Vertex Isoperimetric Parameter of a Computation Graph

Desh Ranjan and Mohammad Zubair
Department of Computer Science, Old Dominion University
Norfolk, VA 23519
dranjan@cs.odu.edu

Received (Day Month Year)
Accepted (Day Month Year)
Communicated by (xxxxxxxxxx)

Let $G = (V, E)$ be a computation graph, which is a directed graph representing a straight line computation and $S \subseteq V$. We say a vertex $v$ is an input vertex for $S$ if there is an edge $(v, u)$ such that $v \not\in S$ and $u \in S$. We say a vertex $u$ is an output vertex for $S$ if there is an edge $(u, v)$ such that $u \in S$ and $v \not\in S$. A vertex is called a boundary vertex for a set $S$ if it is either an input vertex or an output vertex for $S$. We consider the problem of determining the minimum value of boundary size of $S$ over all sets of size $M$ in an infinite directed grid. This problem is related to the vertex isoperimetric parameter of a graph, and is motivated by the need for deriving a lower bound for memory traffic for a computation graph representing a financial application. We first extend the notion of vertex isoperimetric parameter for undirected graphs to computation graphs, and then provide a complete solution for this problem for all $M$. In particular, we show that a set $S$ of size $M = 3k^2 + 3k + 1$ vertices of an infinite directed grid, the boundary size must be at least $6k + 3$, and this is obtained when the vertices in $S$ are arranged in a regular hexagonal shape with side $k + 1$.

Keywords: 

1. Introduction

In this paper we solve a problem of the type: given a positive integer $M$ find the minimum boundary that enclose area $M$. More specifically, we solve the problem of determining the vertex isoperimetric parameter of a computation graph, an infinite grid under certain constraints to minimize the boundary. Discrete isoperimetric inequalities for undirected graphs have been of interest to research community as they find applications in many areas including proving lower bounds [3, 22, 19]. The constraints and the particular notion of the boundary used here are originally motivated by the need to develop techniques to prove lower bounds for memory traffic for straight line computations [18].

1.1. Motivation

A straight line computation can be modelled using a DAG where the vertices represent computations and the edges incoming to a vertex represent the inputs required
for computation at this vertex. The result of a computation at a vertex is a single
value that can be used by multiple vertices in the computation. This is represented
by the multiple edges going out of a vertex. A DAG captures the data dependency
at various stages of the computation. A number of important problems such as
matrix multiplication, FFT, and several financial computations are straight line
computations [18], see Figure 1 and Figure ??.

The vertices of a DAG with in-degree zero are called the input vertices and the
vertices with out-degree zero are called the output vertices. The goal is to compute
the values at the output vertices given the values at the input vertices on an ar-
chitecture with a two-level memory hierarchy. Given a DAG, the computation for
DAG can be carried out in many different ways, essentially determined by the or-
der in which computations corresponding to various vertices on the DAG are done.
There can be multiple valid orderings as at any time in the DAG computation more
than one vertex might be available for next computation. The fast memory in the
memory hierarchy has limited capacity and for many large computations is not big
enough to hold intermediate results during the computation. This forces the archi-
tecture to use slower memory for storing intermediate results. Different orderings
results in different memory traffic to the slower memory [12]. The key in develop-
ing a high performance algorithm for a DAG is to identify the order that results
in minimum memory traffic to the slower memory. To evaluate the effectiveness of
different orderings and also to gain insight into the structure of the DAG and its
relationship to the memory traffic, it is desirable to find minimum possible memory
traffic to slower memory for any ordering.

One approach to analyze memory traffic on advanced architectures with memory
hierarchy is the use of red-blue pebbling game on a DAG[13, 17, 18]. A red pebble on
a vertex of the graph indicates that the corresponding value is in the fast memory.
The number of red-pebbles is determined by the size of the fast memory. One of the
rules in the red-blue pebble game is that a red pebble can be placed on vertex $v$ only
when all immediate predecessors of $v$ have red pebbles on them. This corresponds

Fig. 1. The binomial graph $G_b(n)$ with depth $n$ and $n + 1 = 8$ leaves.
to the notion that in a computation a value can be computed only if all the inputs needed to compute this value are available in the fast memory. As there is a limited supply of red pebbles, we can run out of red pebbles during the computation. In that case, we need to store intermediate outputs in the slow memory. This is modelled by placing a blue pebble on the red pebble. The input from the slow memory is modelled by placing a red pebble on blue pebble. The number of blue pebbles representing the storage in the slow memory is unlimited. A pebbling strategy for a DAG $G$ is a sequence of placement of pebbles on the nodes of $G$ satisfying the rules of the red-blue pebbling game that pebbles all vertices. We are interested in pebbling strategies that compute each vertex value exactly once. Among such strategies, we are interested in strategies that require minimum possible I/O to the slow memory. Simultaneously, we are also interested in deriving lower bounds on the number of I/Os required by any pebbling scheme for pebbling a given DAG $G$.

We have recently devised a novel approach called Boundary Flow Technique to derive these lower bounds. A formal description of the approach is given in [16]. Here we provide a brief overview of the approach to show how the notion of vertex isoperimetric parameter (VIP) of a graph [14, 11] is important in this method for establishing the lower bound. Let $\mathcal{P} = p_1, p_2, \ldots, p_r$ be a pebbling scheme for a DAG $G$ that requires $T$ steps where $p_i$ denotes the $i^{th}$ individual pebbling step. Imagine splitting the sequence of pebbling steps into some $l$ parts of consecutive pebbling steps $\langle p_1, p_2, \ldots, p_{t_1} \rangle, \langle p_{t_1+1}, \ldots, p_{t_2} \rangle, \ldots, \langle p_{t_l} \rangle$ such that each part has an equal number of newly pebbled vertices (except maybe the last). We call these parts sub-pebblings. Let $S_j$ be the set of newly pebbled vertices in the $j^{th}$ sub-pebbling. Consider vertices in $S_j$, which have an edge to some vertex in $S_j$. All of these vertices are predecessors (hence inputs) to some vertex in $S_j$ which are all pebbled in the $j^{th}$ sub-pebbling. Hence, we need to have red or blue pebbles on these vertices at the start of the $j^{th}$ sub-pebbling. If the number of input vertices, $|in(S_j)|$, exceeds the supply of the red pebbles by at least $\sigma_0$, $|in(S_j)| - \sigma_0$ of the input vertices must have blue pebbles. These vertices with blue pebbles contribute at least $|in(S_j)| - \sigma_0$ to the input during the $j^{th}$ sub-pebbling. Similarly, at the end of $j^{th}$ sub-pebbling consider vertices in $S_j$ which are input to the vertices in $S_j$. All of these vertices are predecessors to vertices that will be pebbled in some future sub-pebbling. Thus, we need to have red or blue pebbles on these vertices to preserve the intermediate results. Again, if the number of output vertices, $|out(S_j)|$, exceeds $\sigma_0$, at least $|out(S_j)| - \sigma_0$ output vertices must have blue pebbles which must have been placed during the $j^{th}$ sub-pebbling. Hence the $j^{th}$ sub-pebbling contributes at least $|in(S_j)| + |out(S_j)| - 2\sigma_0$ I/Os. The total number of I/Os then is at least $\sum_{l=1}^{\infty} |in(S_l)| + |out(S_l)| - 2\sigma_0$. Note here that $|S_1| = |S_2| = \ldots = |S_{l-1}|$ (say $M$). This is at least $(l-1) \cdot (\zeta_o(G, M) - 2\sigma_0)$ where $\zeta_o(G, M)$ is the minimum value of $|in(S)| + |out(S)|$ over all sets $S$ of size $M$. To get the strongest lower bound one should choose $M$ appropriately. Observe that $\zeta_o(G, M)$ is very similar to vertex isoperimetric parameter of the graph [14]. However, there are two crucial differences - the standard notion of vertex isoperimetric parameter is defined for an
undirected graph. Here we extend this to the directed graphs. Moreover, there are several ways in which one can define the boundary for a given set of a graph. The particular notion of the boundary used here is driven by our needs for the lower bound technique but is also natural in the context of computation graphs.

