations by $w(C) = \sum_{v\in V} w(v)C(v)$. Then a pebbling step can only preserve or decrease the weight of a configuration. Since the weight of a configuration with a pebble on v_1 is at least 2^{n-1}, we see that 2^{n-1} is a lower bound on the weight of every v_1-solvable configuration. In fact, induction shows that every v_1-unsolvable configuration has weight at most $2^{n-1} - 1$, which equals $\sum_{i=2}^n w(v_i)$. That is, this inequality characterizes v_1-unsolvable configurations on P_n. The Weight Function Lemma 2 generalizes this result to trees, and we explore the applications of the lemma in the following sections.

2.1 Linear Optimization

Let G be a graph and T be a subtree of G rooted at vertex r, with at least two vertices. For a vertex $v \in V(T)$ let v^+ denote the *parent* of v; i.e. the T-neighbor of v that is one step closer to r (we also say that v is a *child* of v^+). We call T a *strategy* when we associate with it a nonnegative, nonzero weight function w with the property that $w(r) = 0$ and $w(v^+) = 2w(v)$ for every other vertex that is not a neighbor of r (and $w(v) = 0$ for vertices not in T). Let $\mathbf{T}$ be the configuration with $\mathbf{T}(r) = 0$, $\mathbf{T}(v) = 1$ for all $v \in V(T)$, and $\mathbf{T}(v) = 0$ everywhere else. With this notation note that the path result above can be restated: C is v_1-unsolvable if and only if $C(v_1) = 0$ and $w(C) \leq w(\mathbf{T})$, where T is the strategy $T = P_n$ with associated weight function w.

Lemma 2 [Weight Function Lemma] *Let T be a strategy of G rooted at r, with associated weight function w. Suppose that C is an r-unsolvable configuration of pebbles on $V(G)$. Then $w(C) \le w(\mathbf{T})$.*

Proof. By contrapositive and induction. The base case is when T is a path, which is proved above. Suppose $w(C) > w(\mathbf{T})$, let y be a leaf of T, and define P to be the path from y to r in T, with P_y being the subpath from y to its closest vertex x on P of degree at least 3 in T (or r if none exists). Denote by T' the tree $T - P_y + x$, and among all such r-unsolvable configurations, choose C to be the one having largest weight on T'. The restriction w' of w to T' witnesses that T' is a strategy for the root r, so induction requires that $w'(C) \le w'(\mathbf{T}')$. Likewise, the restriction w_y of w to P_y witnesses that P_y is a strategy for the root x. Because $w(C) = w'(C) + w_y(C)$ and $w(\mathbf{T}) = w'(\mathbf{T}') + w_y(P_y)$, we must have $w_y(C) > w_y(P_y)$ which by induction means that the restriction of C to $P_y - x$ solves x (and so $x \ne r$). Let C_x be the resulting configuration after moving a pebble from $P_y - x$ to x. Since C is r-unsolvable, so is C_x. Now $w(C_x) = w(C)$, but $w'(C_x) > w'(C)$, which contradicts the initial choice of C. □

For a graph G and root vertex r, let $\mathcal{T}$ be the set of all r-strategies in G, and denote by $z_{G,r}$ the optimal value of the integer linear optimization problem $\mathbf{P}_{G,r}$:

$$\text{Max. } \sum_{v \ne r} C(v) \text{ s.t. } w(C) \le w(\mathbf{T}), \text{ and } \mathbf{T} \in \mathcal{T} \text{ with witnessing weight function } w\ . \qquad (1)$$

We also let $\hat{z}_{G,r}$ be the optimum of the relaxation, which allows configurations to be rational. We will find the relation $z_{G,r} \le \lfloor \hat{z}_{G,r} \rfloor$ useful at times. The following corollary is straightforward.

Corollary 3 *Every graph G and root r satisfies $\pi(G, r) \le z_{G,r} + 1$.*

Proof. By definition, the pebbling number is one more than the size of the largest unsolvable configuration. □

Until now, one could only use trees in an individual manner: $\pi(G, r) \le \pi(T, r)$ for every spanning tree T rooted at r. The Weight Function Lemma allows one to consider all subtrees rooted at r (not only spanning trees) simultaneously, which we will see is significantly more powerful. One strength of the method is that the relaxation frequently has an integer optimum. This means that the dual solution will point out which tree constraints certify the result, and because the dual problem has only $n(G) - 1$ constraints there are at most that many such trees in the certificate. Experience has shown, however, that usually one can find a certificate with only $\mathsf{deg}(r)$ trees (or sometimes a few extra). We will see this behavior starting in Section 3.

2.2 Basic Applications

We begin with the pebbling number of trees, whose formula was first discovered and proved in [8]. View a tree T with root r as a directed graph with every edge directed toward r. Then an *r-path partition* $\mathcal{P}$ of T is a set of edge-disjoint directed paths whose union is T. One r-path partition *majorizes* another if its nonincreasing sequence of path lengths majorizes that of the other. An r-path partition is *maximum* if it majorizes all others. We can use Corollary 3 to give a new proof of the following result of [8].

Theorem 4 *For a tree T and root r let $\mathcal{P}^*$ be a maximum r-path partition of T. Then we have $\pi(T,r)=\sum_{P\in\mathcal{P}^*}2^{e_P}-|\mathcal{P}^*|+1$, where e_P denotes the length (number of edges) of P.*

Proof. We begin by showing that a maximum size r-unsolvable configuration has pebbles on leaves only, and in fact on all leaves. Indeed, if C has a pebble on the nonleaf x, then we define a *pushback* of C at x to be any configuration obtained by removing the $C(x)$ pebbles from x, adding $2C(x)$ pebbles to one of the children of x. Certainly, if C is r-unsolvable then the pushback will also be r-unsolvable, and thus satisfy the constraints of $\mathbf{P}_{T,r}$. It will also be larger than C. The configuration C^* that places $2^{e_P}-1$ pebbles on the leaf of the path $P\in\mathcal{P}^*$ is one possible result of pushing back the empty configuration, and so satisfies the constraints of $\mathbf{P}_{T,r}$. Hence $\pi(T,r)\geq\sum_{P\in\mathcal{P}^*}2^{e_P}-|\mathcal{P}^*|+1$.

