Kohonen, Jukka; Rajamäki, Robin; Koivunen, Visa

**Planar additive bases for rectangles**

*Published in:*
JOURNAL OF INTEGER SEQUENCES

Published: 01/01/2018

*Document Version*
Publisher's PDF, also known as Version of record

*Please cite the original version:*
Planar Additive Bases for Rectangles

Jukka Kohonen
Department of Computer Science
University of Helsinki
P.O. Box 68, FI-00014 University of Helsinki
Finland
jukka.kohonen@helsinki.fi

Visa Koivunen and Robin Rajamäki
Department of Signal Processing and Acoustics
Aalto University
P.O. Box 15400, FI-00076 Aalto
Finland
visa.koivunen@aalto.fi, robin.rajamaki@aalto.fi

Abstract

We study a generalization of additive bases into a planar setting. A planar additive basis is a set of non-negative integer pairs whose vector sum set covers a given rectangle. Such bases find applications in active sensor arrays used in, for example, radar and medical imaging.

We propose two algorithms for finding the minimal bases of small rectangles: one in the unrestricted case where the basis elements can be anywhere in the rectangle, and another in the restricted case, where the elements are confined to the lower left quadrant. We present numerical results from such searches, including the minimal cardinalities and number of unique solutions for all rectangles up to $[0,11] \times [0,11]$ in the unrestricted case, and up to $[0,26] \times [0,26]$ in the restricted case. For squares we list the minimal basis cardinalities up to $[0,13] \times [0,13]$ in the unrestricted case, and up to $[0,46] \times [0,46]$ in the restricted case. Furthermore, we prove asymptotic upper and lower bounds on the minimal basis cardinality for large rectangles.
1 Introduction

An additive basis for an interval of integers \([0, n] = \{0, 1, 2, \ldots, n\}\) is a set of non-negative integers \(A\) such that \(A + A \supseteq [0, n]\). By extension we define that a planar additive basis for a rectangle of integers \(R = [0, s_x] \times [0, s_y]\) is a set of points with non-negative integer coordinates

\[
A = \{(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\}, \quad \text{such that } A + A \supseteq R.
\]

The sumset (or sum co-array) is defined in terms of vector addition, that is

\[
A + A' = \{(x + x', y + y') : (x, y) \in A, (x', y') \in A'\}.
\]

Additive bases for integer intervals have been widely studied since Rohrbach [22]. Often one seeks to maximize \(n\) when the basis cardinality \(|A| = k\) is given. For small \(k\) this has been approached with computations [1, 12, 17, 21], and for large \(k\) with asymptotic bounds [11, 25].

Less is known about planar additive bases. Kozick and Kassam discussed them in an application context of signal processing, and proposed some simple designs [13]. In a rather different line of work, sumsets in vector spaces and abelian groups have been studied with the interest in how small the sumset can be [2, 3, 4]. Boundary effects in planar sumsets have also been studied by e.g., Han [6].

We now aim to minimize the cardinality \(k\) of a planar additive basis, when the target rectangle \(R = [0, s_x] \times [0, s_y]\) is given. To the best of our knowledge, this combinatorial optimization problem has not been addressed before.

Planar bases have an application in signal processing, when an array of sensor elements is deployed on a plane to be used in active imaging or radar surveillance [19]. Here “active” means that the sensors both transmit a signal towards objects such as radar targets or human tissue, and receive the reflections. In this context, the pairwise vector sums of the sensor locations make up a virtual sensor array, called the sum co-array, which may be used to improve imaging resolution, or to reduce the number of sensors without significant performance loss [8].

An important special case is that of restricted bases. A basis \(A\) for \([0, n]\) is restricted if \(A \subseteq [0, n/2]\). Analogously we define that a basis \(A\) for \([0, s_x] \times [0, s_y]\) is restricted if \(A \subseteq [0, s_x/2] \times [0, s_y/2]\). Apart from practical motivations related to the physical placing of sensors, there are other reasons to study restricted bases. Computationally, we can study much larger instances in the restricted case, at least with our current algorithms. Numerically, restricted bases often attain the same minimum cardinality as unrestricted bases. Also, the minimal restricted bases often exhibit interesting geometric structure.