In the next subsection, we present some background and previous work related to the VIP problem as well as our definition of the VIP for the directed graphs.

1.2. Related Work and Definitions

The notion of boundary of a set of vertices of a graph has been extensively studied especially in the context of expander graphs [14]. Isoperimetric parameter of a graph is a way of capturing the notion of the minimum boundary of subgraphs (of the graph) of a fixed size. Very naturally, one thinks of an edge crossing the boundary of a set $S$ of vertices in $G = (V, E)$ if the two endpoints of the edge are in $S$ and $V - S = \bar{S}$. One can then measure the boundary of the set $S$ in several different ways. Two natural ways of “measuring” the boundary of a set $S$ of vertices in a graph are: (i) to count the number of edges with the two endpoints in $S$ and $\bar{S}$ and (ii) to count the number of vertices that have an edge incident on them with two endpoints in $S$ and $\bar{S}$. Within these two one can further refine the notion of boundary. For example, the vertex boundary of a set $S$ can be defined in several natural ways - as the set of vertices in $S$ that have an edge incident to a vertex in $\bar{S}$, as the set of vertices in $\bar{S}$ that have edge incident to a vertex in $S$ or the union of these two. The definition of the boundary for a graph has an impact on the isoperimetric parameter of the graph and the actual sets (shapes) that minimizes the perimeter for a given area [20, 14]. Obviously, the right definition depends on the context and application [14]. Traditionally, two isoperimetric parameters for a graph $G = (V, E)$ are defined [14].

**Definition 1.** The edge isoperimetric parameter of an undirected graph $G = (V, E)$ is given by

$$\zeta_e(G, M) = \min_{S \subseteq V} \{|E(S, \bar{S})| : |S| = M\}$$

**Definition 2.** The vertex isoperimetric parameter of an undirected graph $G = (V, E)$ is given by

$$\zeta_v(G, M) = \min_{S \subseteq V} \{|\Gamma(S) \setminus S| : |S| = M\}$$

Here $\Gamma(S)$ is the neighborhood of $S$. This definition counts all the vertices in $\bar{S}$ that have an edge incident to some vertex in $S$ as boundary vertices.

Almost all research and previous literature on Vertex Isoperimetric Parameter and related topics is for undirected graphs. The related notion of Edge Boundary and Edge Isoperimetric Parameter seems to have been studied more extensively than the Vertex Isoperimetric Parameter [11, 3, 10, 5, 4, 21, 2] but also almost exclusively.
for undirected graphs. An alternate possible definition of $\zeta_v(G, M)$ is:

$$
\zeta_v(G, M) = \min_{S \subset V} \{|\Gamma(\bar{S}) \setminus S| : |S| = M\}
$$

For an undirected graph, this definition counts all the vertices in $S$ that have an edge incident to a vertex not in $S$ as boundary vertices. A more symmetric definition has been suggested [14] where

$$
\zeta_v(G, M) = \min_{S \subset V} \{|\Gamma(S) \setminus S) \cup (\Gamma(\bar{S}) \setminus \bar{S})| : |S| = M\}
$$

We extend the notion of VIP to directed graphs based on previous discussion. First we extend the notion of boundary in an undirected graph to a DAG. Once again, a number of possible definitions are possible. Our definition is based on our needs and motivations as discussed previously. However, this definition is also natural in context of computation graphs.

**Definition 3.** Let $G = (V, E)$ be a directed graph and $S \subset V$. Then

$$
\begin{align*}
\text{out}(S) &= \{u \in S \mid v \in \bar{S} \text{ and } (u, v) \in E\} \\
\text{in}(S) &= \{u \in \bar{S} \mid v \in S \text{ and } (u, v) \in E\} \\
\text{boundary}(S) &= \text{in}(S) \cup \text{out}(S).
\end{align*}
$$

Note that $\text{in}(S) \cap \text{out}(S) = \emptyset$ and consequently $|\text{boundary}(S)| = |\text{in}(S)| + |\text{out}(S)|$.

**Definition 4.** The vertex isoperimetric parameter for a directed graph $G = (V, E)$ is:

$$
\zeta_v(G, M) = \min_{S \subset V} \{|\text{boundary}(S)| : |S| = M\}
$$

While we do not use it in this paper, similar to undirected graphs, we define edge isoperimetric parameter for a directed graph as follows.

**Definition 5.** The edge isoperimetric parameter for a directed graph $G = (V, E)$ is:

$$
\zeta_e(G, M) = \min_{S \subset V} \{|\{(u, v) : u \in S, v \in \bar{S} \text{ or } u \in \bar{S}, v \in S\}| : |S| = M\}
$$

From above definitions it is straightforward to see $\zeta_e(G, M) \geq \zeta_v(G, M)$. In the rest of the paper as we only consider vertex isoperimetric parameter of a specific computation graph, we drop the subscript $v$ and $G$ from the notation and simply refer the vertex isoperimetric parameter as $\zeta(M)$.

### 1.3. Infinite Directed Grid

In this paper, we consider a special computation graph, an infinite directed grid. This is inspired by a large financial computing application[15, 7], where the number of vertices far exceeds the number of red pebbles. In an infinite grid, the vertices are arranged on a grid and the edges have directions from bottom to top and left
to right. We use $V_{\infty}$ to denote the vertices of the infinite directed grid, and $E_{\infty}$ to denote its edges. The problem addressed in this paper is to determine the $\zeta(M)$ for an infinite directed grid.

To determine $\zeta(M)$ we first determine the minimum boundary size possible for a set of size $M$ when the sets $S$ of size $M$ being considered satisfy the pebbling constraint. Working with sets satisfying pebbling constraint simplifies our analysis. We later show that for any $M$ there exists a set of size $M$ that satisfies the pebbling constraint and has the minimum boundary size thus completing the proof.

The pebbling constraint defined below arises naturally in research on computation graphs [16, 18]. Informally, the sets that satisfy the pebbling constraint are “compact” and devoid of “holes”. Additionally they satisfy a monotonicity property that will become apparent in later discussion.

**Definition 6.** A $S \subset V_{\infty}$ satisfies the pebbling constraint if for any two vertices $u$ and $v$ in $S$, all vertices on every directed path from $u$ to $v$ are in $S$.

We refer to sets $S$ that satisfy the pebbling constraint as valid sets. The sets of size $M$ with boundary size $\zeta(M)$ are referred to as optimal sets. Observe that above definition ensures that vertices in a valid $S$ are from contiguous rows and columns of the grid. From now on, whenever we refer to a set $S$ of an infinite directed grid, we imply a valid $S$ unless specified otherwise. Figure 1 illustrates two sets of 7 vertices on a grid with different boundary count. One of the sets is optimal.