For the upper bound we prove that C^* is optimal by using induction to show that the optimal configuration has $2^{e_P}-1$ pebbles on the leaf y_P of P for every $P\in\mathcal{P}^*$. This is true if $|\mathcal{P}^*|=1$, so suppose $|\mathcal{P}^*|\geq 2$ and let Q denote one of the paths in $\mathcal{P}^*$ whose leaf has the highest weight in w, with a tie going to one of the shortest length. This is to guarantee that the graph $T-Q+x$, where x is the root of Q, is a tree (i.e. is connected). In order to maximize the number of pebbles that satisfy $\sum_{P\in\mathcal{P}^*}w(y_P)C(y_P)\leq\sum_{P\in\mathcal{P}^*}w(P)$ we would transfer as many pebbles from y_Q to other leaves as possible because their weights are at most $w(y_P)$ and we could add extra pebbles when the weight is smaller. But by induction on $T-Q+x$ we know from the constraint $\sum_{P\in\mathcal{P}^*-\{Q\}}w(y_P)C(y_P)\leq\sum_{P\in\mathcal{P}^*-\{Q\}}w(P)$ that each $C(y_P)\leq 2^{e_P}-1$, with equality for all $P\neq Q$ if and only if C is maximum. Therefore, since $w(P)=w(y_P)(2^{e_P}-1)$ for all $P\in\mathcal{P}^*$, we have

$$\begin{aligned} w(y_P)C(y_P) &\leq \sum_{P\in\mathcal{P}^*}w(P)-\sum_{P\in\mathcal{P}^*-\{Q\}}w(y_P)(2^{e_P}-1)\\ &= w(Q)\\ &= w(y_Q)(2^{e_Q}-1)\ , \end{aligned}$$

which implies that $C(y_P)\leq 2^{e_Q}-1$. □

A slight weakness of these tree constraints is that they do not classify unsolvable configurations on trees the way that they do on paths. This is because they let in a few solvable configurations. For example, consider the star on four vertices with one of its leaves as root r. Then the configuration with 2 pebbles on each of the other two leaves is r-solvable and

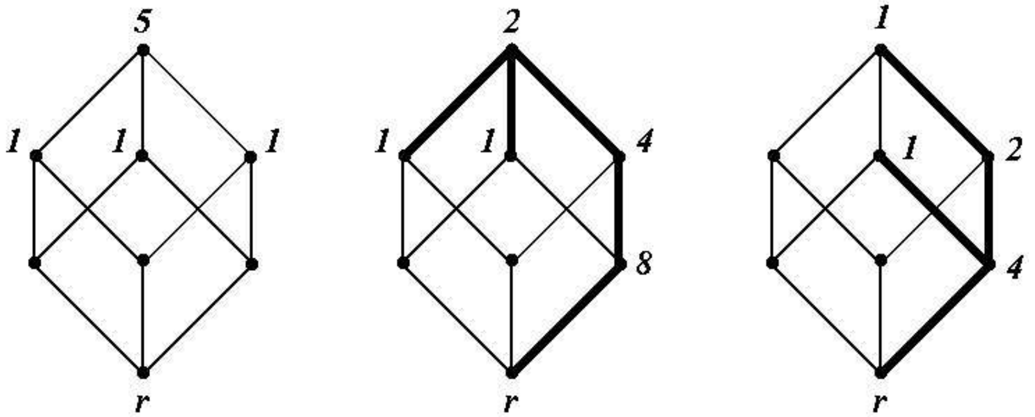

Figure 2: A solvable configuration (left) not recognized by any tree strategy; a canonical strategy (center) used for certifying $\pi(Q^3) \leq 9$; a simplified nonbasic strategy (right) that does the same

satisfies all tree contraints. Since it is the average of the two r-unsolvable configurations that place either 1 and 3 or 3 and 1 on those other leaves, it cannot be cut out by the tree constraints that don't cut out at least one of these two other constraints. In this case it doesn't hurt us, since the strategy bound yields $\pi \leq 5$ and the actual pebbling number is 5, but it can cause trouble on graphs in general. For example, we know that the 3-cube Q^3 in Figure 2 has pebbling number 8, so that the shown configuration C is solvable (pebbles from to top must be split in two directions in its solution). However, no strategy recognizes its solution, and Corollary 3 yields only $\pi(Q^3) \leq 9$ (the three rotations of the strategy in the center in Figure 2 certify this). One can see where the aforementioned star appears in the Figure 2 configuration on Q^3 and is exploited accordingly: moving pebbles from the 5 along one edge yields a $(3, 1)$ configuration, while splitting the moves along two edges yields a $(2, 2)$ configuration.

3 General Applications

In this section we illustrate the method more fully by presenting short proofs of both known and new results. We begin by relaxing strategies in the following way. We now use the term *basic* to describe the strategies as currently defined. A *nonbasic strategy* will satisfy the inequality $w(v^+) \geq 2w(v)$ in place of the equality used in a basic strategy (see Figure 2). The following lemma shows that we can use nonbasic strategies in an upper bound certificate since they are conic combinations of a nested family of basic strategies. Thus the use of nonbasic strategies can simplify and shorten certificates significantly.

Lemma 5 *If T is a nonbasic strategy for the rooted graph (G, r), then there exists basic strategies $T_1, \ldots, T_k$ for (G, r) and nonnegative constants $c_1, \ldots, c_k$ so that $T = \sum_{i=1}^{k} c_i T_i$.*

Proof. We use induction, as the result is true when T has two vertices since T is basic then. Given T, let S be a basic strategy on the edge set of T, define c to be the largest constant for which $cS \leq T$, and denote $T' = T - cS$. Then some vertex v of G satisfies $cS(v) = T(v)$, so T' has fewer vertices than T. Also, because S is basic, any vertex u whose unique ur-path contains v also satisfies $cS(u) = T(u)$, which means that T' is connected, and hence a strategy. Moreover, T' is nonbasic since every nonneighbor x of r has $T'(x^+) = T(x^+) - cS(x^+) \geq 2T(x) - 2cS(x) = 2T'(x)$. By induction, T' is a conic combination of basic strategies, and so therefore is T. □

We use conic combinations of strategies to derive, for some α, the inequality $|C| = \sum_{v \neq r} C(v) \leq \alpha$ for r-unsolvable configurations C. From this we surmise that $\pi(G) \leq \lfloor \alpha \rfloor + 1$. Instead of writing our strategies algebraically, it will be somewhat easier to show them graphically. We will display them so as to derive $m \sum_{v \neq r} C(v) \leq \sum_{v \neq r} m_v C(v) \leq m\alpha$ for some sequence $\{m_v\}_v$ with $m = \min_v m_v$, and let the reader divide by m. In fact, in many instances we will derive $m_v = m$ for all $v \neq r$, which makes for the following observation.