We introduce here the following results. First, a search algorithm is proposed for finding all bases of a given size for a given rectangle; and the minimum basis sizes are determined for all rectangles with \(s_x, s_y \leq 11\) or \(s_x = s_y \leq 13\). Secondly, a meet-in-the-middle method is developed that constructs a restricted planar basis by concatenating together four smaller
bases, one in each corner; and the minimum restricted basis sizes are determined for all even 
$s_x, s_y \leq 26$ or even $s_x = s_y \leq 46$. Thirdly, some asymptotic upper and lower bounds on the 
minimum basis size for large rectangles are established.

2 Definitions and preliminary observations

The target rectangle is $R = [0, s_x] \times [0, s_y]$. If $R$ is square, we call it the $s$-square, with 
$s = s_x = s_y$. A basis containing $k$ elements is a $k$-basis. The size of the smallest basis for 
$[0, s_x] \times [0, s_y]$ is denoted by $k(s_x, s_y)$.

If $s_x$ and $s_y$ are even, we set $h_x = s_x/2$ and $h_y = s_y/2$. Then a basis $A$ is restricted if 
$A \subseteq [0, h_x] \times [0, h_y]$. Note that it follows that $A + A = R$. The size of the smallest restricted 
basis is $k^*(s_x, s_y)$.

If $A$ is a basis for $R$ such that $A \subseteq R$, we say that $A$ is admissible. If not, then it 
cannot be minimal, since one can simply drop the elements that are outside the target. So we 
confine our attention to admissible bases.

We relate the basis size $k = |A|$ to the number of target elements $N = |R| = (s_x+1)(s_y+1)$, which may be understood as the target area measured in grid points. The efficiency of 
a basis is defined as

$$c = N/k^2.$$  (1)

The shape of the target is characterized by its aspect ratio $\rho = (s_y + 1)/(s_x + 1)$.

Two simple basis constructions were proposed by Kozick and Kassam in the context of 
sensor arrays [13]. For any rectangle, the L-shaped basis is

$$([0, s_x] \times \{0\}) \cup \{0\} \times [0, s_y],$$  (2)

which has $s_x + s_y + 1$ elements. If $s_x, s_y \geq 2$ are even, the boundary basis is

$$([0, h_x] \times \{0\}) \cup \{0\} \times [0, h_y],$$  (3)

which has $s_x + s_y$ elements and is restricted. These two provide a minimal basis for most small 
squares (boundary basis if $s \geq 2$ is even, L-shaped otherwise). The smallest counterexample is the 7-square, whose minimal bases have only 14 elements, one less than the L-shaped basis 
(see Figure 1c). However, for non-square rectangles, (2) and (3) are generally not minimal. 
Examples of this will be presented in Section 5, and an asymptotic result in Section 6.

The following observations about the corners and the horizontal edges of planar additive 
bases will be useful. Corresponding results in the vertical direction can be proven by 
transposing $x$ and $y$.

Lemma 1 (Origin corner). If $A$ is a basis for a rectangle with $s_x \geq 1$, then $(0, 0), (1, 0) \in A$.

Proof. The only way to represent $(1, 0)$ as a sum of two pairs of non-negative integers is 
$(0, 0) + (1, 0)$, so those elements must be in the basis.  \[\square\]
Figure 1: (a) The L-shaped basis for the 5-square. (b) The boundary basis for the 6-square. (c) A minimal basis for the 7-square. (d) A minimal basis for the 13-square.

Lemma 2 (Restricted edges). If $A$ is a restricted basis for $[0, s_x] \times [0, s_y]$, then its bottom edge $\{ x : (x, 0) \in A \}$ and top edge $\{ x : (x, h_y) \in A \}$ are (one-dimensional) restricted bases for $[0, s_x]$.

Proof. Consider first the bottom edge. Since the $y$ coordinates in $A$ are non-negative, for any $x \in [0, s_x]$ the point $(x, 0)$ must be the sum of some $(x', 0), (x'', 0) \in A$. Since $A$ is restricted, we have $x', x'' \leq h_x$.