**Specification of a valid set**

For a valid set $S \subset V_{\infty}$, all vertices in $S$ are contained on a subgrid of size $k \times m$ for some $k > 0$ and $m > 0$, where the rows of the subgrid are numbered 1 to $k$ from top to bottom, and the columns of the subgrid are numbered 1 to $m$ from left to right. We refer to $m$ as the width of $S$. For our discussion later, we find it convenient to specify a valid set $S$ by specifying start and end column number for each row 1 to $k$. Notice that we can do this since translation of $S$ does not change its boundary size. Hence, a set $S \subset V_{\infty}$ is identified by $A_S = \langle(s_1, e_1), (s_2, e_2), \ldots, (s_k, e_k)\rangle$, where $s_i$ and $e_i$ are the column numbers associated with leftmost and rightmost grid points respectively of row $i$. We say that the length of row $i$ in $A_S$ is $e_i - s_i + 1$. Throughout this paper, a reference to $A_S$ means we are talking about a valid set $S$. With this convention, the specification for the left set in Figure ?? is $\langle(1, 1), (1, 2), (1, 2), (1, 2)\rangle$ and the specification for the right set in Figure ?? is $\langle(1, 2), (1, 3), (2, 3)\rangle$. Observe that because of the pebbling constraint, for any two consecutive rows $i$ and $i + 1$ in $A_S$, we have $s_{i+1} \geq s_i$ and $e_{i+1} \geq e_i$. In other words $\langle s_1, s_2, \ldots, s_k \rangle$ and $\langle e_1, e_2, \ldots, e_k \rangle$ are monotone non-decreasing sequences. Observe that the width of the set is given by $m = \text{width}(A_S) = e_k - s_1 + 1$.

We refer to the top most row and the bottom most row in a set $A_S$ as extreme
row. Similarly, we refer to the rightmost vertex in the top extreme row and the leftmost vertex in the bottom extreme row as extreme vertices. An extreme row is called complete if the number of vertices in this row is exactly one less than the row adjacent to it. Otherwise we call it incomplete. An important observation to make is that removing the extreme vertices does not increase the boundary. For the extreme vertex $v$ in the top row, this is true because removing $v$ does not introduce any new input vertex and introduces at most one new output vertex. Since $v$ itself is an output vertex, the resulting boundary size is no more than the original. Similar argument applies to the bottom extreme vertex. Throughout our paper, when we talk about removing $t$ extreme vertices from a set $A_S$, we assume that it is done by removing one extreme vertex at a time with the process being repeated $t$ times. Note that once an extreme vertex is removed from a set it results in a new valid set, and removing the next extreme vertex may entail removing vertices from more than one row of the original set $A_S$.

**Vertex isoperimetric parameter for an infinite directed grid**

The problem of determining $\zeta(M)$ for an infinite directed grid under pebbling constraints is non-trivial. There has been considerable research in determining the isoperimetric parameters for undirected grid graphs [8, 1, 5]. There is also related research of arranging $M$-polyomino to get the minimum perimeter [9, 23, 6, 22, 20]. An $M$-polyomino is a collection of $M$ squares of equal size arranged with coincident sides. The polyomino arrangement problem is very related to that of finding the edge isoperimetric parameter of the grid. For example, the polyomino problem of placing $M$ unit squares side by side such that two squares share at least one edge and the arrangement results in a minimum perimeter is equivalent to determining the edge isoperimetric parameter and an optimal set $S$ of size $M$ for an undirected grid. However, none of the results reported for isoperimetric parameter problem or the polyomino arrangement problem are directly applicable to our problem. The difference arises because of how we define the boundary of a set $S$ for an infinite grid. The boundary defined for edge isoperimetric parameter as in Definition 1 results in an optimal set $S$ of size $M$ that has vertices arranged in a square shape [20, 1]. On the other hand, the boundary defined for vertex isoperimetric parameter as in Definition 2 results in an optimal set $S$ that has vertices arranged in a diamond shape [8]. As we show later, neither one of these is optimal in our scenario.

The rest of the paper is organized as follows. In the next section, we derive an expression for the boundary of a set $S$. In Section 3 we establish a result that reduces the space of sets we need to consider when looking for an optimal set of a particular size. In Section 4 we show that for every $M$ there is an optimal set of barrel type. The very restricted nature of barrel type shapes allows us to derive a lower bound, $\zeta(M)$, which we do in Section 5. In Section 6 we show that the pebbling constraint is unnecessary in the sense that for every $M$ there is a set of size $M$ that satisfies the pebbling constraint and has minimum boundary among all
sets of size \( M \). In Section 7 we show some interesting properties of \( \zeta(M) \) including a convexity-like property.

2. Boundary Size for sets of \( V_\infty \)

Given \( A_S \), we partition the set of rows \( \{1, 2, \ldots, k\} \) into \( p \) sets \( S_1, S_2, \ldots, S_p \), such that:

(i) \( \forall x, y \in \{1, 2, \ldots, k\} \) \( \forall 1 \leq i \leq p - 1 \), \( x \in S_i, y \in S_{i+1} \Rightarrow x < y \)

(ii) \( \forall x, y \in \{1, 2, \ldots, k\} \) \( \forall 1 \leq i \leq p \), \( x \in S_i, y \in S_i \Rightarrow s_x = s_y \)

(iii) \( \forall x, y \in \{1, 2, \ldots, k\} \) \( \forall 1 \leq i < j \leq p \), \( x \in S_i, y \in S_j \Rightarrow s_x \neq s_y \).

We refer to \( S_i \) as a left-aligned set. All rows in \( S_i \) have the same start, and we can think of this as a start position repeating \( |S_i| - 1 \) times (observe that it is not \( |S_i| \) times as we consider the first occurrence as "primary"). For \( A_S \), we define total start repeats, \( tsr(A_S) \), as \( \sum_{i=1}^{p} (|S_i| - 1) = k - p \). Observe that this partition is unique.

Similarly, we define another partitioning for the set of rows based on the end position of the row. For \( A_S \), we partition the set of rows \( \{1, 2, \ldots, k\} \) in to \( q \) sets \( E_1, E_2, \ldots, E_q \), such that:

(i) \( \forall x, y \in \{1, 2, \ldots, k\} \) \( \forall 1 \leq i \leq q - 1 \), \( x \in E_i, y \in E_{i+1} \Rightarrow x < y \)

(ii) \( \forall x, y \in \{1, 2, \ldots, k\} \) \( \forall 1 \leq i \leq q \), \( x \in E_i, y \in E_i \Rightarrow e_x = e_y \)

(iii) \( \forall x, y \in \{1, 2, \ldots, k\} \) \( \forall 1 \leq i < j \leq q \), \( x \in E_i, y \in E_j \Rightarrow e_x \neq e_y \).

We refer to \( E_i \) as a right-aligned set. All rows in \( E_i \) have the same end, and we can think of this as an end position repeating \( |E_i| - 1 \) times. For \( A_S \), we define total end repeats, \( ter(A_S) \), as \( \sum_{i=1}^{q} (|E_i| - 1) = k - q \). Observe that this partition is also unique. See Figure ? for illustration.

**Theorem 7.** Given \( A_S = (s_1, e_1), (s_2, e_2), \ldots, (s_k, e_k) \) such that \( m = e_k - s_1 + 1 \), the boundary size of \( S \) is given by \( 2m + tsr(A_S) + ter(A_S) + 1 \).