Lemma 6 [Uniform Covering Lemma] *Let $\mathcal{T}$ be a set of strategies for the root r of the graph G. If there is some m such that, for each vertex $v \neq r$, we have $\sum_{T \in \mathcal{T}} T(v) = m$, then $\pi(G, r) = n(G)$.* □

3.1 Specific Graphs

It has been said in jest that every graph theory paper should contain the Petersen graph, so we get it out of the way first.

Theorem 7 *Let P denote the Petersen graph. Then $\pi(P) \leq 10$.*

Of course, the vertex lower bound implies $\pi(P) = 10$, but since the focus of this paper regards upper bounds, we prove them only.
Proof. The 3 strategies shown in Figure 3 certify the result. □

Without such nice symmetry, the Lemke graph requires a different certificate for each possible root.

Theorem 8 *Let L denote the Lemke graph. Then $\pi(L, r) \leq 8$.*

Proof. We refer the reader to [21] for pictures of short certificates for each possible root vertex. □

To illustrate that larger graphs can be tackled, we move on to one of order 24 that is not Class 0. Define the (weak) *Bruhat* graph of order m (see Figure 4) to have all

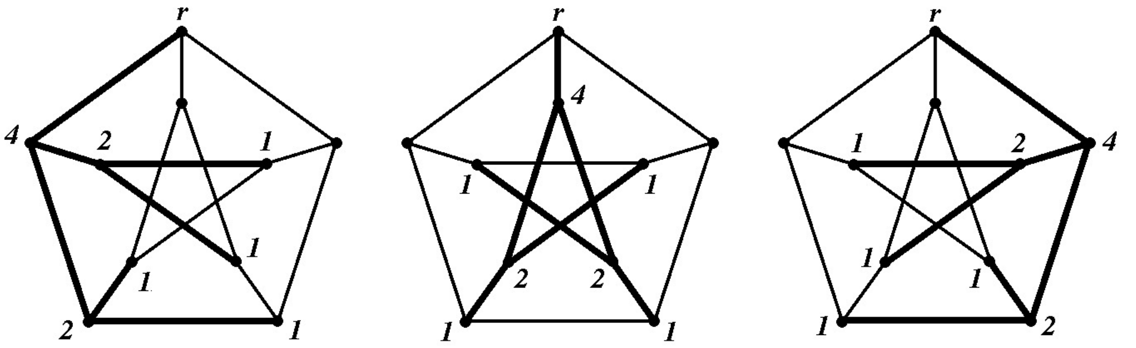

Figure 3: Petersen Class 0 certificate

permutations of $[m]$ as vertices, with an edge between pairs of permutations that differ by an adjacent transposition. One can recognize it as the Cayley graph of S_m, generated by adjacent transpositions, and also note that B_4 is the cubic Ramanujan (expander) graph of [7]. Intuitively, expander graphs would seem to have low pebbling numbers, but because B_4 has diameter 6 we have the lower bound $\pi(B_4) \geq 64$. We give here a fairly tight bound.

Theorem 9 *Let B_4 be the Bruhat graph of order 4. Then $\pi(B_4) \leq 72$.*

Proof. Because the graph is vertex transitive, only one root must be checked. The 3 strategies shown in Figure 4 certify the result. We combined them into one figure, separated by edge styles. □

Work in [21] gives results on $\pi(L\square L, r)$ for various roots r, where $L\square L$ is the square of the Lemke graph L. Because L has diameter 3, $L\square L$ has diameter 6, and so $\pi(L\square L) \geq 64$. Among the results that strategies deliver is the following upper bound.

Theorem 10 *Let $L\square L$ be the square of the Lemke graph L. Then $\pi(L\square L, (v_4, v_4)) \leq 68$.* □

3.2 Graph Classes

Next we turn our attention to classes of graphs, and begin with an extremely simple proof of the pebbling numbers of cycles, first proved in [26].

Theorem 11 *For $k \geq 1$ we have $\pi(C_{2k}) = 2^k$ and $\pi(C_{2k+1}) = \lceil (2^{k+2} - 1)/3 \rceil$.*

Proof. For both results we use two basic strategies: one path in each direction. For even cycles the paths of length k will overlap in the vertex opposite the root. This yields $\pi(C_{2k}) \leq 2(2^k - 1)/2 + 1 = 2^k$. For odd cycles with $k \geq 3$ the paths will be of length $k + 3$, which gives $\pi(C_{2k+1}) \leq \lfloor 2(2^{k+3} - 1)/(2^2 + 2^3) \rfloor + 1 = \lfloor (2^{k+2} - 1/2)/3 \rfloor + 1 = \lceil (2^{k+2} - 1)/3 \rceil$. For

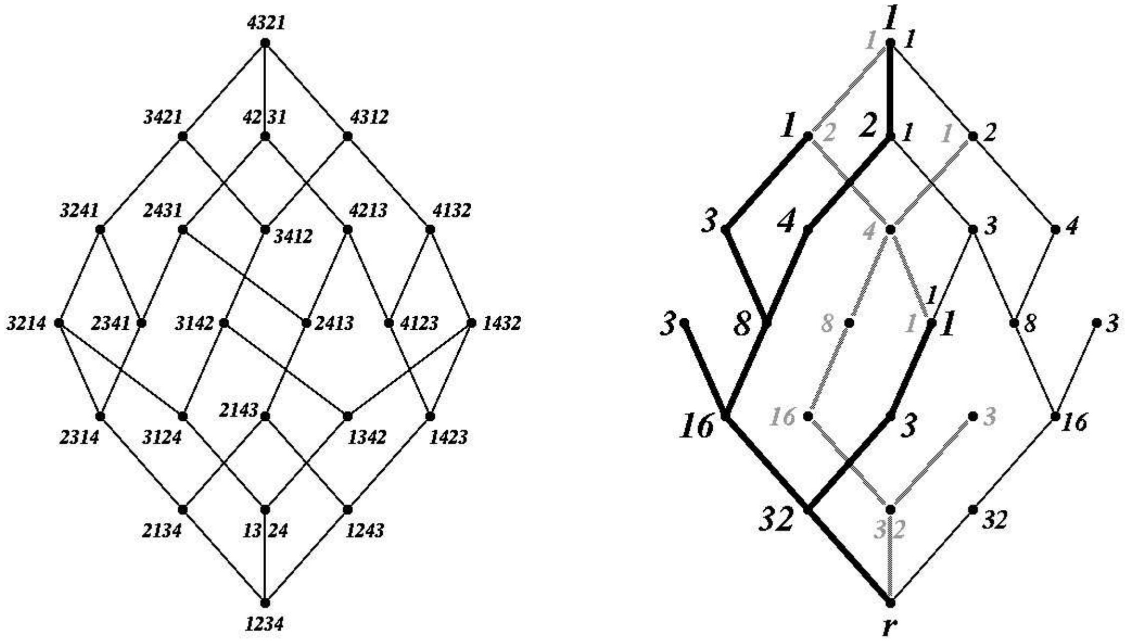

Figure 4: Bruhat graph B_4 (left) and strategies (right)