Consider next the top edge. Since the $y$ coordinates in $A$ are at most $h_y$, for any $x \in [0, s_x]$ the point $(x, s_y)$ must be the sum of some $(x', h_y), (x'', h_y) \in A$. Since $A$ is restricted, we have $x', x'' \leq h_x$. \hfill $\square$

Lemma 3 (Two rows). For any even $s_x \geq 0$, we have $k^*(s_x, 2) = 2k^*(s_x, 0)$.

Proof. Let $A$ be a restricted basis for $[0, s_x] \times [0, 2]$. By Lemma 2 its bottom and top edges are restricted bases for $[0, s_x]$, so each has at least $k^*(s_x, 0)$ elements. Thus $|A| \geq 2k^*(s_x, 0)$.

To see that $k^*(s_x, 2) \leq 2k^*(s_x, 0)$, let $A^*$ be a restricted basis for $[0, s_x]$. Then $A^* \times [0, 1]$ is a restricted basis for $[0, s_x] \times [0, 2]$.

3 Search algorithm for admissible bases

In this section we develop a method to find all admissible $k$-bases for a given rectangle. Then we can also establish the minimum value of $k$. For example, the L-shaped basis suffices to show that $k(9, 9) \leq 19$, but to prove that $k(9, 9) = 19$ we must ascertain that there is no 18-basis for the 9-square. Trying out the $\binom{100}{18} \approx 3 \cdot 10^{19}$ ways of placing 18 elements in $[0, 9] \times [0, 9]$ is obviously impractical.

Our Algorithm 1 is a relatively straightforward generalization of Challis’s algorithm, which finds one-dimensional bases [1]. Assume for simplicity that $s_x \geq 2$. By Lemma 1 the points $(0, 0)$ and $(1, 0)$ must be included in the basis. Next we branch on the decision
Algorithm 1 Find all admissible $k$-bases for $[0, s_x] \times [0, s_y]$

1: procedure FindBases($k, s_x, s_y$)  
2:   \textbf{Extend}($k, s_x, s_y, \{(0, 0), (1, 0)\}, 1, 0$)  
3: \textbf{procedure} Extend($k, s_x, s_y, A, x, y$)  
4:   \triangleright (x, y)\ is\ the\ latest\ location\ considered\ (either\ filled\ or\ left\ empty)  
5:   $j \leftarrow |A|$ \quad \triangleright \ Number\ of\ elements  
6:   $G \leftarrow |[0, s_x] \times [0, s_y] \setminus (A + A)|$ \quad \triangleright \ Number\ of\ gaps  
7:   if \ ($j = k$) \wedge \ ($G = 0$) then \textbf{Print}($A$) \quad \triangleright \ Found\ a\ basis  
8:   if \ ($j = k$) then return \quad \triangleright \ Reached\ full\ size  
9:   $M \leftarrow (k + j + 1)(k - j)/2$ \quad \triangleright \ Max.\ sums\ to\ expect  
10:  if \ $M < G$ then return \quad \triangleright \ Too\ many\ gaps  
11: if \ $x < s_x$ then  
12:    $x \leftarrow x + 1$ \quad \triangleright \ Proceed\ right  
13: else if \ $y < s_y$ then  
14:    $x \leftarrow 0$ \quad \triangleright \ Begin\ next\ row  
15:    $y \leftarrow y + 1$  
16: else  
17:   return \quad \triangleright \ Reached\ the\ top\ right  
18: if \ $(x, y) \in A + A$ then \quad \triangleright \ Already\ covered?  
19:   \textbf{Extend}($k, s_x, s_y, A, x, y$) \quad \triangleright \ Branch\ without\ \((x, y)\)  
20:  \textbf{Extend}($k, s_x, s_y, A \cup \{(x, y)\}, x, y$) \quad \triangleright \ Branch\ with\ \((x, y)\)  

whether $(2, 0)$ is included. We proceed to the right and rowwise, branching at each location on whether that point is included, until we have $k$ elements or reach the top right corner.