**Proof.** Consider \( A_S = (s_1, e_1), (s_2, e_2), \ldots, (s_k, e_k) \) of \( k \) rows that spans \( m \) columns, where \( m = e_k - s_1 + 1 \). Let \( S_1, S_2, \ldots, S_p \) be the left-aligned sets, and \( E_1, E_2, \ldots, E_q \) be the right-aligned sets of \( A_S \). We compute the boundary size of \( A_S \) by first counting the number of boundary vertices in each column of \( A_S \). The top most vertex of each column is an output vertex, and the bottom most vertex of each column has a distinct input vertex just below it. As none of these boundary vertices in a column are shared by other columns, we have a count of \( 2m \) boundary vertices for \( A_S \). We now look at the rows of \( A_S \) and find the boundary vertices that were not counted earlier. Observe that because of pebbling constraint for any two consecutive rows in \( A_S \), we have \( s_{i+1} \geq s_i \) and \( e_{i+1} \geq e_i \). With this constraint the only way we can miss boundary vertices while counting for columns is when there is a left-aligned set \( S_i \) with \( |S_i| > 1 \) or a right-aligned set \( E_i \) with \( |E_i| > 1 \) present in \( A_S \). Consider one such \( S_i \) in \( A_S \). The row above \( S_i \) is shifted to the left by one
or more steps and the row below is shifted to the right by one or more steps. Each row of \( S_l \) has an input vertex for the left-most vertex in the row, and all these input vertices except for the first row are not counted as part of the column counting. In other words we have to add \(|S_l| - 1\) for \( 1 \leq l \leq p \) to the boundary count of \( 2m \) resulting in a boundary size of \( 2m + \text{tsr}(A_S) \). Similarly, when an \( E_l \) with \(|E_l| > 1\) is present in \( A_S \), the right-most vertex of each row of \( E_l \) is an output vertex, and all these output vertices except for the top most row of \( E_l \) is not counted as part of the column counting. Hence, we need to further add \(|E_l| - 1\) for \( 1 \leq l \leq q \) to the boundary count of \( 2m + \text{tsr}(A_S) \) resulting in a boundary size of \( 2m + \text{tsr}(A_S) + \text{ter}(A_S) \).

At this point the only boundary vertex that has not been accounted for is the left input vertex for the left-most vertex of the first row in \( A_S \), resulting in a boundary of \( 2m + \text{tsr}(A_S) + \text{ter}(A_S) + 1 \). See Figure ?? for an example.

3. Reducing the Search Space for Optimal sets

In this section we establish a result that helps us in reducing the space of sets that we need to consider when looking for an optimal set of size \( M \).

**Lemma 8.** For any set \( A_S = (s_1, e_1), (s_2, e_2), \ldots, (s_k, e_k) \) where \(|S| = M\), there exists another set \( A_{S'} = (s'_1, e'_1), (s'_2, e'_2), \ldots, (s'_{k'}, e'_{k'}) \) such that

(i) \(|S| = |S'|

(ii) \(|\text{boundary}(A_{S'})| \leq |\text{boundary}(A_S)|

(iii) \text{width}(A_{S'}) is the length of the longest row in \( A_S \).

**Proof.** When the length of longest row in \( A_S \) equals \( \text{width}(A_S) \), we have nothing to prove. Otherwise let \( L \) be the row number of the longest row. If \( s_1 < s_L \) then let \( l \) be the number of rows in the left aligned set \( S_1 \). Observe that \( s_{l+1} > s_l \) and \( e_{l+1} = e_l \). We consider two cases.

**Case 1:** \( e_{l+1} > e_l \). Shift all rows in \( S_1 \) by one step to the right to obtain another valid set \( A_{S''} \) with \( \text{width}(A_{S''}) = \text{width}(A_S) - 1 \). Note that \( \text{tsr}(A_{S''}) \leq \text{tsr}(A_S) + 1 \) and \( \text{ter}(A_{S''}) \leq \text{ter}(A_S) + 1 \). From Theorem 7 it follows \(|\text{boundary}(A_{S''})| \leq |\text{boundary}(A_{S'})| \).

**Case 2:** \( e_{l+1} = e_l \). Let \( j \geq 1 \) be the number of rows below row \( l \) that have their ends equal to \( e_l \), that is \( e_{l+j} = e_{l+j-1} = \ldots = e_l \). Add to \( S \) the vertex immediately to the right of the end vertex for all \( j \) rows below row \( l \), and then shift all rows in \( S_1 \) one step to the right to obtain another valid set \( A_{S''} \). Note that \( \text{width}(A_{S''}) = \text{width}(A_S) - 1 \), \( \text{tsr}(A_{S''}) \leq \text{tsr}(A_S) + 1 \), and \( \text{ter}(A_{S''}) = \text{ter}(A_S) \). From Theorem 7 it follows \(|\text{boundary}(A_{S''})| \leq |\text{boundary}(A_S)| \). Note that for this case \(|S''| > |S| \).

We continue this process until \( s_1 = s_L \). We follow a similar process to move all rows below row \( L \) to make their end points same as \( e_L \). Let \( r \) be the total number of vertices added to the original set \( S \) while handling Case 2. We now remove \( r \) extreme vertices resulting in \( S' \), which is the desired set.
Lemma 8 characterizes the structure of an optimal set. Essentially the vertices of the set can be thought of as arranged in two groups of consecutive rows where the top group of rows has the same start and the bottom group of rows has the same end. Figure ?? illustrates such a set. Notice that in such a set the width is determined by a row with maximum number of vertices. Observe that there can be more than one row with maximum number of vertices. We denote this by $c_{\text{max}}(A_S)$. When $A_S$ is a repeat in set $A_S$ is a repeat in set $A_S = (s_1, e_1, \ldots, s_k, e_k)$ if $s_i = s_{i+1}$ and $e_i = e_{i+1}$. The total number of repeats in $A_S$, $\text{rep}(A_S)$, is defined to be $\{|\{j \mid \text{row } j \text{ is a repeat in } A_S\}\}$. We use $c_{\text{max}}(A_S)$ to refer to repeats in $A_S$ of row of size $c_{\text{max}}$.

**Theorem 9.** Consider a set $A_S = (s_1, e_1, \ldots, s_k, e_k)$ that satisfies the properties in Lemma 8. Then $|\text{boundary}(A_S)| = 2c_{\text{max}}(A_S) + k + \text{rep}(A_S)$.

**Proof.** Let $S_1, S_2, \ldots, S_p$ be the left-aligned sets, and $E_1, E_2, \ldots, E_q$ be the right-aligned sets of $A_S$. Observe that $|S_1| + |E_q| - 1$ equals the number of rows plus the number of times the row of width $c_{\text{max}}$ repeats. Also, $\sum_{l=2}^{p}(|S_l| - 1)$ is the number of row repeats strictly below the rows of width $c_{\text{max}}$, and $\sum_{l=1}^{q-1}(|E_l| - 1)$ is the number of row repeats strictly above the rows of width $c_{\text{max}}$. From Theorem 7,

$$
|\text{boundary}(A_S)| = 2m + \text{ter}(A_S) + tsr(A_S) + 1 \\
= 2c_{\text{max}}(A_S) + \sum_{l=1}^{p}(|S_l| - 1) + \sum_{l=1}^{q}(|E_l| - 1) + 1 \\
= 2c_{\text{max}}(A_S) + |S_1| + |E_q| - 1 + \sum_{l=2}^{p}(|S_l| - 1) + \sum_{l=1}^{q-1}(|E_l| - 1) \\
= 2c_{\text{max}}(A_S) + (k + c_{\text{max}}(A_S)) \\
+ \#\text{repeats of other rows} \\
= 2c_{\text{max}}(A_S) + k + \text{rep}(A_S).
$$

The above theorem can be verified for example $A_S$ shown in Figure ??(a). The value for $c_{\text{max}}(A_S)$ is 7, $k = 8$, and $\text{rep}(A_S) = 3$, which using Theorem 9 results in the correct boundary size $|\text{boundary}(A_S)| = 25$.