$k \leq 2$, paths of length $k+1$ suffice. □

Now we consider a generalization of the Petersen graph. For $m \geq 3$ and $d \geq 2$ define $P_{m,d}$ to have vertices u and $v_{i,X}$, where $1 \leq i < m$ and X is a binary k-tuple for some $0 \leq k < d$. Furthermore, $uv_{i,\emptyset}$ is an edge for all i, and $v_{i,X}v_{i,X^-}$ is an edge for all i and nonempty X, where X^- denotes the truncation of X obtained from dropping its final digit. Finally, for every i and length $d-1$ X, we include the edge $v_{i,X}v_{i+1,X}$ (addition modulo m), with the exception that when $i = m-1$ we use $v_{m-1,X}v_{0,X^+}$ instead, where X^+ denotes the $(d-1)$-tuple that satisfies $N(X^+) = N(X) + 1 \pmod{2^{d-1}}$, and $N(X)$ is the natural number represented by X in binary. Figure 5 shows the graph $P_{5,2}$; it is easy to check that $P_{3,2}$ is the Petersen graph P. Also, $P_{m,d}$ has $m2^d - m + 1$ vertices and is diameter $2d$ when $m > 3$ and $2d - 1$ when $m = 3$.

Interest regarding these graphs comes from two sources. In [26] the problem is raised of finding the minimum number $e(n)$ of edges in a Class 0 graph on n vertices. Because of the Petersen and Wheel graphs, we have $e(n) \leq 2n - 2$. Blasiak, et al. [5], show that $e(n) \geq \lfloor 3n/2 \rfloor$, and conjecture that $e(n) = 3n/2 + o(1)$. Furthermore, they believe that the graphs $P_{m,d}$ are Class 0 when $(m,d) = (3,2)$ (which is true) or $m \geq 2^d + 1$ (which is required by needing $n(G) \geq 2^{\mathsf{diam}(G)}$), which, if true, would prove the conjecture because they are 3-regular except for the central vertex u, having degree m (with $n = m(2^d - 1) + 1$). The graphs also appear in [15] as n-vertex graphs having the fewest edges among those of minimum degree 3 and radius d. In fact, the intuition for the conjecture comes from this result.

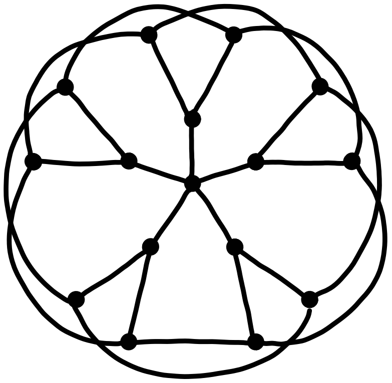

Figure 5: The generalized Petersen graph $P_{5,2}$.

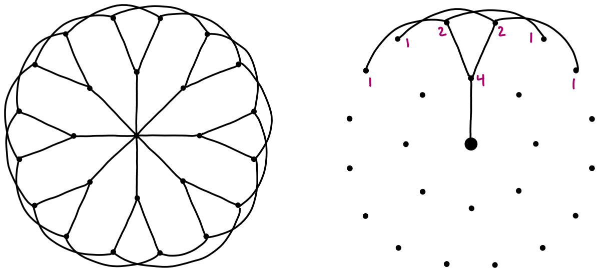

Figure 6: The graph $P_{8,2}$ and its main strategy T for root u.

Here, we are most interested in the graphs $P_{m,2}$ for $m > 3$. Note that $P_{m,2}$ has rotational symmetry: the exceptional edges for $k = 2$ can be "rotated onward" by swapping $v_{1,0}$ with $v_{1,1}$ in the drawing. This makes $P_{m,2}$ transitive on the set $\{v_{i,X}\}$ for fixed length X. Thus, when calculating $\pi(P_{m,2})$, one only need consider the three vertices u, $v = v_{1,\emptyset}$, and $w = v_{1,0}$ as root.

Theorem 12 *For all $m \geq 4$ we have the following, where $n = n(P_{m,2}) = 3m + 1$:*

1. $\pi(P_{m,2}, u) = n$,
2. $\pi(P_{m,2}, v) \leq n + 5$, *and*
3. $\pi(P_{m,2}, w) \leq n + 17$.

Proof. When the root is u, we use all rotations of the following basic strategy T (see Figure 6). Let $v_{1,\emptyset}$ have weight 4, $v_{1,0}$ and $v_{1,1}$ have weight 2, and each of their two neighbors have weight 1. Clearly, of the sum of all rotations of T has weight 4 everywhere but u, and so the Uniform Covering Lemma applies.

When the root is v, we build slightly more complex strategies. First we use the nonbasic strategy S, having weight 3 on $v_{0,0}$ and $v_{0,1}$ and weight 1 on each of their neighbors. Next, write the rotations of T as $T_0, \ldots, T_{m-1}$. For $j \in \{0, 1, 2\}$, we build the basic strategies S_j from these by combining all T_i ($i \neq 0$) with $i \equiv j \mod 3$, with the exception that if

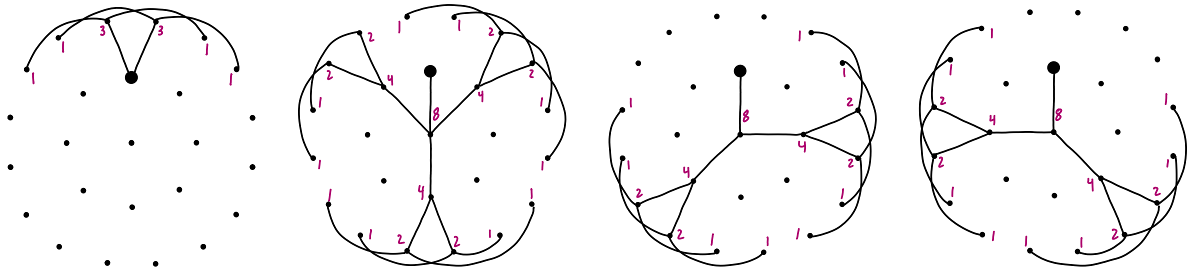

Figure 7: The four strategies for root v in $P_{8,2}$.