During the search, two tests prune unfruitful branches. One of them (line 18) concerns unfillable holes in the sumset. Suppose that we are currently at $(x, y)$. Because of the way how the search proceeds, any location $(x', y')$ considered deeper in the search will have $x' > x$ or $y' > y$ (or both). Thus any such elements will not generate the sum $(x, y)$, by the non-negativity of coordinates. If $(x, y)$ has not already been covered, then $(x, y)$ has to be included in the basis.

The other test (line 10) is based on a counting argument. Suppose that after placing $j$ elements there are $G$ gaps, or target points not covered by the current sumset. No matter where the remaining $k - j$ elements are placed, they will generate at most $M = (j + 1) + (j + 2) + \cdots + k = (k + j + 1)(k - j)/2$ more sums. If $M < G$, then the current search branch cannot lead to any solutions.

This algorithm is quite simple, and there may be several ways to speed it up by exploiting the geometry of the problem. For example, instead of proceeding rowwise, the target rectangle can be explored in a different order: after completing the bottom edge $(y = 0)$, do next all of the left edge $(x = 0)$, then second row, second column, and so on.
The idea is to introduce necessary conditions from both the left and bottom edges early on. This change does not affect the validity of the algorithm. Empirically we observed that it saves about 37% of the running time in the example case of 19-bases of the 9-square.

As is typical for a combinatorial branch-and-bound method, the time requirement of this algorithm grows rapidly as $k$ increases. We implemented the algorithm in C++ and ran it on Intel Xeon E7-8890 processors (nominal clock frequency 2.2 GHz). For 19-bases of the 9-square the search took 0.44 hours of processor time; for 23-bases of the 11-square it took 1058 hours. Results are summarized in Table 1 (squares) and Table 2 (rectangles).

<table>
<thead>
<tr>
<th>$s$</th>
<th>$k$</th>
<th>$m$</th>
<th>$m_u$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>15</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>11</td>
<td>137</td>
<td>76</td>
</tr>
<tr>
<td>6</td>
<td>12</td>
<td>24</td>
<td>14</td>
</tr>
<tr>
<td>7</td>
<td>14</td>
<td>14</td>
<td>9</td>
</tr>
<tr>
<td>8</td>
<td>16</td>
<td>103</td>
<td>54</td>
</tr>
<tr>
<td>9</td>
<td>19</td>
<td>3531</td>
<td>1792</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>360</td>
<td>182</td>
</tr>
<tr>
<td>11</td>
<td>23</td>
<td>26857</td>
<td>13465</td>
</tr>
<tr>
<td>12</td>
<td>24</td>
<td>1585</td>
<td>797</td>
</tr>
<tr>
<td>13</td>
<td>26</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Minimal bases for squares.

4  Meet-in-the-middle method for restricted bases

In one dimension, i.e., for integer intervals, Kohonen proposed a meet-in-the-middle (MIM) method to find optimal restricted bases [10]. In its simplest form the MIM method splits a restricted basis at its midpoint into two components, a prefix and a suffix, which are then sought separately among the admissible bases of a smaller interval. It is much faster to consider all pairs of these components than to search directly for restricted bases by a method similar to Algorithm 1. The largest known optimal restricted bases for integer intervals have been computed by this method, with $k^*(734, 0) = 48$ [9].
The MIM method is extended to the planar setting as follows. We want to find all $k$-bases for $R = [0, s_x] \times [0, s_y]$, subject to the restriction $A \subseteq R_h = [0, h_x] \times [0, h_y]$, where $h_x = s_x/2 > 0$ and $h_y = s_y/2 > 0$. First we divide $R_h$ into four disjoint rectangles by choosing breaking points $a_x \in [0, h_x - 1]$ and $a_y \in [0, h_y - 1]$ arbitrarily, and defining

$$R_I = [0, a_x] \times [0, a_y],$$
$$R_{II} = [a_x + 1, h_x] \times [0, a_y],$$
$$R_{III} = [a_x + 1, h_x] \times [a_y + 1, h_y],$$
$$R_{IV} = [0, a_x] \times [a_y + 1, h_y].$$