4. Barrel Type Optimal sets

We will establish that for any positive integer $M$ there exists a boundary optimal set $A_S$ with $|S| = M$ that satisfies the following properties:

(a) $A_S$ has no repetitions of any row.

(b) Consecutive rows of $A_S$ differ from each other in length by no more than one except possibly for the top most row and the row below it (or the bottom most row with the row above it).

(c) $A_S$ is “symmetric”, that is, the difference between between the number of rows above and below the row of width $c_{\text{max}}$ is at most one.
We show this by transforming any set $A_S$ with $|S| = M$ satisfying the properties in Lemma 8 into a set $A_{S'}$ with $|S'| = M$ such that $|\text{boundary}(A_{S'})| \leq |\text{boundary}(A_S)|$ and $A_{S'}$ satisfies the properties (a), (b), and (c) above. We take an example of $A_S$ with $|S| = 42$ to illustrate various transformations, see Figure ??(a).

(i) Removing repeats other than $c_{\text{max}}$: If a row of length $c_i < c_{\text{max}}$ is repeated, add a new row of size $c_{\text{max}}$ adjacent to other rows of size $c_{\text{max}}$. Delete the row of length $c_i$ and then remove $c_{\text{max}} - c_i$ vertices from the extreme. This set $A_{S'}$ has the same $c_{\text{max}}$ as $A_S$, no more repeats than $A_S$ and no more rows than $A_S$. Hence, $|\text{boundary}(A_{S'})| \leq |\text{boundary}(A_S)|$. By repeating this process, all repeats other than those of $c_{\text{max}}$ can be eliminated. See Figure ??(b) and Figure ??(c). Let $S^1$ denote the resulting set.

(ii) Decreasing repeats of $c_{\text{max}}$ to at most one: If $c_{\text{max}}$ is repeated more than once in $A_{S^1}$, say $t_{\text{max}} \geq 2$ times, add $t_{\text{max}} - 1$ vertices to the set, one each as the rightmost vertex of the row of length $c_{\text{max}}$ other than the top most and bottom most row of length $c_{\text{max}}$. Then shift every row below the bottom row of width $c_{\text{max}} + 1$ by one step to the right. This gives a set of $M + t_{\text{max}} - 1$ vertices. Now remove $t_{\text{max}} - 1$ vertices from the extreme. The resulting set $A_{S^2}$

(a) has width $c_{\text{max}} + 1$,
(b) has no more rows than $A_S$ and,
(c) has number of repeats that is at least two less than that in $A_S$ since the repeats of $c_{\text{max}}$ are decreased by two and removing extreme vertices does not increase repeats.

Hence, $|\text{boundary}(A_{S^2})| \leq |\text{boundary}(A_{S^1})|$. By repeating this process, the number of repeats of the maximum width row can be decreased to be no more than one. See Figure ??(d) and Figure ??(e).

(iii) Ensuring that number of vertices on consecutive rows differ by one and the shape is symmetric: We next transform $A_{S^2}$ to ensure that the number of vertices on consecutive rows differ by one and to make it symmetric without increasing its boundary. The new set is constructed in the following way. Create one or two rows of width $c_{\text{max}}$ as in the set $A_{S^2}$. Next add $c_{\text{max}} - 1$ vertices to create a left-aligned row above the top row of width $c_{\text{max}}$, and add another set of $c_{\text{max}} - 1$ vertices to create a right-aligned row below the bottom row of width $c_{\text{max}}$. Next add vertices to create rows of length $c_{\text{max}} - 2$ above and below the row of size $c_{\text{max}} - 1$. Continue in this fashion until the set has $M$ vertices. See Figure ??(f).

The resulting shape is symmetric with the possible exception that:

(A) there is an extra row above the row(s) of width $c_{\text{max}}$ which may additionally be incomplete.
(B) the number of rows below the row(s) of width $c_{\text{max}}$ is same as the number above but the extreme bottom row is incomplete whereas the
top one is complete.

Call this set \( A_{S^3} \). Then \( |\text{boundary}(A_{S^3})| \leq |\text{boundary}(A_{S^2})| \). This follows because \( c_{\text{max}} \) and the number of repeats in \( A_{S^3} \) is same as that in \( A_{S^2} \). Moreover the number of rows in \( A_{S^3} \) is no more than that in \( A_{S^2} \) as there are no row repetitions in \( A_{S^2} \) other than that of \( c_{\text{max}} \).

In our next transformation we talk about removing \( t \) extreme vertices from the set \( A_{S^3} \). What we mean by this is that we remove an extreme vertex from \( A_{S^3} \) and then remove \( t - 1 \) extreme vertices from the resulting set. At any time in the process the extreme vertex to be removed is selected as follows. If there is an incomplete row, the extreme vertex is selected from this row. Else if there is an extra row at the top, the extreme vertex from this row is selected. Else the extreme vertex from the bottom row is selected.

Similarly, we talk about adding \( t \) vertices to the set \( A_{S^3} \). At any time in the process a vertex is added as follows. If there is an incomplete row, the vertex is added to this row. Else if there is an extra row at the top, we start a new row at the bottom with this vertex. Else we start a new row at the top with this vertex. This way of adding and removing vertices maintains properties (i), (ii), and (iii). In fact, this way of removing and adding vertices in a set is used throughout the rest of the paper.

(iv) Decreasing repeats of \( c_{\text{max}} \) to zero In \( A_{S^3} \) there is at most one repeat of \( c_{\text{max}} \). If there is no repeat of \( c_{\text{max}} \), then we can choose \( A_{S'} \) to be \( A_{S^3} \). Otherwise, let \( r_t \) be the number of rows in \( A_{S^3} \) above and including the top \( c_{\text{max}} \), and \( l_b \) be the length of the bottom row. If \( r_t < l_b \), then we obtain \( A_{S'} \) by removing the end vertex from each of the \( r_t \) rows and adding \( r_t \) vertices to the bottom extreme row and below keeping the property that consecutive rows differ by one except possibly the extreme row. The fact that \( r_t < l_b \) ensures that no more than one row is added at the bottom. \( A_{S'} \) has no repeats, and it can be easily verified that \( |\text{boundary}(A_{S'})| \leq |\text{boundary}(A_{S^3})| \) (See Figure 2). Else consider the case when \( r_t \geq l_b \). Then \( A_{S'} \) is obtained by adding a vertex to the immediate left of the start vertex for each of the \( r_t \) rows at the top. Now delete \( r_t \) extreme vertices starting from the bottom keeping the shape symmetric. The fact that \( r_t \geq l_b \) ensures that at least one row is removed starting from the bottom. \( A_{S'} \) has no repeats, and \( |\text{boundary}(A_{S'})| \leq |\text{boundary}(A_{S^3})| \) since we increase the width by at most one, decrease the number of repeats by one, and the number of rows by at least one.

