CMU-CS-85-101

White Pebbles Help

Robert Wilber

December 1984

Abstract

A family of directed acyclic graphs of vertex indegree 2 is constructed for which there are strategies of the black-white pebble game that use asymptotically fewer pebbles than the best strategies of the black pebble game. This shows that there are straight-line programs that can be evaluated nondeterministically with asymptotically less space than is required by any deterministic evaluation.

[redacted]

This research was sponsored by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 3597, monitored by the Air Force Avionics Laboratory under contract F33615-81-K-1539. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. government.

1. Introduction

The ***black-white pebble game*** is played by placing pebbles on, and removing them from, the vertices of a directed acyclic graph (henceforth called a dag). Pebbles are of two types, black and white, and are placed according to the following rules.

1. A black pebble may be placed on a vertex iff all of its immediate predecessors have pebbles (of either color).
2. A black pebble may be removed at any time.
3. A white pebble may be placed on a vertex at any time.
4. A white pebble may be removed iff all of its immediate predecessors have pebbles.

Note that rules 1 and 4 imply that a black pebble may always be placed on a vertex of indegree 0 and a white pebble may always be removed from such a vertex. The goal is to start with a pebble-free dag and carry out a sequence of pebbling moves s.t. every vertex is pebbled at some time and at the end there are no pebbles left on the dag. Such a sequence of moves is called a ***black-white strategy*** for the dag. The ***black pebble game*** is the restriction of the black-white pebble game in which there are no white pebbles (only rules 1 and 2 apply). A ***black strategy*** for a dag is a black-white strategy in which no white pebble moves are made. The number of pebbles used by a strategy is the maximum number of pebbles on the dag at once when the moves of the strategy are carried out. When we are considering pebbling strategies for an infinite class of dags we require that the indegrees of the vertices of the dags be uniformly bounded by some constant.

The black pebble game was introduced by Hewitt and Paterson [3]. It models the deterministic evaluation of straight-line programs. Vertices of a dag correspond to variables of a straight-line program and black pebbles correspond to registers containing deterministically computed results. The minimum number of pebbles used by any black strategy for a dag equals the minimum number of registers needed to deterministically evaluate the corresponding straight-line program.

The black-white pebble game was introduced by Cook and Sethi [2]. It models the nondeterministic evaluation of straight-line programs. White pebbles correspond to registers containing nondeterministic guesses which can be made at any time but which must be verified before they can be overwritten.

The black and black-white pebble games have been studied extensively; Pippenger [9] provides a comprehensive survey. The principle question remaining has been whether there is a class of dags of bounded indegree for which there are black-white strategies that use asymptotically fewer pebbles than the best black strategies. Meyer auf der Heide [8] proved that for pyramid graphs there are black-white strategies that use half as many pebbles as the optimal black strategies. He also showed that any dag that has a black-white strategy using n pebbles has a black strategy using $O(n^2)$ pebbles. Loui [7] and Meyer auf der Heide [8] independently proved that black-white strategies for balanced trees require at least half as many pebbles as the optimal black strategies. Lengauer and Tarjan [5] showed that this result applies to arbitrary trees. Klawe [4] proved a more general result, showing that for a large class of dags that "spread out" sufficiently rapidly, black-white strategies require at least half as many pebbles as the best black strategies. These "spreading graphs" include pyramid graphs and various generalizations of pyramid graphs.

The only previous result that shows any asymptotic difference between black and black-white pebble strategies is a space-time trade-off due to Lengauer and Tarjan [6]. Their result applies to a class of dags called bit reversal graphs. For black strategies on bit reversal graphs with N nodes they showed that the number of pebbles used, S, is related to the number of moves required, T, by the equation $T = \Theta(N^2/S)$. For black-white strategies the space-time trade-off is

$T = \Theta(N^2/S^2) + \Theta(N)$. So for a fixed time bound nondeterminism reduces the space required to pebble bit reversal graphs by a square root. However, if no bound is placed on the time any bit reversal graph has a black strategy using just 3 pebbles.

Here we show that there is a family of dags with vertex indegrees bounded by 2 for which there are black-white strategies that are better than the best black strategies by an arbitrarily large factor. The pth dag in the class has a black-white strategy using $O(p^2)$ pebbles while any black strategy requires $\omega(p^2)$ pebbles. Thus, nondeterministic computations can produce an asymptotic space savings compared to deterministic computations in the model of straight-line programs.

Throughout this paper we regard pebbling moves as being made at consecutive integer times. A pebble is placed on vertex v at time t if v has no pebble at time $t-1$ and has a pebble at time t. A pebble is removed from vertex v at time t if v has a pebble at time $t-1$ and has no pebble at time t.

2. Overview

The intuition as to why nondeterminism should save space in the model of straight-line programs is as follows. Suppose that for some straight-line program we have a deterministic evaluation strategy for which there is a time interval during which many intermediate values must be kept in registers while some hard to compute variable, say x, is computed. During that interval, in addition to the registers already in use, enough registers to compute x will have to be allocated, resulting in a pile-up of many registers in use at one time. A nondeterministic evaluation strategy can avoid this pile-up by guessing the value of x when it is first needed, continuing with the computation while retaining the guessed value in a register, and then verifying the guess (by computing the value of x) when sufficient space has been freed. In other words, a nondeterministic strategy can take out a loan when costs are high and pay it back when costs are low, thus reducing the worst case cost.

The proof is divided into two parts. In the first part a family of dags of vertex indegree 2 is defined. These dags can be "easily" black-white pebbled. In order to show that pebbles must pile-up during any black strategy it is necessary to be able to talk about the behavior of whole groups of pebbles at once. For this purpose a new type of pebble game, played on the cells of a matrix, is defined. For each dag in the family there is a corresponding matrix. Each pebble on the matrix represents many pebbles on the dag. In this way the matrix pebble game allows us to concentrate on the overall motion of groups of pebbles in a black strategy for the dag and to ignore unenlightening details of individual pebble moves.

Most of the first part of the proof consists of a number of lemmas that show that a black strategy for a dag can be translated into a matrix pebbling strategy in such a way that lower bounds on matrix pebbling strategies yield lower bounds on black strategies for the dags. In the second part of the proof a lower bound on matrix pebbling is proved that is sufficient to show that black strategies for the dags require asymptotically more pebbles than black-white strategies require.