These are the colored rectangles in the left part of Figure 2. Now split a basis $A$ into components $A_I, A_{II}, A_{III}, A_{IV}$ so that $A_I = A \cap R_I$, and similarly with the others. By the

| $s_x$ | $s_y$ | $k$ | $\Delta k$ | $m_u$ | $s_x$ | $s_y$ | $k$ | $\Delta k$ | $m_u$ | $s_x$ | $s_y$ | $k$ | $\Delta k$ | $m_u$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 0 | 0 | 1 | 0 | 1 | 6 | 5 | 12 | 0 | 660 | 9 | 7 | 17 | 0 | 5433 |
| 1 0 | 2 | 0 | 1 | 1 | 6 | 12 | 0 | 14 | 8 | 18 | 0 | 9171 |
| 1 3 | 0 | 1 | $\Delta k$ | 2 | 7 | 0 | 4 | $-1$ | 9 | 19 | 0 | 1792 |
| 2 0 | 2 | 0 | 1 | 1 | 7 | 1 | $-2$ | 28 | 10 | 0 | 5 | $-1$ | 8 |
| 1 4 | 0 | 3 | 2 | 2 | 8 | $-2$ | 5 | 1 | 8 | $-4$ | 19 |
| 2 4 | 0 | 1 | 3 | 10 | $-1$ | 25 | 2 | 10 | $-2$ | 174 |
| 3 0 | 3 | 0 | 2 | 4 | 11 | $-1$ | 50 | 3 | 12 | $-2$ | 203 |
| 1 5 | 0 | 6 | 5 | 13 | 0 | 924 | 4 | 13 | $-1$ | 64 |
| 2 6 | 0 | 16 | 6 | 14 | 0 | 3576 | 5 | 15 | $-1$ | 267 |
| 3 7 | 0 | 10 | 7 | 14 | $-1$ | 9 | 6 | 16 | 0 | 357 |
| 4 0 | 3 | 0 | 2 | 8 | 0 | 4 | $-1$ | 1 | 7 | 17 | $-1$ | 81 |
| 1 5 | $-1$ | 3 | 1 | 7 | $-3$ | 6 | 8 | 18 | 0 | 212 |
| 2 6 | 0 | 6 | 2 | 8 | $-2$ | 1 | 9 | 20 | 0 | 17076 |
| 3 8 | 0 | 75 | 3 | 11 | $-1$ | 325 | 10 | 20 | 0 | 182 |
| 4 8 | 0 | 5 | 4 | 11 | $-1$ | 4 | 11 | 0 | 5 | $-2$ | 1 |
| 5 0 | 4 | 0 | 5 | 5 | 13 | $-1$ | 3 | 1 | 9 | $-4$ | 258 |
| 1 6 | $-1$ | 10 | 6 | 14 | 0 | 73 | 2 | 10 | $-4$ | 3 |
| 2 7 | $-1$ | 1 | 7 | 15 | $-1$ | 16 | 3 | 13 | $-2$ | 1368 |
| 3 9 | 0 | 86 | 8 | 16 | 0 | 54 | 4 | 14 | $-2$ | 109 |
| 4 10 | 0 | 283 | 9 | 0 | 5 | $-1$ | 11 | 5 | 16 | $-1$ | 534 |
| 5 11 | 0 | 76 | 1 | 8 | $-3$ | 70 | 6 | 17 | $-1$ | 96 |
| 6 0 | 4 | 0 | 5 | 2 | 10 | $-2$ | 647 | 7 | 18 | $-1$ | 92 |
| 1 6 | $-2$ | 4 | 3 | 12 | $-1$ | 1940 | 8 | 19 | $-1$ | 12 |
| 2 8 | 0 | 101 | 4 | 13 | $-1$ | 920 | 9 | 21 | 0 | 13860 |
| 3 9 | $-1$ | 1 | 5 | 15 | 0 | 11479 | 10 | 22 | 0 | 42862 |
| 4 10 | 0 | 16 | 6 | 15 | $-1$ | 2 | 11 | 23 | 0 | 13465 |

Table 2: Minimal bases for rectangles.
non-negativity of all coordinates, any sumset involving \( A_{II}, A_{III} \) or \( A_{IV} \) is completely outside the lower left corner \( R_I \). So in order to have \( A + A \supseteq R \) we need \( A_I + A_I \supseteq R_I \). That is, \( A_I \) must be an admissible basis for \( R_I \). All candidates for \( A_I \) can be listed by Algorithm 1.