Consider sets \( A_S \) of \( M \) vertices of the following type: The set has a maximum width row of width \( c_{\text{max}} \). The rest of the set is simply obtained by filling the rows above and below by consecutively decreasing the lengths of the rows by 1 until there are \( M \) vertices in the set. Note that in order to ensure that \( A_S \) can “accommodate” \( M \) vertices one has to choose sufficiently large \( c_{\text{max}} \) (\( c_{\text{max}}^2 > M \)). We refer to such sets as barrel type sets with \( M \) vertices. See Figure 2 for an example barrel type
set. From discussion above, we can conclude that for any $M$ there is a boundary optimal set of barrel type. A barrel type set is completely specified by $(M, c_{\text{max}})$. To determine the boundary optimal set of barrel type with $M$ vertices, we need to determine the value of $c_{\text{max}}$ that minimizes the boundary size for given $M$.

5. A Lower Bound on the Boundary Size

In this section, we describe how to find value of $c_{\text{max}}$ for a given $M$ that minimizes the boundary size $\zeta(M)$. Obviously, this also allows us to calculate the minimum boundary size $\zeta(M)$. We start by two lemmas that characterize the minimum boundary barrel type sets for two specific values of $M$, namely $M = J_1(r) = (3r^2 + 1)/4$ for odd $r$, and $M = J_2(r) = 3r^2/4$ for even $r$.

Lemma 10. Let $r$ be any odd integer and let $M = J_1(r)$. Then a minimum boundary set $A_S$ of barrel type with $M$ vertices is obtained by choosing $c_{\text{max}} = r$.

Proof. Consider a barrel type set $A_S$ of $r$ rows with $c_{\text{max}} = r$. It is straightforward to see that the number of vertices in such a set is $M = J_1(r)$, and the length of the first and the last row is $l_t = l_b = (r+1)/2$. We now show that for any other set $A_S'$ of barrel type with $M$ vertices, if $c_{\text{max}} \neq r$ then $|\text{boundary}(A_S')| > |\text{boundary}(A_S)|$.

First consider the case when $c_{\text{max}} = r + \delta > r$, where $\delta > 0$. We construct the barrel type set for this new $c_{\text{max}}$ by first adding $\delta$ vertices at the end of each row in $A_S$ which maintains the barrel shape and then removing $\delta r$ extreme vertices (maintaining the barrel shape) to get the new set $A_S'$. Observe that the number of rows in $A_S'$ is strictly greater than $r - 2\delta$. This is true because each row in $A_S'$ has at least $(r+1)/2 + \delta$ vertices. Hence, by Theorem 7 we have $|\text{boundary}(A_S')| > |\text{boundary}(A_S)|$. Now consider the case when $c_{\text{max}} = r - \delta < r$, for $r > \delta > 0$. Then $(r - \delta)^2 \geq M$ because the maximum number of vertices in a barrel type set is $c_{\text{max}}^2$. Note that this also implies that $\delta < (r+1)/2$. We construct the barrel type set for the new $c_{\text{max}}$ by first removing $\delta$ vertices at the end of each row in $A_S$, and then add additional rows to accommodate $\delta r$ vertices to get the new set $A_S'$. Observe that the number of rows in $A_S'$ is strictly greater than $r + 2\delta$ as every row that is added has fewer than $(r+1)/2 - \delta$ vertices. Hence, by Theorem 7 we have $|\text{boundary}(A_S')| > |\text{boundary}(A_S)|$.  

Lemma 11. Let $r$ be any even integer and let $M = J_2(r)$. Then a minimum boundary set $A_S$ of barrel type with $M$ vertices is obtained by choosing $c_{\text{max}} = r$.

Proof. Similar to Lemma 10.

Theorem 12. Let $r$ be any odd integer and let $J_1(r) \leq M \leq J_1(r+2)$. Then there
exists a boundary optimal set of barrel type with $M$ vertices, where $(\zeta(M), c_{\text{max}}) =$

$$
\begin{align*}
(3r, r) & \quad \text{if } M = J_1(r) \\
(3r+1, r) & \quad \text{if } J_1(r) < M \leq J_1(r) + (r-1)/2 \\
(3r+2, r+1) & \quad \text{if } J_1(r) + (r-1)/2 < M \leq J_1(r) + r \\
(3r+3, r+1) & \quad \text{if } J_1(r) + r < M \leq J_1(r) + (3r+1)/2 = J_2(r+1) \\
(3r+4, r+1) & \quad \text{if } J_1(r) + (3r+1)/2 < M \leq J_1(r) + (2r+1) \\
(3r+5, r+2) & \quad \text{if } J_1(r) + (2r+1) < M \leq J_1(r) + (5r+3)/2 \\
(3r+6, r+2) & \quad \text{if } J_1(r) + (5r+3)/2 < M \leq J_1(r) + 3(r+1) = J_1(r+2)
\end{align*}
$$

**Proof.** Let $J_1(r) \leq M \leq J_1(r+2)$. If $M = J_1(r)$ or $M = J_1(r+2) = J_1(r) + 3(r+1)$ we established the desired result in Lemma 10. If $M = J_2(r+1) = J_1(r) + (3r+1)/2$, we established the desired result in Lemma 11. We split the range for remaining values of $M$ into two parts $J_1(r) < M < J_2(r+1)$ and $J_2(r+1) < M < J_1(r+2)$. When $J_1(r) < M < J_2(r+1)$ we claim that there exists a minimum boundary set of barrel type that has $c_{\text{max}} = r$ or $c_{\text{max}} = r + 1$. To prove this claim consider a minimum boundary set $A_S$ with $M$ vertices. If $c_{\text{max}}(A_S)$ is $r$ or $r + 1$, we are done. Otherwise first assume that $c_{\text{max}} = r - \delta$, $\delta > 0$. From $A_S$ we can obtain a barrel type set $A_{S'}$ with $J_1(r)$ vertices by deleting appropriate number of extreme vertices. Note that $c_{\text{max}}(A_{S'}) = r - \delta$. Let $A_{S''}$ be the minimum boundary barrel type set with $J_1(r)$ vertices. Then we know from Lemma 10 that $c_{\text{max}}(A_{S''}) = r$ and $|\text{boundary}(A_{S''})| < |\text{boundary}(A_S)|$. Since minimum row size in $A_{S''}$ is strictly greater than minimum row size in $A_S$, it is now obvious that the boundary obtained by extending $A_{S''}$ to a barrel type set with $M$ vertices is strictly less than the boundary size of $A_S$ which shows that $c_{\text{max}}$ cannot be less than $r$. Similarly, we can show that $c_{\text{max}}$ cannot be more than $r + 1$ which establishes the claim. A similar argument applies for the case $J_2(r+1) < M < J_1(r+2)$ and shows that $c_{\text{max}}$ for a minimum boundary barrel type set must be $r + 1$ or $r + 2$. Hence, for any given $M$, we now need to consider only two values of $c_{\text{max}}$. Simple arithmetic now establishes the correctness of the lemma. \qed

The above theorem completely characterize an optimal boundary set of barrel type for a given value of $M$. In particular, for $M = J_1(r) = (3r^2 + 1)/4$ for odd $r$ the vertices in the optimal set of barrel type are arranged in a regular hexagonal shape of size $k + 1 = (r + 1)/2$. We now give the optimal boundary set of barrel type in terms of $k$, which directly follows from the Theorem 12.