3. The Matrix Pebble Game

Let M be a $p \times (p+1)$ matrix. The order p matrix pebble game is played on cells in the set $M^p = \{M_{ij} | 1 \le i \le p \text{ and } 1 \le j \le 2\lceil i/2 \rceil\}$, ie., the cells on and below the main diagonal together with the cell $M_{i,i+1}$ when i is odd. A cell M_{ij} is *above* M_{kl} iff $M_{ij} \in M^p$, $j = l$, and $i < k$. The *predecessor* of $M_{ij} \in M^p$ is $M_{i,j-1}$ if i is odd and $j > 1$ and is $M_{i,j+1}$ if i is even and $j < i$. If either i is odd and $j = 1$ or i is even and $j = i$ then M_{ij} has no predecessor.

The rules for the matrix pebble game are as follows.

1. A pebble may be placed on a cell $m \in M^p$ iff all of the cells above m have pebbles and the predecessor of m (if it has one) has a pebble. If m has no predecessor we only require that all of the cells above m have pebbles.
2. A pebble may be removed at any time.

The goal is to carry out a sequence of legal moves starting from a matrix with no pebbles on it such that every cell in M^p is pebbled at some time. Such a sequence of moves is called a *matrix strategy*. Equivalently, a matrix strategy is a legal sequence of moves starting with a pebble-free matrix that eventually places a pebble on $M_{p,p+1}$ when p is odd and on $M_{p,1}$ when p is even. It is easy to see that these cells can't be pebbled without first pebbling all other cells in M^p. Figure 1. shows a configuration of the matrix pebble game for $p = 6$ with x's in all positions that may be pebbled on the next move.

The *kth diagonal* of M^p is the set $\{M_{ij} \in M^p | i - j + 2 = k\}$. M^p has $p + 1$ diagonals. For $1 \leq k \leq p+1$ each cell of diagonal k is assigned the weight $w_k = 1/k^2$. The weight of a configuration of the matrix pebble game is the sum of the weights of all cells in M^p with pebbles on them. The weight of a matrix strategy is the maximum weight of any of the configurations produced by the strategy.

We construct a family of dags such that the pth dag has a black-white strategy using $O(n)$ pebbles, where $n = 2(p+1)^2$, and has a black strategy using q pebbles only if there is a matrix strategy for M^p with weight $O(q/n)$. In the next section we show that as p increases the weight of matrix strategies for M^p can not be bounded by any constant. Thus $q = \omega(n)$.

The dag G_p is constructed by connecting together a number of smaller dags called *blocks*. The blocks of G_p are B_{ij}, for $1 \leq i \leq p$ and $1 \leq j \leq 2\lceil i/2 \rceil$. The relations "above" and "predecessor" are defined for blocks as they are for cells in M^p. We say that two blocks are in the same row [column] when their first [second] indices are the same. The diagonals of G_p are defined as they are for M^p. Let $n = 2(p+1)^2$. The *height* of block B_{ij} in diagonal $k = i - j + 2$ is defined to be $\lceil n \cdot w_k \rceil$. Let h be the height of block B, let h_P be the height of its predecessor (0 if B has no predecessor), and let $h_1, \ldots, h_r$ be the heights of the blocks above B, in order from lowest numbered row to highest numbered row. Let $m = h_1 + \cdots + h_r$. The *length*, λ, of B is defined to be $h + h_P + m$. Block B with height h and length λ has vertices v_{ij}, where $1 \leq i \leq h$ and $1 \leq j \leq \lambda$. The edges of B are $(v_{ij}, v_{i,j+1})$, for $1 \leq i \leq h$ and $1 \leq j < \lambda$, and $(v_{ij}, v_{(i \bmod h)+1,j+1})$, for $1 \leq i \leq h$ and $\lambda - h + 1 \leq j < \lambda$. Figure 2. shows block $B_{2,1}$ for $p = 3$.

Row i of B is the set $\{v_{ij} | 1 \leq j \leq \lambda\}$, and *column* j is the set $\{v_{ij} | 1 \leq i \leq h\}$. The first h_P columns of B are called the *start section* of B, the next m columns are the *middle section*, and the last h columns are the *end section*. The vertices in column 1 are called the *input vertices* of B. The vertices in column λ are called the *output vertices* of B, and for convenience we denote the ith output vertex, $v_{i\lambda}$, by o_i. For $1 \leq i \leq m+1$, column $h_P + i$ is called the ith *level* of B. When $i \leq m$ level i is the ith column of the middle section, and level $m+1$ is the first column of the end section.

When blocks are connected together to form G_p any edge that leaves a block leaves an output vertex, and all edges that enter a block enter one of the vertices in the start or middle sections. A vertex u in G_p which is not in B is *connected* to column j of B if for each v in column j the edge (u, v) is in G_p. We say in that case that u is a *pre-vertex* of B. The edges entering each block B are as follows. If B has a predecessor, each of the h_P output vertices of the predecessor is connected to a distinct column in the start section of B. Let $B_1, B_2, \ldots, B_r$ be the blocks above B, in order from lowest numbered row to highest numbered row. Each of the h_1 output vertices of B_1 is connected to a distinct column of B among the first h_1 levels of B. Each of the h_2 output vertices of B_2 is connected to a distinct column of B among the next h_2 levels of B, and so on, ending with each of the h_r output vertices of B_r being connected to distinct columns among the last h_r levels of B's middle section. The blocks and the edges connecting them make up G_p. Every vertex has indegree 2 or less. Figure 3. shows a schematic representation of G_5, where blocks are represented by rectangles and the arrows within the rectangles point from lower numbered columns to higher numbered columns (the "direction of black pebbling"). The numbers within the rectangles are the heights of the blocks. Each edge in the figure leaving a block B_1 of height h_1 and entering a block B_2 represents the collection of h_1 connections from output vertices of B_1 to consecutive columns in the start or middle section of B_2. An edge entering the side of a block represents a collection of connections to columns in the start section of the block. Edges entering the top of a block represent collections of connections to columns in the middle section of the block, and are shown entering the block in the order that the connections are made to the columns.

LEMMA 1: *There is a black-white strategy for G_p that uses $O(n)$ pebbles.*

PROOF: Note that the sum of the heights of all blocks within a single column of G_p is less than $\sum_{i=1}^{p+1} \lceil nw_i \rceil < p + 1 + \sum_{i=1}^{p+1} n/i^2 < n\pi^2/6 + \sqrt{n/2} < 3n$. Also note that a block of height h with pebbles on all of its pre-vertices can be pebbled using either $h + 2$ black pebbles or $h + 2$ white pebbles. The idea is to pebble G_p one column at a time, pebbling in the "forward" direction on odd rows of blocks with black pebbles and in the "reverse" direction on even rows with white pebbles.