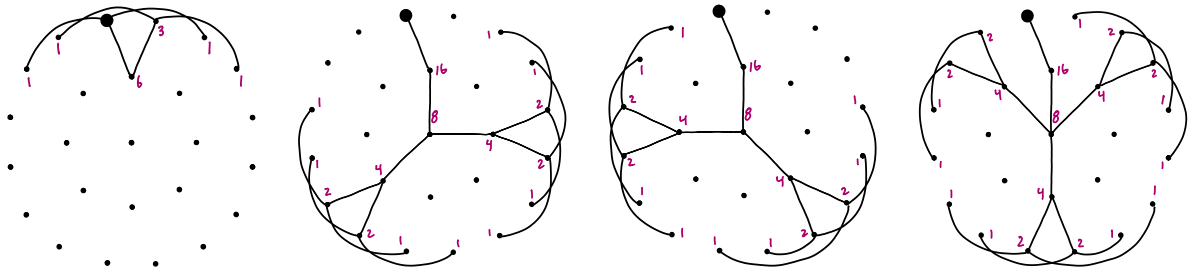

Figure 8: The four strategies for root w in $P_{8,2}$.

$j = 1$ and $i = m - 1$ then we do not include the vertices $v_{0,0}$ and $v_{0,1}$ twice. Moreover, each S_j includes weight 8 on u (see Figure 7). One quarter of the sum of these four strategies has weight 0 on v, weight 6 on u, and weight 1 everywhere else. Now the Weight Function Lemma (actually Corollary 3) applies.

The description for the strategies when the root is w is almost identical to that for root v (see Figure 8). In this case, one quarter of the sum of the four strategies has weight 0 on w, weight 13.5 on v, weight 6 on u, and weight 1 elsewhere. □

Next we define the generalized Coxeter p-graph $C(p)$ for odd primes $p = 2q + 1$. Like the graphs $P_{m,d}$, $C(p)$ has the potential (for $p \geq 5$) to be a Class 0 graph with fewer than $2n - 2$ edges. Set $V_i = \{(i, j) \mid 0 \leq j < p\}$ for $0 \leq i \leq q$ and $V = \cup_{i=0}^{q} V_i$, so that $C(p)$ has $n = p(q + 1) = \binom{p+1}{2}$ vertices. Define the $2pq = 2n - 2p$ edges $(i, j)(i, j + i \pmod{p})$ and $(0, j)(i, j)$ for each $1 \leq i \leq q$ and $0 \leq j < p$. The vertices in V_0 have degree q and all others have degree 3, making $C(7)$, the original Coxeter graph, 3-regular. Here we show the following $n + O(\sqrt{n})$ upper bound.

Theorem 13 *For every prime $p \geq 11$ we have $\pi(C(p)) \leq n + 15q + 12$.*

Proof. First we exhibit the symmetry of $G = C(p)$. To do so, we note that the arithmetic that follows will be modular in q in the first coordinate, with the speciality that we use the representative q in place of 0, and modular in p (as normal) in the second. Let ρ be the automorphism that sends vertices (i, j) to $(i, j + 1)$ for all $0 \leq i \leq q$ and $0 \leq j < p$. This is a rotation of the vertices within each V_i. Let α generate Z_p^* and define σ to be the automorphism that sends $(0,j)$ to $(0,\alpha j)$ and (i,j) to $(i+1,\alpha j)$ for all $1 \leq i \leq q$ and $0 \leq j < p$. While permuting V_0, the key aspect of σ is that it also permutes the sets of vertices from V_i to V_{i+1}. Together, ρ and σ act transitively on V_0 and on $\overline{V_0}$. This means that we only need calculate $\pi(G,(0,0))$ and $\pi(G,(1,0))$. (It turns out that $C(7)$ is fully vertex transitive and so the root $(0,0)$ suffices.)

Next we define the q basic strategies $\mathbf{T}_i$ $(1 \leq i \leq q)$ for the root $r=(0,0)$, with each root child having weight 4. The child of r in $\mathbf{T}_i$ has weight 4 and two children, (i,i) and $(i,-i)$. We describe the descendents of (i,i) only, as those of $(i,-i)$ are identical but with their second coordinate negated. The left subtree under (i,i) is the path $(i,2i)$, $(i,3i)$, …, (i,qi). The right subtree under (i,i) is a collection of $q-2$ paths, starting with $(0,i)$ and its children (k,i) for $1 \leq k \leq q$ with $k \notin \{i, \pm 2i\}$. The child $(k,i)=(k,sjk)$, for some $s \in \{\pm 1\}$ and $1 \leq j \leq q$, and from it hangs the path $(k,s(j+1)k)$, …, (k,sqk).

Now we look at the sum of the weights of vertices over all strategies, and by the symmetries described above we need only consider the vertices (i,j) for $i \in \{0,1\}$ and $0 \leq j \leq q$. We note first that $(0,j)$ is the grandchild of $(j,0)$ in $\mathbf{T}_j$, and so has weight 1. The left path under $(1,1)$ shows that vertex $(1,j)$ has weight 2^{2-j} in $\mathbf{T}_i$. For $j \geq 3$ we see that if $k \leq j$ then vertex $(1,j)$ has weight 2^{k-j-1}. Therefore, except for the q weight 4 children of r and their $2q$ weight 2 children, every other nonroot vertex has weight 1. This gives total weight $(n-1)+(4-1)q+(2-1)2q$, implying that $\pi(G,r) \leq n+5q$.