**Corollary 13.** Let $M$ be any integer and let $k$ be the largest integer such that $H(k) \leq M$, where $H(k) = 3k^2 + 3k + 1$. Then there exists a boundary optimal set
of barrel type with $M$ vertices, where $(\zeta(M), c_{\text{max}}) =$

\[
\begin{cases}
(6k + 3), 2k + 1 & \text{if } M = H(k) \\
(6k + 3) + 1, 2k + 1 & \text{if } H(k) < M \leq H(k) + k \\
(6k + 3) + 2, 2k + 2 & \text{if } H(k) + k < M \leq H(k) + 2k + 1 \\
(6k + 3) + 3, 2k + 2 & \text{if } H(k) + 2k + 1 < M \leq H(k) + 3k + 2 \\
(6k + 3) + 4, 2k + 2 & \text{if } H(k) + 3k + 2 < M \leq H(k) + 4k + 3 \\
(6k + 3) + 5, 2k + 3 & \text{if } H(k) + 4k + 3 < M \leq H(k) + 5k + 4 \\
(6k + 3) + 6, 2k + 3 & \text{if } H(k) + 5k + 4 < M \leq H(k) + 6k + 6 = H(k + 1)
\end{cases}
\]

For example, consider $M = 46$. The value of $k = 3$. This gives $\zeta(M) = 24$, $c_{\text{max}} = 8$. Hence, an optimal set of barrel type consists of one row of width $c_{\text{max}} = 8$ with four rows above it and three rows below it. Note that in this set, the extreme row at the top is incomplete and has only two vertices.

We now state two additional corollaries that will be useful in proving some interesting properties of $\zeta(M)$.

**Corollary 14.** Let $M$ be any integer and let $k$ be the largest integer such that $H(k) \leq M$. Then

\[
\zeta(M) = \begin{cases}
6k + 3 & \text{if } M = H(k) \\
6k + 3 + \left\lceil \frac{t+1}{k+1} \right\rceil & \text{if } M = H(k) + t, 0 < t < 6k + 6
\end{cases}
\]

**Corollary 15.** Let $M$ be any integer and let $k$ be the smallest integer such that $H(k) \geq M$. Then

\[
\zeta(M) = \begin{cases}
6k + 3 & \text{if } M = H(k) \\
6k + 3 - \left\lfloor \frac{t-1}{k} \right\rfloor & \text{if } M = H(k) - t, 0 < t < 6k
\end{cases}
\]

6. Some Interesting Properties for $\zeta(M)$

In this section, we show some interesting properties for the minimum boundary function $\zeta(M)$. First, we show that the function $\zeta(M)$ is monotonically non-decreasing, that is, if $M < M'$ then any optimal set with $M$ vertices has a boundary no more than any set with $M'$ vertices. We begin by proving this result.

**Lemma 16.** If $M \leq M'$ then $\zeta(M) \leq \zeta(M')$.

**Proof.** Consider a boundary optimal set $A_{S'}$ with $M + 1$ vertices. Delete the extreme vertex from $A_{S'}$ to get a set $A_S$ with $M$ vertices. We know by Theorem 9 that $|\text{boundary}(A_S)| \leq |\text{boundary}(A_{S'})|$. Let $A_{S''}$ be a boundary optimal set with $M$ vertices. Then $|\text{boundary}(A_{S''})| \leq |\text{boundary}(A_S)| \leq |\text{boundary}(A_{S'})|$. This completes the proof.
Next we show a convexity-like property for $\zeta(M)$ namely that $\zeta(M_1 + M_2) \leq \zeta(M_1) + \zeta(M_2)$, for $M_1, M_2 > 1$. For any $k \geq 0$, we define $H(k) = 3k^2 + 3k + 1$.

Let $H(k) = 3k^2 + 3k + 1$. Then for all $t \geq 1$

$$\zeta(H(k) + t) \leq \zeta(H(k)) + \left\lfloor \frac{t + 1}{k + 1} \right\rfloor$$

**Proof.** By induction on $t$. From Corollary 14 we know that for all $k \geq 0$

$$\zeta(H(k) + t) = \begin{cases} 
\zeta(H(k)) + \left\lfloor \frac{6(k + 1) + 1}{k + 1} \right\rfloor & \text{if } 1 < t < 6(k + 1) \\
\zeta(H(k + 1)) = \zeta(H(k)) + 6 & \text{if } t = 6(k + 1)
\end{cases}$$

Note that

$$\zeta(H(k)) + 6 < \zeta(H(k)) + \left\lfloor \frac{6(k + 1) + 1}{k + 1} \right\rfloor$$

If $t > 6(k + 1) + 1$ then $H(k) + t = H(k + 1) + t'$ where $t' = t - 6(k + 1)$. Notice that $1 < t' < t$. Hence, by induction,

$$\zeta(H(k) + t) = \zeta(H(k + 1) + t') \leq \zeta(H(k + 1)) + \left\lfloor \frac{t' + 1}{k + 1} \right\rfloor$$

$$\leq \zeta(H(k + 1)) + \left\lfloor \frac{t - 6(k + 1) + 1}{k + 1} \right\rfloor$$

$$= \zeta(H(k + 1)) - 6 + \left\lfloor \frac{t + 1}{k + 1} \right\rfloor$$

$$= \zeta(H(k)) + \left\lfloor \frac{t + 1}{k + 1} \right\rfloor + \left\lfloor \frac{6(k + 1) + 1}{k + 1} \right\rfloor - 6$$

$$= \zeta(H(k)) + \left\lfloor \frac{t + 1}{k + 1} \right\rfloor.$$  \hfill \Box

**Corollary 18.** Let $M > 0, t > 0$ be any integers. Let $k$ be the greatest integer such that $M \geq H(k)$. Then,

$$\zeta(M + t) \leq \zeta(M) + \left\lfloor \frac{t + 1}{k + 1} \right\rfloor.$$

**Proof.** Let $M = H(k) + t'$. If $t' = 0$, the result follows immediately from Corollary 17. Else $1 < t' < 6(k + 1)$. Then

$$\zeta(M + t) = \zeta(H(k) + t' + t)$$

$$\leq \zeta(H(k)) + \left\lfloor \frac{t' + t + 1}{k + 1} \right\rfloor \text{ from Corollary 17}$$

$$\leq \zeta(H(k)) + \left\lfloor \frac{t' + 1}{k + 1} \right\rfloor + \left\lfloor \frac{t}{k + 1} \right\rfloor$$

$$\leq \zeta(M) + \left\lfloor \frac{t + 1}{k + 1} \right\rfloor \text{ as the first two terms sum to } \zeta(M). \hfill \Box
Corollary 19. For all \( k \geq 1 \)
\[
\zeta(H(k)) \geq \left\lceil \frac{H(k)}{k} \right\rceil.
\]

**Proof.** \( \zeta(H(k)) = (6k+3) \). Hence \( k\zeta(k) = k(6k+3) = 6k^2 + 3k = H(k) + 3k^2 - 1 > H(k) \) for all \( k \geq 1 \). Hence \( \zeta(H(k)) > H(k)/k \) and since \( \zeta(H(k)) \) is an integer
\[
\zeta(H(k)) \geq \left\lceil \frac{H(k)}{k} \right\rceil.
\]