Columns of blocks are pebbled in sequence, from 1 to $2\lceil p/2 \rceil$. The invariant maintained is that before any blocks in column j are pebbled, $j > 1$, the output vertices of all blocks in column $j - 1$ have pebbles (black pebbles on even row blocks, white pebbles on odd row blocks). Column j is pebbled as follows.

1. White pebbles are placed on the output vertices of each even row block in column j.
2. If $j > 1$, all white pebbles in column $j - 1$ are removed by pebbling their respective blocks in order from highest numbered row to lowest numbered row. When white pebbles are removed in this order each even row block in column $j - 1$ has pebbles on all of its pre-vertices during the time it is pebbled.
3. The blocks in odd rows of column j are pebbled with black pebbles, in order from lowest numbered row to highest, leaving black pebbles on each output vertex of each odd row block. At each step the block being pebbled has all of its pre-vertices pebbled.
4. If $j > 1$ all black pebbles in column $j - 1$ are removed.

After these steps have been carried out the invariant for column $j + 1$ is achieved, and there are no pebbles in columns less than j. After pebbles have been placed on the output vertices of the last column steps 2 and 4 are used to first remove the white pebbles in the column and then the black pebbles. It is easy to verify that the procedure uses no more than $6n + 2$ pebbles. □

We now describe how to translate black strategies for G_p into matrix strategies for M^p.

DEFINITION: A block B of height h is *loaded* if there are h vertex-disjoint paths each starting at a distinct input vertex of B and ending at a distinct vertex of B with a pebble on it.

DEFINITION: A block B of height h is *full* if there are h vertex-disjoint paths each starting at a distinct input vertex of B and ending at a distinct vertex with a pebble on it in the middle or end section of B.

DEFINITION: A block is ***supported*** if it and all the blocks above it are full.

For each move of a black strategy for G_p we carry out 0 or more pebbling moves on M^p, according to the following rules.

1. For each $M_{ij} \in M^p$ s.t. M_{ij} does not already have a pebble and B_{ij} is supported, place a pebble on M_{ij}, proceeding in order from the lowest numbered row to the highest.
2. If $M_{ij} \in M^p$ has a pebble and B_{ij} is not full and its successor is not loaded then remove the pebble from M_{ij}. If M_{ij} has no successor remove its pebble if B_{ij} is not full.

We must show that if G_p and M^p are initially pebble-free then the above rules will translate black strategies for G_p using q pebbles into matrix strategies for M^p of weight $O(q/n)$. This is done in the remainder of this section. The next three lemmas establish some needed technical properties of black strategies for G_p. The reader may find it convenient at first to skip Lemma 2 and read only the statements of Lemmas 3 and 4. We assume that all pebbling strategies for G_p are black strategies in the rest of this section.

LEMMA 2: *Let B be a block with height h and m columns in its middle section. Let $T = [t_s, t_e]$ be a time interval. Let there be times $t_1, \ldots, t_h \in T$ s.t. at time t_i there is a pebble on the ith output vertex of B, o_i. Let $t_{min} = \min_{1 \le i \le h} t_i$. Then one of the following is true.*

1. *B is full throughout the interval $[t_s, t_{min}]$, or*
2. *There exist times $\tau_1, \ldots, \tau_{m+1} \in T$ with $\tau_1 < \cdots < \tau_{m+1}$ s.t. B is full throughout the interval $[\tau_1, \tau_{m+1}]$ and at time τ_i a pebble is placed on a vertex in the ith level of B.*

PROOF: Suppose (1) does not hold. Then at some time $t_a \in [t_s, t_{min}]$ there is a row with no pebbles on it within the middle or end sections, say row j. Call an output vertex *covered* if every path to that vertex from a vertex in level 1 contains a vertex with a pebble. At time t_a vertex o_j is not covered, whereas at time $t_j > t_a$ vertex o_j is covered. Let t_b be the latest time in the interval $[t_a, t_j]$ at which o_j is not covered. Let $\tau_1 = t_b + 1$. At time t_b there is a pebble-free path π ending at o_j and starting at some vertex u in level 1. Let k be the index of the row that contains u, and let y be the vertex in level $m+1$ and row k. Path π from u to o_j must include y. Let π' be the part of π that begins at y and ends at o_j. Path π' is pebble-free at time t_b but not at time t_j so there must be a first time τ_{m+1} in the interval $[\tau_1, t_j]$ when y is pebbled. Vertex u must be pebbled at time τ_1, otherwise π would still be pebble-free at τ_1. At time τ_1 row k cannot have any pebbles in levels 2 through $m+1$. Thus, since y is pebbled at time τ_{m+1} there must be times $\tau_2 < \cdots < \tau_m$ s.t. $\tau_1 < \tau_2$, $\tau_m < \tau_{m+1}$, and the vertex in row k and level i is pebbled at time τ_i.

We show that block B is full throughout the time interval $[\tau_1, \tau_{m+1}]$. Let $t \in [\tau_1, \tau_{m+1}]$. Path π' is pebble-free at time t (except there might be a pebble on y). Let set S be the last $h-1$ vertices of π'. Each vertex in S has indegree 2, and no vertex in S has a pebble at time t. We may construct $h-1$ vertex-disjoint paths, $\pi'_1, \ldots, \pi'_{h-1}$, from distinct vertices in level $m+1$ other than y to the vertices in S. This is done as follows. For $v \in S$, one of the edges entering v is not on π'. Extend a path backwards straight along the direction of this edge to a vertex in level $m+1$. (See Figure 4.) This is the same argument used by Cook [1]. Each path π'_i starting from a vertex e_i in level $m+1$, $e_i \ne y$, can be extended backwards along the row containing e_i to an input vertex, and can be extended forwards along π' to o_j, to yield a path π_i from an input vertex to o_j. Let $\pi_h = \pi$. At time t vertex o_j is covered, so there must be a pebble in the middle or end section on each of the paths π_i, for $1 \le i \le h$. For $1 \le i \le h$, let ρ_i be a portion of π_i starting at an input vertex and ending at a vertex of π_i in the middle or end section with a pebble. For $1 \le i < \ell \le h$ paths π_i and π_ℓ terminate at vertex $o_j \in S$ and the vertices they have in common are contained in S, all of whose vertices are pebble-free. So the ρ_i's must be disjoint, and, therefore, block B is full. □