A similar argument applies in the lower right corner of the target, with some necessary coordinate transformations. Let \( C_{II} = [h_x + a_x + 1, s_x] \times [0, a_y] \). Then we need \( A_{II} + A_{II} \supseteq C_{II} \), since all the other component sumsets are outside \( C_{II} \). Consider the “mirror image” of \( A_{II} \), namely \( B_{II} = \{(h_x - x, y) : (x, y) \in A_{II}\} \). By construction, we have \( B_{II} \subseteq [0, b_x] \times [0, a_y] \), where for convenience we have written \( b_x = h_x - a_x - 1 \). Now the condition \( A_{II} + A_{II} \supseteq C_{II} \) implies that \( B_{II} + B_{II} \supseteq [0, b_x] \times [0, a_y] \). So \( B_{II} \) must be an admissible basis for \([0, b_x] \times [0, a_y]\), and again all candidates can be found by Algorithm 1.

![Figure 2: MIM decomposition of a restricted basis \( A \). The four components \( A_I, \ldots, A_{IV} \) are contained in the colored rectangles (left). Consequently, \( A + A \) (right) is the union of \( A_S, A_N \) and \( A_D \) (center), which are the self, neighboring and diagonal sums of the components. The extreme corners of \( A + A \) (areas within dashed rectangles) are covered only by the self sumsets, so \( A_I, \ldots, A_{IV} \) must be admissible bases for those rectangles (up to suitable coordinate transformations).

Similar conditions for \( A_{III} \) and \( A_{IV} \) apply in the remaining two corners. Consequently, \( A \) must be the union of four components, which are (mirror images of) admissible bases of suitable rectangles. Since we have so far only dealt with necessary conditions, we have not lost any possible solutions. The conditions guarantee only that the four extreme corner regions are covered; for any candidate solution \( A = A_I \cup A_{II} \cup A_{III} \cup A_{IV} \) we must finally check whether in fact \( A + A \supseteq R \).

Algorithm 2 gives a formal description of the MIM method. We choose \( a_x = \lfloor h_x/2 \rfloor \) and \( a_y = \lfloor h_y/2 \rfloor \) so the components have roughly equal dimensions. The final ingredient of the algorithm, on lines 8–14, concerns how the overall budget of \( k \) elements is allocated to the
Algorithm 2 Find all restricted $k$-bases for $[0, s_x] \times [0, s_y]$