Consider the other root $r=(1,0)$. Here we need one more basic strategy — we will define $\mathbf{T}_i$ for $i \in \{\pm 1\} \cup \{2,\ldots,q\}$ — and the weights of the children of r will be 8 instead of 4. Similar to above, the strategy $\mathbf{T}_{-1}$ will be identical to $\mathbf{T}_1$ except that the second coordinates will be negated. We will describe the first four levels of each strategy explicitly, and their remaining levels implicitly. The child of r in $\mathbf{T}_1$ is $(1,1)$, having children $(1,2)$ and $(0,1)$. From $(1,2)$ hangs the path $(1,3)$ and $(1,4)$, while the children of $(0,1)$ are $(2,1),\ldots(q,1)$. The children of each $(i,1)$ are $(i,1 \pm i)$, subject to them not having already listed in some other $\mathbf{T}_j$, which we now describe. For $i \in \{2,\ldots,q\}$, the child of r in $\mathbf{T}_i$ is $(0,0)$, which has the single child $(i,0)$. The two children of $(i,0)$ are $(i,\pm i)$, the two children of which are $(i,\pm 2i)$ and $(0,\pm i)$, correspondingly, with the exception that $(i,\pm 2i)$ does not appear in $\mathbf{T}_q$ because it already appears in $\mathbf{T}_{\pm 1}$ (as $(q,\pm 1)$). At this point, there are 1 vertex of weight $8q-8$, 2 vertices of weight 8, $q+3$ vertices of weight 4, and $4q-2$ vertices of weight 2, with all other vertices mentioned having weight 1.

The remaining construction of the strategies proceeds in stages, with each new vertex added with several fractional weights adding up to 1. In the first stage, each vertex (a,b) that is adjacent to a current leaf (a,c) of some strategy is added to that strategy as a child of (a,c), accounting for weight $1/2$. The other $1/2$ weight comes from adding it as a child of $(0,b)$ as well. In each subsequent stage, every vertex (a,b) that is adjacent to a current leaf (a,c) is added as a child of (a,c) in every strategy that contains (a,c), accounting for weight $1/2$. The other $1/2$ weight comes from adding it as a child of $(0,b)$ as well.

Hence we obtain $\pi(G,r) \leq (n-1)+(8q-9)+7(2)+3(q+3)+1(4q-2)+1 = n+15q+12$.
□

We note that the number of strategies for the root $(1,0)$ can be reduced whenever two strategies (not including $\mathbf{T}_{\pm 1}$) share no vertices (other than $(0,0)$). Thus one can define the intersection graph G_p of the family of sets W_i, $2 \leq i \leq q$, where $W_i = V(\mathbf{T}_i) - \{(0,0)\}$. With chromatic number $\chi_p = \chi(G_p)$, we obtain the upper bound of $n + 7\chi_p + 7q + 21$, which is an improvment when $\chi_p < q - 1$. For example, $\chi_p \leq 1$ when $p \leq 7$. In those cases, further improvements can be made as well, and it is not too hard to show that $\pi(C(5)) \leq n + 6$ and $\pi(C(7)) \leq n + 15$.

Finally, we discuss powers of cycles. For a given graph G and integer k, we denote by $G^{(k)}$ the graph on the same vertex set as G, with edges uv whenever the distance $\mathsf{dist}_G(u,v) \leq k$ in G. For example, $G^{(\mathsf{diam}(G))} = K_n$ for every connected G. Pachter, et al [26], define the *pebbling exponent* of G to be the minimum $e = e_\pi(G)$ for which $G^{(e)}$ is Class 0. Consequently $e_\pi(G) \leq \mathsf{diam}(G)$ for all G. The problem raised in [26] is to find $e_\pi(C_n)$. Here we prove the following.

Theorem 14 *The pebbling exponent of the cycle C_n satisfies*

$$\frac{n/2}{\lg n} \leq e_\pi(C_n) \leq \frac{n/2}{\lg n - \lg\lg n} \ .$$

Proof. The lower bound follows from the general fact that $\pi(G) \geq 2^{\mathsf{diam}(G)}$ for all G, along with the observation that $\mathsf{diam}(C_n^{(e)}) = \lceil n/2e \rceil$. Therefore, a requirement for Class 0 is that $n \geq 2^{\lceil n/2e \rceil}$.

For the upper bound, we prove that $\pi(C_n^{(2^k)}) = n$ for $n = (2k+1)2^k + 3 = f(k)$. Then we show that $\pi(C_n^{(2^k)}) = n$ for $f(k-1) < n < f(k)$. Our method will be to split the cycle into two identical paths with endpoints at the root r, and use identical strategies on each one. We will invoke the Uniform Covering Lemma 6 to obtain the result.

In more detail, let us give a useful labeling of the vertices V of C_n as follows. We partition $V = \cup_{i=0}^{k+2}(U_i \cup W_i)$ so that

- each U_i and W_i induces a path in C_n,
- $U_0 = W_0 = \{r\}$,
- U_{i+1} follows U_i when traversing C_n clockwise from r,
- W_{i+1} follows W_i when traversing C_n counterclockwise from r, and
- $U_{k+2} = W_{k+2}$.

The two identical paths mentioned above are seen to be $U = \cup_{i=0}^{k+2} U_i$ and $W = \cup_{i=0}^{k+2} W_i$, where now we see that the term 'split' was a white lie because of the overlap on $U_{k+2} = W_{k+2}$. We will describe a family $\mathcal{T}$ of strategies on U with the property that, for some m (actually 2^{k+1}) and all $u \in \cup_{i=1}^{k+1} U_i$, we have $\sum_{T \in \mathcal{T}} T(u) = m$, while for all $u \in U_{k+2}$ we have $\sum_{T \in \mathcal{T}} T(u) = m/2$. Then we copy these strategies symmetrically onto W and, because of the overlap, the Uniform Covering Lemma 6 applies.

Next we describe the vertices within each U_i. First, $|U_i| = 2^k$ for $i \in \{1, k+2\}$, and $|U_i| = 2^k - 2^{k-i+1}$ for $2 \leq i \leq k+1$ (thus $n = (2k+1)2^k + 3$). We clockwise order the vertices of $U_i = \{v_{i,0}, \ldots, v_{i,|U_i|-1}\}$ in their natural order by subscript, and will find it useful to identify $v_{i,j}$ with the encoding $[i, b_j]$, where b_j is the k-bit binary representation of j (so that leading zeros are not suppressed). For example, $v_{3,6}$ is encoded as $[3, 00110]$ in C_{355} since $k = 5$ in that case. Also, the root r is encoded as $[0, 0^k]$, where we write x^j to denote the concatenation $xx \cdots x$ of length j. Furthermore, we use the notation $\mathbf{v}_j$ to mean some binary word of length j, subject to context. That is, $[1, \mathbf{v}_k]$ is always a vertex in U_1 but $[2, \mathbf{v}_k]$ is not always in U_2 — every vertex in U_2 looks like $[2, 0\mathbf{v}_{k-1}]$. Moreover, for $2 \leq i \leq k+1$, vertices in U_i look like everything but $[i, 1^{i-1}\mathbf{v}_{k-i+1}]$.