LEMMA 3: *Let the first ℓ blocks of a column of G_p be $B_1, \ldots, B_\ell$, in order from lowest numbered row to highest. Let B_ℓ have height h and output vertices $o_1, \ldots, o_h$. Let $T = [t_s, t_e]$ be a time interval, and for $1 \le i \le h$ let $t_i \in T$ be a time at which there is a pebble on o_i. Let $t_{min} = \min_{1 \le i \le h} t_i$. Then one of the following is true.*

1. *B_ℓ is full throughout the interval $[t_s, t_{min}]$, or*
2. *There is a time $t \in T$ at which B_ℓ is supported.*

PROOF: (By induction on ℓ.) Assume the lemma is true for all positive $\ell' < \ell$. If B_ℓ is full throughout the interval $[t_s, t_{min}]$ then we are done. Otherwise, let m be the number of columns in the middle section of B_ℓ. For each positive $i < \ell$ let h_i be the height of B_i. Let $u_1 = 1$ and, for $2 \le i < \ell$, let $u_i = u_{i-1} + h_{i-1}$, so that u_i is the lowest index of those levels of B_ℓ which an output vertex of B_i is connected to. By Lemma 2 there are times $\tau_1, \ldots, \tau_{m+1} \in T$, where $\tau_1 < \cdots < \tau_{m+1}$, s.t. at time τ_i a pebble is placed on a vertex in level i of B_ℓ, and B_ℓ is full throughout the interval $[\tau_1, \tau_{m+1}]$. Let k be the largest positive integer less than ℓ s.t. there is a time in the interval $[\tau_1, \tau_{u_k}]$ when B_k is not full. (If there is no such k we are done — blocks $B_1, \ldots, B_\ell$ are all full at time τ_1.) At the times $\tau_{u_k}, \ldots, \tau_{u_{k+1}-1}$ pebbles are placed on the levels of B_ℓ to which the outputs of B_k are connected. So each output vertex of B_k must have a pebble on it at some time in the interval $[\tau_{u_k}, \tau_{u_{k+1}-1}]$. By the inductive hypothesis, using the substitutions $t_s \to \tau_1$, $t_e \to \tau_{u_{k+1}-1}$, and $t_i \to \tau_{u_k+i-1}$, there is a $t \in [\tau_1, \tau_{u_{k+1}-1}]$ at which B_k is supported, i.e., $B_1, \ldots, B_k$ are all full. Since $t \in [\tau_1, \tau_{u_i}]$, for $k < i < \ell$, we have that $B_{k+1}, \ldots, B_{\ell-1}$ are also full at time t. Finally, $t \in [\tau_1, \tau_{m+1}]$ so B_ℓ is full at time t. □

LEMMA 4: *Let B be a block of G_p with predecessor C. Let G_p have no pebbles at time 0. If at time $t_e > 0$ block B is full then there is a time $t_s \in [1, t_e]$ at which C is supported and s.t. for all times in the interval $[t_s, t_e]$ either C is full or B is loaded.*

PROOF: Let the height of B be h, and let s be the number of columns in its start section. At time t_e there are vertex-disjoint paths $\pi_1^1, \ldots, \pi_h^1$ leading from input vertices of B to vertices with pebbles in the middle or end section of B. Let u_i^1 be the last vertex of π_i^1. Let $t_1 = t_e$. For $i > 1$ we define t_i and, for $1 \le j \le h$, we define u_j^i and π_j^i, as follows. Time t_i is the latest time in the interval $[1, t_{i-1}]$ for which there is a k s.t. vertex u_k^{i-1} is not pebbled. There must be such a time because there are no pebbles on G_p at time 0. If u_k^{i-1} has no predecessor in B let $i_0 = i$ and leave t_r undefined for $r > i_0$ and u_j^r and π_j^r undefined for $r \ge i_0$. Otherwise, for $j \ne k$ vertex $u_j^i = u_j^{i-1}$ and path $\pi_j^i = \pi_j^{i-1}$, vertex u_k^i equals the predecessor of u_k^{i-1} on path π_k^{i-1}, and π_k^i equals π_k^{i-1} with vertex u_k^{i-1} removed. (Since for $1 \le j \le h$ one of the π_j^i's decreases in length each time i increases, there must be a finite value for i_0.) At time $t_i + 1$ a pebble is placed on u_k^{i-1} so if $i < i_0$ then at time t_i there must be a pebble on its predecessor, u_k^i. Thus for $1 \le i < i_0$ there are throughout the interval $[t_{i+1} + 1, t_i]$ disjoint paths $\pi_1^i, \ldots, \pi_h^i$ starting at input vertices and ending at vertices with pebbles, $u_1^i, \ldots, u_h^i$, respectively. So B is loaded throughout the interval $T = [t_{i_0} + 1, t_1]$.

Let $u_{k_0}^{i_0-1}$ be the vertex pebbled at time $t_{i_0} + 1$. It has no predecessors in B so it is an input vertex. Successive vertices of path $\pi_{k_0}^1$ are pebbled during the interval T, starting with $u_{k_0}^{i_0-1}$ in the first column of B and ending with $u_{k_0}^1$ in the middle or end section of B. Thus there are times $\tau_1, \ldots, \tau_s \in T$ s.t. at time τ_i the vertex of $\pi_{k_0}^1$ in column i of B is pebbled. At these times the various output vertices of C must have pebbles. Let t_a be the latest time in the interval $[0, t_{i_0}]$ s.t. C is not full. There is such a time since C is not full at time 0. By Lemma 3 there is a time $t_s \in [t_a, t_1]$ s.t. C is supported. If $t_s \le t_{i_0}$ then by definition of t_a block C is full throughout the interval $[t_s, t_{i_0}]$. Block B is loaded throughout $[t_{i_0} + 1, t_e]$, so either C is full or B is loaded throughout $[t_s, t_e]$. □

We can now easily prove that the previously given rules translate black strategies for G_p into matrix strategies for M^p in the desired way.