1: procedure MIM($k, s_x, s_y$)
2: \hspace{1cm} $h_x \leftarrow s_x/2$ \hspace{1cm} \triangleright \text{dimensions of rectangle containing } A$
3: \hspace{1cm} $h_y \leftarrow s_y/2$
4: \hspace{1cm} $a_x \leftarrow \lceil h_x/2 \rceil$ \hspace{1cm} \triangleright \text{dimensions of rectangle containing } A_I$
5: \hspace{1cm} $a_y \leftarrow \lceil h_y/2 \rceil$
6: \hspace{1cm} $b_x \leftarrow h_x - a_x - 1$
7: \hspace{1cm} $b_y \leftarrow h_y - a_y - 1$ \hspace{1cm} \triangleright \text{dimensions of other rectangles}
8: \hspace{1cm} $k_{I_{\min}} \leftarrow k(a_x, a_y)$ \hspace{1cm} \triangleright \text{look up minimum sizes of the components}
9: \hspace{1cm} $k_{II_{\min}} \leftarrow k(b_x, a_y)$
10: \hspace{1cm} $k_{III_{\min}} \leftarrow k(b_x, b_y)$
11: \hspace{1cm} $k_{IV_{\min}} \leftarrow k(a_x, b_y)$
12: \hspace{1cm} \triangleright \text{Iterate feasible ways of allocating } k \text{ among the four quadrants}
13: \hspace{1cm} for $(k_I, k_{II}, k_{III}, k_{IV})$ such that $k_I + k_{II} + k_{III} + k_{IV} = k$ do
14: \hspace{1cm} \hspace{1cm} if $k_I \geq k_{I_{\min}} \land k_{II} \geq k_{II_{\min}} \land k_{III} \geq k_{III_{\min}} \land k_{IV} \geq k_{IV_{\min}}$ then
15: \hspace{1cm} \hspace{1cm} \hspace{1cm} \triangleright \text{Compute or look up admissible component bases}
16: \hspace{1cm} \hspace{1cm} $B_I \leftarrow \text{output from FINDBASES}(k_I, a_x, a_y)$
17: \hspace{1cm} \hspace{1cm} $B_{II} \leftarrow \text{output from FINDBASES}(k_{II}, b_x, a_y)$
18: \hspace{1cm} \hspace{1cm} $B_{III} \leftarrow \text{output from FINDBASES}(k_{III}, b_x, b_y)$
19: \hspace{1cm} \hspace{1cm} $B_{IV} \leftarrow \text{output from FINDBASES}(k_{IV}, a_x, b_y)$
20: \hspace{1cm} \hspace{1cm} for $(B_I, B_{II}, B_{III}, B_{IV}) \in B_I \times B_{II} \times B_{III} \times B_{IV}$ do
21: \hspace{1cm} \hspace{1cm} \hspace{1cm} $A_I \leftarrow B_I$
22: \hspace{1cm} \hspace{1cm} \hspace{1cm} $A_{II} \leftarrow \{(h_x - x, y) : (x, y) \in B_{II}\}$ \hspace{1cm} \triangleright \text{Mirror } x \text{ coordinates}
23: \hspace{1cm} \hspace{1cm} \hspace{1cm} $A_{III} \leftarrow \{(h_x - x, h_y - y) : (x, y) \in B_{III}\}$ \hspace{1cm} \triangleright \text{Mirror } x, y \text{ coordinates}
24: \hspace{1cm} \hspace{1cm} \hspace{1cm} $A_{IV} \leftarrow \{(x, h_y - y) : (x, y) \in B_{IV}\}$ \hspace{1cm} \triangleright \text{Mirror } y \text{ coordinates}
25: \hspace{1cm} \hspace{1cm} $A \leftarrow A_I \cup A_{II} \cup A_{III} \cup A_{IV}$ \hspace{1cm} \triangleright \text{Concatenate components}
26: \hspace{1cm} \hspace{1cm} if $A + A = R$ then PRINT($A$) \hspace{1cm} \triangleright \text{Found a basis}

four components. Note that $A_I$ need not be a minimal basis for $R_I$. It may have more than $k(a_x, a_y)$ elements, and indeed this may be necessary to find any solutions for $A + A \supseteq R$. The same goes for the other three components.

In order to determine the value of $k^*(s_x, s_y)$, just run Algorithm 2 repeatedly, beginning with $k = k_{I_{\min}} + k_{II_{\min}} + k_{III_{\min}} + k_{IV_{\min}}$ since certainly there are no solutions below that size, and increase $k$ in steps of 1 until some solutions are found.

Example 4. A restricted basis $A$ for $R = [0, 10] \times [0, 10]$ satisfies $A \subseteq R_h = [0, 5] \times [0, 5]$. The first quadrant of $R_h$ is $R_1 = [0, 2] \times [0, 2]$, and the other quadrants have the same size. Since $k(2, 2) = 4$, we have necessarily $|A| \geq 4 + 4 + 4 + 4 = 16$. There is only one 4-basis for $[0, 2] \times [0, 2]$, so for $k = 16$ there is only one combination to check in the innermost loop of Algorithm 2. But this combination does not give a basis for $[0, 10] \times [0, 10]$, so more than 16 elements are needed.
It turns out that \( k = 20 \) is enough. After some simple pruning conditions (not shown in Algorithm 2) we find that the only possible allocations of 20 elements are \((k_1, k_{II}, k_{III}, k_{IV}) = (4, 6, 4, 6)\) and \((5, 5, 5, 5)\). There are nine 5-bases and eighteen 6-bases for \([0, 2] \times [0, 2]\), so the first allocation leads to \(1 \cdot 18 \cdot 1 \cdot 18 = 324\) combinations to be checked, and the second gives \(9 \cdot 9 \cdot 9 \cdot 9 = 6561\) combinations. Out of these, we find 17 restricted solutions. This is less than one second of computation. In comparison, finding all 20-bases for the 10-square with our implementation of Algorithm 1 takes more than an hour.