Now we describe a partially ordered set P that we will use to define our strategies. The elements of P are the vertices U, and the covering relations are given by

1. $[0, 0^k] > [1, \mathbf{v}_k]$ for all $\mathbf{v}_k$,
2. $[1, \mathbf{v}_{k-1}x] > [2, 0\mathbf{v}_{k-1}]$ for all $\mathbf{v}_{k-1}$ and all $x \in \{0, 1\}$,
3. $[i, \mathbf{v}_{i-1}\mathbf{v}_{k-i}x] > [i+1, \mathbf{v}_{i-1}b\mathbf{v}_{k-i}]$ for all $2 \leq i \leq k$, all $\mathbf{v}_{i-1} \neq 1^{i-1}$, all $\mathbf{v}_{k-i}$, and each $x, b \in \{0, 1\}$,
4. $[i, 1^{i-2}0\mathbf{v}_{k-i}x] > [i+1, 1^{i-1}0\mathbf{v}_{k-i}]$ for all $2 \leq i \leq k$, all $\mathbf{v}_{k-i}$, and all $x \in \{0, 1\}$,
5. $[k+1, \mathbf{v}_k] > [k+2, \mathbf{v}_k]$ for all $\mathbf{v}_k \neq 1^k$, and
6. $[k+1, 1^{k-1}0] > [k+2, 1^d]$.

All other relations of P are determined by transitivity.

For $z \in P$, define its *downset* $D(z) = \{y \in P \mid y < z\}$. Notice that each $D = D([1, \mathbf{v}_k])$ forms a tree in P. Indeed, simple induction shows that if $\mathbf{v}_k = v_1 \cdots v_k$ then every vertex of U_{i+1} in D has the form $[i+1, \mathbf{x}_i v_1 \cdots v_{k-i}]$, where $\mathbf{x}_i$ is anything but 1^i. This means that $[i+1, x_1 \cdots x_i v_1 \cdots v_{k-i}]$ has exactly one element from D that covers it, namely either $[i, x_1 \cdots x_{i-1} v_1 \cdots v_{k-i} v_{k-i+1}]$ or $[i, x_2 \cdots x_i v_1 \cdots v_{k-i} v_{k-i+1}]$. So no cycles exist in D.

For each $\mathbf{v}_k$, then, define the basic strategy $T = T(\mathbf{v}_k)$ to have vertices $\{r\} \cup D([1, \mathbf{v}_k])$, with edge zy whenever $z > y$ is a covering relation in P. In order that T is a strategy in $C_n^{(2^k)}$ we must verify that the distance $d = \mathsf{dist}(z, y)$ between z and y in C_n is at most 2^k. When $z = r$, we have $y = v_{1,j}$ for some $0 \leq j \leq 2^k - 1$, so $d = j + 1$. When $z = v_{1,j}$, we have $y = v_{2,\lfloor j/2 \rfloor}$, so $d = 2^k - \lceil j/2 \rceil$. When $z = v_{i,j}$ and $2 \leq i \leq k+1$, then we can write $j = h2^{k-i+1} + l$ for some $0 \leq l < 2^{k-i+1}$. From relation 3 or 4 we have $y = v_{2,j'}$, where $j' = h2^{k-i+1} + b2^{k-i} + \lfloor l/2 \rfloor$ and $b \in \{0, 1, 2\}$. Thus the greatest distance comes from relation 4, in which case $d = |U_i| - j + j' = 2^k - \lceil l/2 \rceil$.

Note that the characterization of elements in $D([1, \mathbf{v}_k])$ implies that each vertex of U_i is in 2^{i-1} basic strategies when $1 \leq i \leq k+1$, and in 2^k when $i = k+2$. Because each T is basic, this means that $\sum_T T(v) = 2^{k+1}$ for all $v \in U_i$ (resp. W_i), $1 \leq i \leq k+1$, and 2^k for $v \in U_{k+2}$ (resp. W_{k+2}). Now the overlap from $U_{k+2} = W_{k+2}$ gives sum $m = 2^{k+1}$ for all $v \neq r$, and the Uniform Covering Lemma 6 applies.

Finally, whenever n is smaller than $f(k)$ we simply erase sufficiently many vertices of $U_{k+2} = W_{k+2}$ (but don't renumber any indicies/encodings), as they are leaves in all strategies and so don't destroy the uniform covering. When such vertices are exhausted continue erasing vertices of $U_{k+1} \cup W_{k+1}$ with similar results. Since $f(k) - f(k-1) = 2^{k+1}$, no more considerations are necessary. □

4 Remarks

In this paper we have shown several different strengths of the Weight Function Lemma in combination with linear optimization, highlighting its versatility. It has been used to compute upper bounds on and exact values of the pebbling number of small graphs. It has also been successful in calculating the pebbling numbers of much larger graphs than previous algorithms. For such graphs having too many strategies than time allows to construct, the technique of creating a smaller set of them at random seems to perform just as well. This is most likely due to the property that nonoptimal solutions (derived from having fewer constraints) seem, in most instances, to be near optimal (have the same floor function). In fact, by restricting strategies to be breadth-first search, one obtains upper bounds on the greedy pebbling numbers of graphs (which requires pebbling steps to move toward the root). The method also yields results for many families of graphs, in many cases by hand, with much simpler and remarkably shorter proofs than given in previously existing arguments. This is especially so with highly symmetric graphs. We note also that the technique can be used in conjunction with more traditional arguments, as in [21], and it has delivered an array of upper bounds, such as n, $n+c$, and $n+o(n)$, most of which are the best known and might possibly be best possible. It's two main shortcomings are the inability to overcome the kind of splitting structure found in cubes, for example, in which solutions to some configurations require nontree solutions, and the difficulty in dealing with large diameter, although success has been found with cycles and their graph powers, in addition to Petersen and Coxeter generalizations. When the technique gives upper bounds, it would be of great use to know how good the bound might be. That is, does the Weight Function Lemma yield an approximation algorithm for graph pebbling?

Question 15 *Is there a constant c such that, for all graphs G and every root $r \in V(G)$, $z_{G,r} \le c\pi(G,r)$?*

For example, is $c = 2$? The cube Q^d shows that it couldn't be any smaller.

Theorem 16 *For all $d \ge 1$ and all $r \in V(Q^d)$ we have $z_{Q^d,r} < 2\pi(Q^d,r)$.*

Proof. We exploit the symmetry of Q^d as follows. Because Q^d is vertex transitive we need only consider one root r. We identify $V(Q^d)$ with the power set of $\{1,\ldots,d\}$ and take $r = \emptyset$. We define a single strategy $\mathbf{T}$ for r and then apply every permutation of $\{1,\ldots,d\}$ to obtain other strategies. Finally we average over this collection of strategies. The result will be that $\pi(Q^d)$ will be at most one more than the sum of the weights in this average.