LEMMA 5: *If G_p and M^p are initially pebble-free then any legal sequence of black pebbling moves for G_p is translated into a legal sequence of pebbling moves for M^p by rules 1 and 2.*

PROOF: Pebbles can be removed at any time in the matrix pebble game, so we must show that all moves placing pebbles are legal. Rule 1 is the only rule that causes pebbles to be placed. Suppose a pebble placed at time t_e of the sequence of moves for G_p causes a pebble to be placed on cell M_{ij} in the matrix pebble game. Rule 1 ensures that all cells in M^p above M_{ij} have pebbles at the time M_{ij} is pebbled, because if a block is supported all the blocks above it are supported. We must show that the predecessor of M_{ij} (if there is one) also has a pebble at this time. Suppose M_{ij} has a predecessor. At time t_e block B_{ij} is full since it is supported. Let C be the predecessor of B_{ij}. By Lemma 4 there is a time $t_s \leq t_e$ s.t. C is supported at time t_s and throughout the interval $[t_s, t_e]$ either C is full or B_{ij} is loaded. By rule 1 a pebble is placed on the predecessor of M_{ij} at time t_s (if there wasn't one there already). By rule 2 this pebble can only be removed if C is not full and B_{ij} is not loaded, a condition which doesn't occur in the interval $[t_s, t_e]$. So at time t_e the predecessor of M_{ij} has a pebble. □

LEMMA 6: *Any black strategy for G_p is translated by rules 1 and 2 into a matrix strategy for M^p.*

PROOF: Lemma 5 shows that the matrix pebbling moves are legal, so it suffices to show that the most difficult cell to pebble of the matrix pebble game is pebbled. Let $m = M_{p,p+1}$ or $M_{p,1}$, according to whether p is odd or even, and let $B = B_{p,p+1}$ or $B_{p,1}$ be the corresponding block of G_p. Let t_f be the time of the final move of the black strategy for G_p. Then for each output vertex of B there is a time in $T = [0, t_f]$ when that vertex has a pebble. Since B is not full at time 0 we conclude by Lemma 3 that there is a time $t \in T$ when B is supported. At this time a pebble is placed on m by rule 1. □

LEMMA 7: *If a black strategy for G_p uses q pebbles then it is translated by rules 1 and 2 into a matrix strategy for M^p of weight no more than $5q/n$.*

PROOF: Lemma 6 shows that a black strategy for G_p is translated into a matrix strategy for M^p; we must show that the bound on the weight holds. By rule 2 a pebble is on M_{ij} only if either B_{ij} or its successor (if it has one) is loaded. (When a block is full it is also loaded.) When a block of height h is loaded it has at least h pebbles on its vertices. The weight of M_{ij} on diagonal $k = i - j + 2$ is w_k, the height of B_{ij} is at least nw_k, and the height of its successor is at least $\frac{1}{4}nw_k$. Thus the pebble on M_{ij} which contributes a weight of w_k can be associated with the at least $\frac{1}{4}nw_k$ pebbles on B_{ij} or its successor. For any i and j the pebbles on B_{ij} are associated with at most two pebbles on M^p — one on M_{ij} and one on M_{ij}'s predecessor. Also, if they are associated with two pebbles only one can contribute a weight of $4w_{i-j+2}$, the other contributes only w_{i-j+2}. So allowing for this double counting when a configuration of pebbles on M has weight w there are at least $\frac{1}{5}nw$ pebbles on G_p. The claim follows. □

4. A Lower Bound for the Matrix Pebble Game

In this section we prove that as p increases the weight of optimal matrix strategies for M^p must grow without bound. Roughly, we will show that there must be some time during any strategy for M^p at which there are a large number of pebbles on a single diagonal.

DEFINITION: The *channel* of M^p is the set $\{M_{ij} \in M^p | j \geq i \text{ or } (i \text{ odd and } j = i-1)\}$. Figure 5. shows M^6 and its channel.

LEMMA 8: *If at some time there are no pebbles in the channel of M^p between row i_1 and row $i_2 > i_1$ then before a pebble can be placed on a cell of the channel in row i_2 pebbles must be placed on every cell of the channel in rows i_1 through $i_2 - 1$.*

PROOF: This follows immediately from the rules for the matrix pebble game. □

LEMMA 9: *For any positive integer a, there is an integer p_a s.t. for all $p \geq p_a$ the minimum weight strategy for M^p has weight greater than a.*

PROOF: Let a be a positive integer. We shall assume that for all p there is a strategy for M^p of weight no greater than a and show that this leads to a contradiction. The outline of the proof is as follows. First, we use a to obtain, for each diagonal, a rough upper bound on how many pebbles may be on that diagonal during any strategy of weight a for any size matrix. We then tighten the bounds to make them as small as possible (the first three diagonals are treated specially). Finally, we show that there is a sufficiently large matrix for which there is no strategy that satisfies the tight bounds.

For $i \geq 1$ let $b_i = a/w_i$. Call a strategy for M^p a *b-strategy* if at no time are there more than b_i pebbles on diagonal i, for $1 \leq i \leq p+1$. By the assumption there is a b-strategy for M^p, for all p. For otherwise there would be an p' s.t. for any strategy for $M^{p'}$ there is some diagonal j which at some time has more than a/w_j pebbles, each contributing a weight of w_j, yielding a total greater than a.

Let s be the smallest integer s.t. for all p there is a b-strategy for M^p which never has more than s pebbles at a time on diagonals 1, 2, and 3 combined. Under the assumption, s must exist and is in fact no greater than $b_1 + b_2 + b_3$. Call a b-strategy which never has more than s pebbles at a time on diagonals 1, 2, and 3 combined an *s-strategy*. There must be a p_0 s.t. any b-strategy for M^{p_0} requires s pebbles on diagonals 1, 2, and 3 combined at some time, for otherwise s could be made smaller. If $p > p_0$ then any b-strategy for M^p that uses less than s pebbles in diagonals 1, 2, and 3 combined yields a b-strategy for M^{p_0} that uses less than s pebbles in those diagonals, because M^{p_0} is the subset of M^p consisting of its first p_0 rows. So for all $p \geq p_0$, any s-strategy for M^p must have a time when diagonals 1, 2, and 3 contain a total of s pebbles.

For $i \geq 4$ we define c_i in terms of the b_k's, s, and c_j for $j < i$, as follows. The integer c_i is the smallest integer s.t. for all p there is an s-strategy for M^p which uses no more than c_j pebbles on diagonal j at a time, for $4 \leq j \leq i$. Each c_i exists and is in fact no greater than b_i. A *c-strategy* for M^p is an s-strategy which never has more than c_i pebbles at a time on diagonal i, for $4 \leq i \leq p+1$. By construction, for all p there is a c-strategy for M^p. For each $i \geq 4$ there is a p'_i s.t. any c-strategy for $M^{p'_i}$ requires c_i pebbles on diagonal i at some time, for otherwise c_i could have been made smaller. By the argument of the previous paragraph, for all $p > p'_i$ any c-strategy for M^p requires c_i pebbles at some time on diagonal i. For $i \geq 4$, let $p_i = \max(p_0, \max_{4 \leq j \leq i} p'_j)$. Then for all $p \geq p_i$ any c-strategy for M^p has a time at which s pebbles are on diagonals 1, 2, and 3 combined and, for each $4 \leq j \leq i$, a time at which c_j pebbles are on diagonal j.