Corollary 20. Let \( u \geq 1 \) be any integer and let \( k \geq 0 \) be the largest integer such that \( u \geq H(k) \). Then,
\[
\zeta(u) \geq \left\lceil \frac{u+1}{k+1} \right\rceil.
\]

**Proof.** If \( k = 0 \) then \( 1 \leq u \leq 6 \). It is easily verified that for these \( u \), \( \zeta(u) \geq u + 1 \). Else \( k \geq 1 \). Let \( u = H(k) + t \) where \( 0 \leq t < 6k \). If \( t = 0 \), i.e. \( u = H(k) \), then by Corollary 19,
\[
\zeta(u) = \zeta(H(k)) \geq \left\lceil \frac{H(k)}{k} \right\rceil
\geq \left\lceil \frac{u}{k} \right\rceil \geq \left\lceil \frac{u+1}{k+1} \right\rceil
\text{ since } \forall k, u = H(k) > k.
\]

Otherwise \( u = H(k) + t \) where \( 1 < t < 6(k+1) \). Then
\[
\zeta(u) = \zeta(H(k)) + \left\lceil \frac{t+1}{k+1} \right\rceil
\geq \left\lceil \frac{H(k)}{k+1} \right\rceil + \left\lceil \frac{t+1}{k+1} \right\rceil
\geq \left\lceil \frac{H(k)+t+1}{k+1} \right\rceil \geq \left\lceil \frac{u+1}{k+1} \right\rceil.
\]

Corollary 21. For all \( M_1, M_2 > 1 \), \( \zeta(M_1 + M_2) \leq \zeta(M_1) + \zeta(M_2) \).

**Proof.** Without loss of generality, assume that \( M_1 \geq M_2 \). Let \( k_1 \) be the largest integer such that \( M_1 \geq H(k_1) \) and let \( k_2 \) be the largest integer such that \( M_2 \geq H(k_2) \). Then \( k_1 \geq k_2 \). It then follows that
\[
\zeta(M_1 + M_2) \leq \zeta(M_1) + \left\lceil \frac{M_2+1}{k_1+1} \right\rceil \text{ by Corollary 18}
\leq \zeta(M_1) + \left\lceil \frac{M_2+1}{k_2+1} \right\rceil \text{ since } k_2 \leq k_1
\leq \zeta(M_1) + \zeta(M_2) \text{ by Corollary 20.} \]
\]
7. Removing the Pebbling Constraint

In this section, we show that for every $M$ there is a set of size $M$ that satisfies the pebbling constraint and has the minimum boundary among all sets of size $M$. We do so by “transforming” a set $S$ of size $M$ that has minimum boundary size into a set that satisfies the pebbling constraint and has boundary size no more than $S$. Let $S$ be any set of vertices of the infinite grid such that $|S| = M$ and it has the minimum boundary. For now we assume the vertices in $S$ induce a subgraph for which the underlying undirected graph is connected. Later we relax this condition.

We will show how to transform $S$ into $S'$ so that $|S'| = |S|$, $|\text{boundary}(S')| \leq |\text{boundary}(S)|$ and $S'$ satisfies:

(a) All rows and columns of $S'$ are continuous and contiguous. That is vertices in $S'$ for a row are from contiguous column, and similarly vertices in $S'$ for a column are from contiguous rows. Hence, $S'$ can be expressed as $A_{S'} = \{(s_1, e_1), \ldots (s_k, e_k)\}$ for some $k$ and $s_i, e_i, i = 1 \ldots k$.

(b) For all rows $i, i+1$ in $A_{S'}$, $s_i \leq s_{i+1}$

(c) For all rows $i, i+1$ in $A_{S'}$, $e_i \leq e_{i+1}$.

$A_{S'}$ then satisfies the pebbling constraint and has optimum boundary. We first show the transformation needed to satisfy the first property. The insight for this transformation is best understood in non-discrete domain, where a a non-convex shape is transformed to a convex shape resulting in a lower boundary by ensuring that every line segment connecting two points of the shape is inside the shape. Suppose $S$ did not satisfy the constraint and it has vertices in column $j$ with non-contiguous rows. Let $i$ be the lowest row number such that the vertex $(i, j)$ is in $S$ but $(i+1, j)$ is not in $S$, and $i'$ be the lowest row number such that the vertex $(i', j)$ is in $S$ but $(i' - 1, j)$ is not in $S$. Consider the underlying undirected graph $G_S$ induced by vertices in $S$. Because of our assumption $G_S$ is connected. This implies there is an undirected path from $(i,j)$ to $(i',j)$. We now consider two cases.

Case 1: The shortest path $p$ from $(i,j)$ to $(i',j)$ is on the left side of column $j$. Observe that this path has at least $i' - i - 1$ output vertices. If we include the vertices bounded by the segment $(i + 1, j), (i' - 1, j)$ and $p$ in the set $S$, there will be no increase to the boundary size of $S$.

Case 2: The shortest path $p$ from $(i,j)$ to $(i',j)$ is on the right side of column $j$. Observe that this path has at least $i' - i - 1$ input vertices. If we included the vertices bounded by the segment $(i + 1, j), (i' - 1, j)$ and $p$ in the set $S$, there will be no increase to the boundary size of $S$.

For both cases, we can now repeatedly remove extreme vertices until we get $|S'| = M$. Note that the argument in Section 4 about removing extreme vertices applies to any set. Hence, this does not increase the boundary. We can make similar argument for a row of vertices in $S$ with non-contiguous columns. After this transformation, $S'$ satisfies (a) above. The transformations to satisfy (b) and (c) are very similar to
the transformations outlined in Lemma 8. These transformations involve shifting
a block of contiguous rows to the left or right, where the top row of the block
is the first row that violates one of these conditions, and the bottom row is the
last row of the set. To satisfy (b) consider the smallest $i$ such that $s_i > s_{i+1}$ and
$s_{i+1} - s_i = l > 0$. Shifting row $i + 1$ along with all the rows below it by $l$ steps
to the right does not increase the boundary and eliminates the violation of the
monotonicity constraint due to row $i + 1$. We repeat this process to eliminate all
such violations. Similarly we can ensure (c).

We now relax the connected constraints on $G_S$. In general, a set $S$ induces sev-
eral connected subgraphs. Let $S^{(i)}$ be one such set of $S$ that induces a connected
subgraph. We apply all the transformation mentioned above to each $S^{(i)}$ individ-
ually. We can then invoke the convexity-like property Corollary 21 to show that
minimum boundary size for $S$ occurs if we maintain $G_S$ as connected.

8. Conclusion

We defined the notion of vertex isoperimetric parameter and edge isoperimetric pa-
rameter for directed graphs. The definition of the boundary for vertex isoperimetric
parameter is motivated by computation graphs. We presented a complete solution
for the problem of determining the vertex isoperimetric parameter for the infinite
directed grid. In particular, we demonstrated that any set of $M = 3k^2 + 3k + 1$
vertices on the infinite directed grid, the boundary must be at least $6k + 3$, and this
is obtained by choosing the vertices in a regular hexagonal shape with side $k + 1$.
It will be interesting to count and enumerate the number of optimal solutions for
each $M$.

References

for the powers of the diamond graph. Discrete Mathematics, 308(11):2067 – 2074,
2008.
[5] Bela Bollobas and Imre Leader. Edge-isoperimetric inequalities in the grid. Combin-