For each $1 \leq k \leq d$ such that $2^k \leq \binom{d}{k}$ we define $a_k = \lfloor \binom{d}{k}/2^k \rfloor$ and $b_k = \binom{d}{k} \mod 2^k$. We form $\mathbf{T}$ by first taking a neighbor of r. This neighbor will be a set of size 1, to which we assign the weight 2^{d-1} — this is step $i = 1$, with $k = d-i$. For future steps i, while $2^k > \binom{d}{k}$, we continue adding a single neighbor, a set of size i, to the current leaf of $\mathbf{T}$, and assign the weight 2^k. If $2^k \leq \binom{d}{k}$, however, we add $a_k + 1$ vertices to the current leaves: a_k of these will have weight 2^k and one will have weight b_k. All of these will be connected to leaves of weight 2^{k+1} and none will be connected to leaves of weight b_{k+1}. This is possible because the degree of each vertex of size $i-1$ is $k+1$, and $a_k < (k+1)a_{k+1}$ whenever $k+1 < d-1$, so there are enough potential neighbors to accomplish this.

Hence we obtain weight 1 on average for vertices of size i for which $2^k \leq \binom{d}{k}$, and weight $2^k/\binom{d}{k}$ otherwise. This yields the bound $z_{Q^d,r} \leq 1 + \sum_{k=0}^{d-1} \max\{\binom{d}{k}, 2^k\} < \sum_{k=0}^{d} \binom{d}{k} + \sum_{k=0}^{d-1} 2^k < 2^{d+1} = 2\pi(Q^d)$. □

If one restricts their attention to only polynomially many strategies, this linear optimization technique becomes a polynomial algrorithm. It would be useful to investigate how good the approximation can be under these circumstances.

References

[1] L. Alcón, M. Gutierrez and G. Hurlbert, Pebbling in split graphs, *SIAM J. Discrete Math.* **28**, no. 3, (2014), 1449–1466.

[2] A. Bekmetjev and A. Cusack, Pebbling algorithms in diameter two graphs, *SIAM J. Discrete Math.* **23**, no. 2, (2009), 634–646.

[3] A. Bekmetjev, G. Brightwell, A. Czygrinow and G. Hurlbert, Thresholds for families of multisets, with an application to graph pebbling, *Discrete Math.* **269** (2003), no. 1-3, 21–34.

[4] A. Bekmetjev and G. Hurlbert, The pebbling threshold of the square of cliques, *Discrete Math.*, **308** no. 19 (2008), 4306–4314.

[5] A. Blasiak, A. Czygrinow, A. Fu, D. Herscovici, G. Hurlbert and J. Schmitt, Sparse graphs with small pebbling number, *preprint.*

[6] A. Blasiak and J. Schmitt, Degree sum conditions in graph pebbling, *Australas. J. Combin.* **42** (2008), 83–90.

[7] P. Chiu, Cubic Ramanujan graphs, *Combinatorica* **12** (1992), no. 3, 275–285.

[8] F. R. K. Chung, Pebbling in hypercubes, *SIAM J. Discrete Math.* **2** (1989), no. 4, 467–472.

[9] T. Clarke, R. Hochberg and G. Hurlbert, Pebbling in diameter two graphs and products of paths, *J. Graph Th.* **25** (1997), no. 2, 119–128.

[10] C. Cusack, L. Dion and T. Lewis, The complexity of pebbling reachability and solvability in planar and outerplanar graphs, *Discrete Appl. Math.* **172** (2014), 62–74.

[11] C. Cusack, T. Lewis, D. Simpson and S. Taggart, The complexity of pebbling in diameter two graphs, *SIAM J. Discrete Math.* **26** (2012), 919–928.

[12] A. Czygrinow and G. Hurlbert, Girth, pebbling, and grid thresholds, *SIAM J. Discrete Math.*, **20** no. 1 (2006), 1–10.

[13] A. Czygrinow and G. Hurlbert, On the pebbling threshold of paths and the pebbling threshold spectrum, *Discrete Math.*, **308** no. 15 (2008), 3297–3307.

[14] A. Czygrinow, G. Hurlbert, H. Kierstead, and W.T. Trotter, A note on graph pebbling, *Graphs and Combinatorics* **18** (2002), 219–225.

[15] P. Dankelmann and L. Volkmann, Minimum size of a graph or digraph of given radius, *Inform. Process. Lett.* **109** (2009), no. 16, 971–973.

[16] D. Herscovici, Graham's pebbling conjecture on products of many cycles, *Discrete Math.* **308** (2008), no. 24, 6501–6512.

[17] D. Herscovici, B. Hester and G. Hurlbert, t-Pebbling and extensions, *Graphs and Combin.* **29** (2013), no. 4, 955–975.

[18] G. Hurlbert, A survey of graph pebbling, *Congr. Numer.* **139** (1999), 41–64.

[19] G. Hurlbert, Recent progress in graph pebbling, *Graph Theory Notes of New York* **XLIX** (2005), 25–37.

[20] G. Hurlbert, General graph pebbling, *Discrete Appl. Math.* **161** (2013), 1221–1231.

[21] G. Hurlbert, A Linear Optimization Technique for Graph Pebbling, *Preprints of the Centre de Recerca Matematica* **988** (2010).

[22] G. Hurlbert, Graph Pebbling, in *Handbook of Graph Theory* (2nd ed.), J. L. Gross, J. Yellen, and P. Zhang (eds.), CRC Press, Boca Raton, 2014.

[23] G. Hurlbert, *The Graph Pebbling Page*, `http://www.people.vcu.edu/~ghurlbert/pebbling/pebb.html`.

[24] G. Hurlbert and H. Kierstead, Graph pebbling complexity and fractional pebbling, *unpublished* (2005).

[25] K. Milans and B. Clark, The complexity of graph pebbling, *SIAM J. Discrete Math.* **20** (2006), no. 3, 769–798.

[26] L. Pachter, H. Snevily and B. Voxman, On pebbling graphs, *Congr. Numer.* **107** (1995), 65–80.

[27] N. Sieben, A graph pebbling algorithm on weighted graphs, *J. Graph Algorithms Appl.* **14** (2010), no. 2, 221–244.

[28] N. Watson, The complexity of pebbling and cover pebbling, `arXiv:math/0503511` (2005).