We need the following parameters.

$$m_1 = \max(4, 2\lceil p_0/2 \rceil) \qquad m_2 = \max(m_1 + 1, 2\lfloor p_{m_1+1}/2 \rfloor + 1)$$
$$\beta = c_4 + \cdots + c_{m_2+1} \qquad m_3 = 2 + m_1 + (\beta + 1)(m_1 + m_2 + 3)$$

Note that m_1 and m_3 are even and m_2 is odd. We have argued that for all p, M^p has a c-strategy, so in particular M^{m_3} has a c-strategy. We shall show that this is impossible, yielding the desired contradiction.

Refer to Figure 6. for the remainder of this proof. Let A be the set of cells of M^{m_3} in columns $m_3 - m_1 + 1$ through m_3. A is equivalent to M^{m_1}. Consider a c-strategy for M^{m_3}. Let t_1 be the first time at which cell $e_1 = M_{m_3, m_3 - m_1}$ is pebbled. At time t_1 the predecessor of e_1, cell $e_2 = M_{m_3, m_3 - m_1 + 1}$, must have a pebble. Let t_2 be the latest time before t_1 at which a pebble is placed on e_2. Let t_0 be the latest time before t_2 at which A contains no pebbles, so that A contains at least one pebble throughout the interval $[t_0 + 1, t_1]$. Cell e_2 is the most difficult cell in A to pebble, so during the interval $[t_0, t_2]$ the strategy for M^{m_3} must include a strategy for A. Because $m_1 \geq p_0$, there is a time $t_A \in [t_0 + 1, t_2]$ at which there are s pebbles in A on diagonals 1, 2, and 3 combined. At time t_A there are at most β pebbles on the first $m_2 + 1$ diagonals in rows 1 through $m_3 - m_1 - 2$. Thus, if we consider the row intervals $[i(m_1 + m_2 + 3) + 1, (i + 1)(m_1 + m_2 + 3)]$, for $0 \leq i \leq \beta$, all of which are included in rows 1 through $m_3 - m_1 - 2$, there must be at least one interval, say the i_0th, which has no pebbles in the first $m_2 + 1$ diagonals. Let $k = i_0(m_1 + m_2 + 3) + 1$ be the first row in this interval. Let $B = \{M_{ij} \in M^{m_3} | i \leq k + m_1 \text{ and } j \geq k\}$ and let $C = \{M_{ij} \in M^{m_3} | i \leq k + m_1 + m_2 + 1 \text{ and } j \geq k + m_1 + 2\}$. Then B is equivalent to M^{m_1+1} and C is equivalent to M^{m_2} and since both are contained in rows k through $k + m_1 + m_2$ and diagonals 1 through $m_2 + 1$ they are pebble-free at time t_A.

We distinguish four more cells of M^{m_3}.

$$e_3 = M_{m_3 - m_1 - 1, m_3 - m_1} \qquad e_4 = M_{k+m_1+m_2+1, k+m_1+m_2+2}$$
$$e_5 = M_{k+m_1+2, k+m_1+2} \qquad e_6 = M_{k+m_1, k+m_1+1}$$

At time t_A cell e_3 does not have a pebble because it is on diagonal 1 and all pebbles on the first three diagonals are in A at time t_A. At time t_1 cell e_3 must have a pebble because it is above e_1 and a pebble is placed on e_1 at time t_1. So let t_3 be the earliest time in $[t_A + 1, t_1 - 1]$ that a pebble is placed on e_3. At time t_A the channel of M^{m_3} between rows $k + m_1 + m_2 + 1$ and $m_3 - m_1 - 1$ is pebble-free, because all pebbles in the first three diagonals are in A. So by Lemma 8 there is a time $t_4 \in [t_A, t_3]$ at which a pebble is placed on e_4. Since C is pebble-free at time t_A and e_4 is the most difficult cell of C to pebble, the pebbling strategy for M^{m_3} must include a strategy for C during the interval $[t_A, t_4]$. The first cell of C to be pebbled must be e_5, so let t_5 be the earliest time in $[t_A, t_4]$ at which e_5 is pebbled. (Thus the pebbling strategy for C occurs during $[t_5, t_4]$.) The channel of M^{m_3} between rows $k + m_1$ and $k + m_1 + 2$ is pebble-free at time t_A so by Lemma 8 there is a time $t_6 \in [t_A, t_5 - 1]$ at which a pebble is placed on e_6. Cell e_6 is the most difficult cell of B to pebble so since B is pebble-free at time t_A the pebbling strategy for M^{m_3} must include a strategy for B during the interval $[t_A, t_6]$.

Because $m_1 + 1 > p_0$, there is a time $t_B \in [t_A, t_6]$ when all s pebbles on the first three diagonals are in B. For as long as there are no pebbles in the channel of A no pebbles can be placed in A, because every cell not in the channel has a channel cell above it. No pebbles can be placed in the channel of A until e_3 is pebbled, so throughout the interval $[t_B, t_3]$ no pebbles can be placed in A, and there are no pebbles in the first three diagonals of A. Although some pebbles in A may be removed during this interval they can not all be removed because $[t_B, t_3] \subseteq [t_0 + 1, t_1]$, and there is at least one pebble in A throughout the latter interval. Thus there is at least one pebble in A on a diagonal $j \in [4, m_1 + 1]$ which remains in place throughout the interval $[t_B, t_3]$. The strategy for M^{m_3} includes a strategy for C during the interval $[t_5, t_4]$, which is contained in $[t_B, t_3]$. Because $m_2 \geq p_{m_1+1}$ there is a time in $[t_5, t_4]$ at which there are c_j pebbles on diagonal j in C. Together with the fixed pebble on diagonal j in A this gives $c_j + 1$ pebbles on diagonal j of M^{m_3} at that time. This contradicts the claim that we are pebbling M^{m_3} with a c-strategy.

The assumption that for all p there is a strategy for M^p with weight no greater than a leads to a contradiction, so there is a p_a s.t. all pebble strategies for M^{p_a} have weight greater than a. If $p > p_a$ then M^{p_a} is contained in M^p so M^p does not have a strategy of weight a. □

THEOREM 1: *For $p \geq 1$ there is a black-white strategy for G_p which uses $O(n)$ pebbles, where $n = 2(p+1)^2$, whereas all black strategies for G_p require $\omega(n)$ pebbles.*

PROOF: The first part of the statement is Lemma 1, and the second part follows from Lemmas 7 and 9. ☐

5. Remarks

We have shown that the ratio of the minimum number of pebbles needed by any black strategy for G_p to the minimum number of pebbles needed by any black-white strategy for G_p must grow without bound as p increases. The proof does not give any hint as to how rapidly this ratio increases. This is because the critical step of setting the c_i's and p_i's in Lemma 9 is nonconstructive. In particular, it seems difficult to get an upper bound on the p_i's in terms of the parameter a.

One can show, using a divide and conquer argument, that M^p has a matrix strategy of weight $O(\log p)$. This strategy is easily translated into an $O(n \log n)$ black strategy for G_p (where as usual $n = 2(p+1)^2$). The matrix pebble game can be generalized in an obvious way to $p \times q$ matrices, with the weight of cell M_{ij} set to some nonnegative value w_{ij} in such a way that the sum of the weights in any column is bounded by a constant, for all p and q. (The lower triangular matrices used in this paper can be fit into this scheme by setting the weights of cells not in M^p to 0.) For a fixed p and q, call column $j \in [2, q-1]$ an *inflection column* if there is some row i s.t. either $w_{i,j-1} > w_{ij}$ and $w_{ij} < w_{i,j+1}$ or $w_{i,j-1} < w_{ij}$ and $w_{ij} > w_{i,j+1}$. Let γ be the number of inflection columns in the $p \times q$ matrix. The divide and conquer argument is easily extended to show that the $p \times q$ matrix has a strategy of weight $O(\log \gamma)$. In particular, since $\gamma < q$, there is a matrix strategy of weight $O(\log q)$. Thus if there is to be any chance of extending the method of this paper to show a large gap between optimal black-white and black pebble strategies, say an n versus $n^{1+\epsilon}$ gap, then very long matrices, with q large compared to n, must be used.

Acknowledgment. I thank Merrick Furst for checking the proof and improving the presentation.

REFERENCES

[1] Cook, S. A., *An observation on time-storage trade-off.* J. Comput. Syst. Sci., Vol. 9, pp. 308–316, 1974.

[2] Cook, S. A., and Sethi, R., *Storage requirements for deterministic polynomial finite recognizable languages.* J. Comput. Syst. Sci., Vol. 13, pp. 25–37, 1976.

[3] Hewitt, C. E., and Paterson, M. S., *Comparative schematology.* Project MAC Conf. on Concurrent Systems and Parallel Computation, Woods Hole, Mass., pp. 119–127, 1970.

[4] Klawe, M. M., *A tight bound for black and white pebbles on the pyramid.* FOCS, 24, pp. 410–419, 1983.

[5] Lengauer, T., and Tarjan, R., *The space complexity of pebble games on trees.* IPL, Vol. 10, pp. 184–188, 1980.

[6] Lengauer, T., and Tarjan, R., *Asymptotically tight bounds on time-space trade-offs in a pebble game.* J. ACM, Vol. 29, No. 4, pp. 1087–1130, Oct. 1982.

[7] Loui, M. C., *The space complexity of two pebble games on trees.* TM-133, Laboratory for Computer Science, Massachusetts Institute of Technology, 1979.

[8] Meyer auf der Heide, F., *A comparison of two variations of a pebble game on graphs.* TCS, Vol. 13, pp. 315–322, 1981.

[9] Pippenger, N., *Pebbling.* IBM Research Report RC 8258, 1980.

Figure 1. A configuration of M^6. Legal moves are marked with "x".

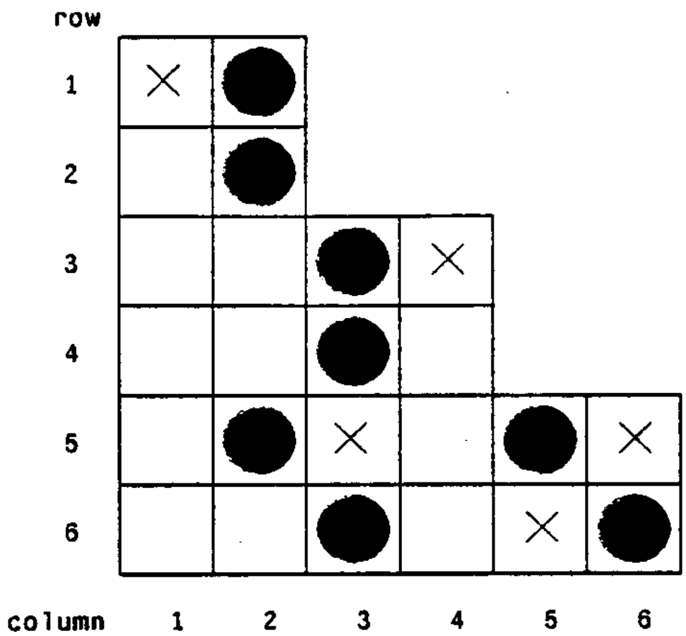

Figure 2. The block $B_{2,1}$ when $p = 3$.

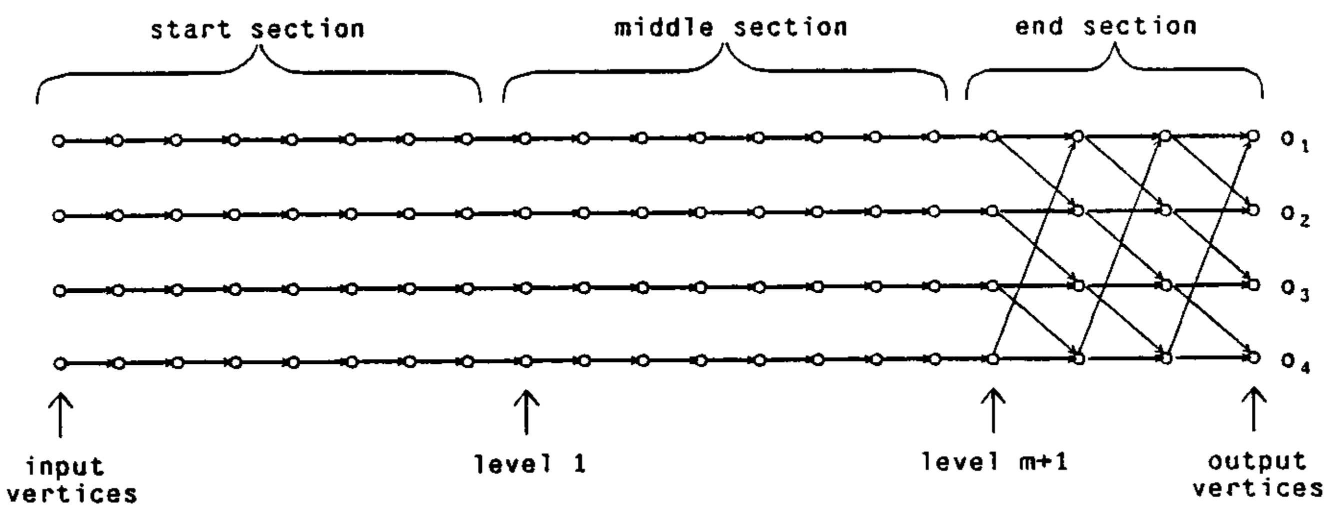

Figure 3. Schematic representation of G_5.

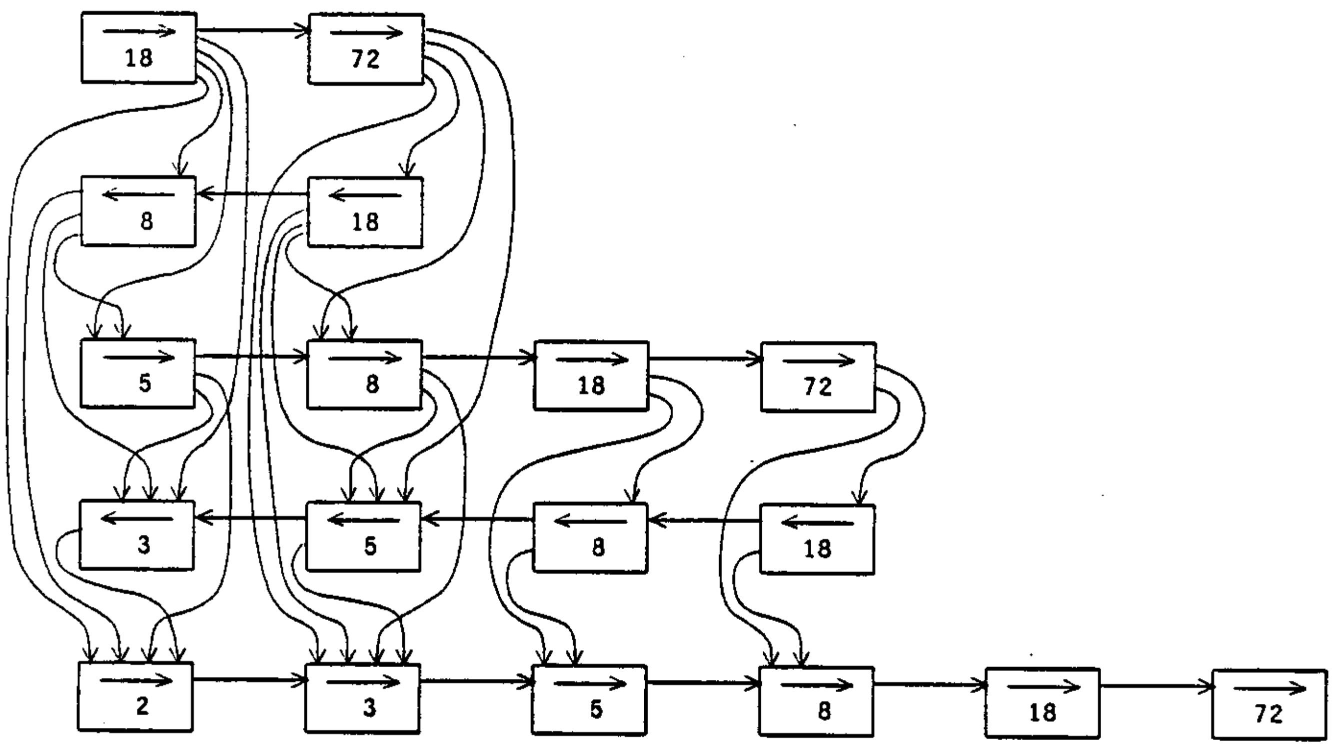

Figure 4. Construction of $\pi'_1, \ldots, \pi'_{h-1}$.

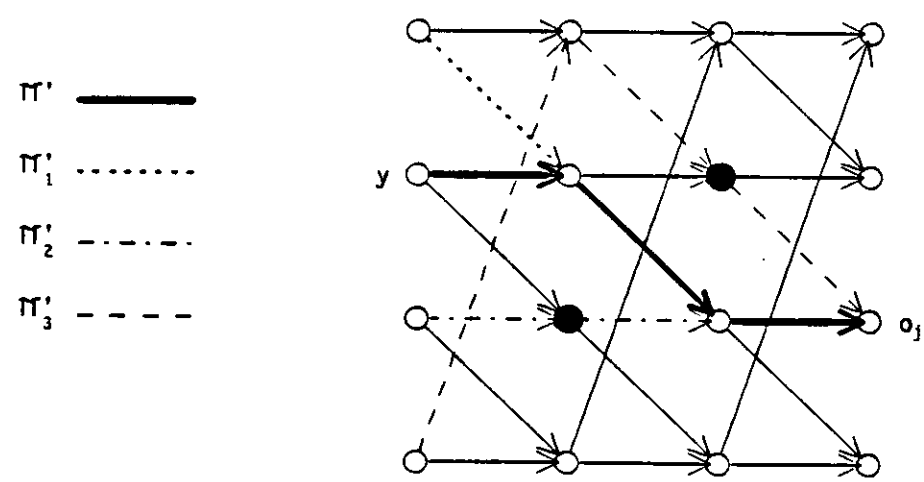

Figure 5. M^{6} with its channel shaded.

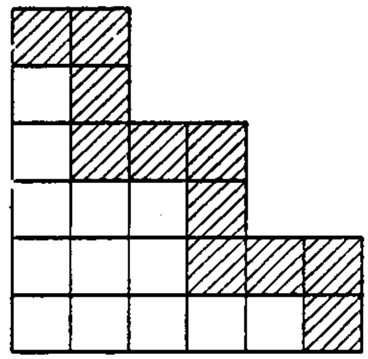

Figure 6. M^{m}_{3} at time t_A. Shaded areas are pebble-free.