There are a few ways to significantly prune the number of candidate solutions that need to be checked. Firstly, the complete sumset of a candidate restricted basis does not have to be calculated immediately. A necessary condition for a restricted basis is that any two neighboring quadrants form a restricted basis along one of the coordinate axes. It therefore suffices to first check whether this condition is satisfied for all four neighboring quadrant pairs. Only if the condition is met does the full sumset need to be checked.

Secondly, often some of the component pieces have the same dimensions (indeed all of them if \(h_x, h_y\) are both odd). If the pieces also have the same cardinality, then the set of candidate solutions is the same for all of them, up to suitable coordinate transformations.

**Example 5.** Consider a restricted basis \(A\) for the square \([0, s] \times [0, s]\), with \(s/2 = 2a + 1\) odd and \(a \geq 0\). Each quadrant has the same dimensions \(a_x = b_x = a_y = b_y = a\). If all component sets also have equal cardinality, then the candidates for \(A_{II}, A_{III}\) and \(A_{IV}\) are the same as for \(A_I\), up to suitable mirroring. Furthermore, if the sumset \((A_I \cup A_{II}) + (A_I \cup A_{II})\) does not cover \([0, s] \times [0, a]\), then all candidate solutions containing any rotation of this pair can be pruned.

Thirdly, when components have different cardinalities, the order in which they are concatenated matters. One possible strategy is to first concatenate component pairs of low cardinality, not only because they usually have fewer component solutions to try out, but also because they are less likely to produce feasible concatenations than pairs of higher cardinality. Occasionally, an infeasible concatenation rules out all potential solutions containing high cardinality components. Then these components do not even have to be computed in the first place.

**Example 6.** Consider the square restricted basis \(A\) in Example 5. Let the cardinality of this basis be \(4k_{\text{min}} + \tilde{k} = 4k_{\text{min}} + (\tilde{k}_I + \tilde{k}_{II} + \tilde{k}_{III} + \tilde{k}_{IV})\), where \(\tilde{k}_I, \ldots, \tilde{k}_{IV}\) represent the number of extra elements in each quadrant, which are now of equal dimensions. Let \(\tilde{k} = 3\). After accounting for rotational and mirror symmetries of \(A\), it turns out that there are four unique ways to distribute the extra elements: \((\tilde{k}_I, \tilde{k}_{II}, \tilde{k}_{III}, \tilde{k}_{IV}) = (0, 0, 0, 3), (0, 0, 1, 2), (0, 1, 0, 2),\) or \((0, 1, 1, 1)\). If the concatenation of pair \((\tilde{k}_I, \tilde{k}_{II}) = (0, 0)\) gives no solutions, then the candidate solutions containing pairs \((\tilde{k}_{III}, \tilde{k}_{IV}) = (0, 3)\) and \((\tilde{k}_{III}, \tilde{k}_{IV}) = (1, 2)\) are discarded. Consequently, we avoid computing the potentially many solutions of the \((k_{\text{min}} + 3)\)-basis altogether.
5 Numerical results

We now describe some results obtained for small rectangles with Algorithms 1 and 2. Examples of minimal bases are shown in Figures 3 and 4. We note that especially the restricted solutions in Figure 4 exhibit regular structure that can perhaps be generalized to larger bases.

In the result listings, \( m \) is the number of all minimal bases, and \( m_u \) is the number of “unique” bases after taking into account rotation and mirror symmetries. Each basis may have up to 8 symmetric variants if the target is square, and up to 4 variants otherwise.

![Figure 3: Some minimal bases for \( s_x = 7 \) and varying \( s_y \).](image)

5.1 Results for squares

Table 1 summarizes the minimal basis sizes for squares up to \( s = 13 \). For \( s \leq 12 \) we have generated and counted all minimal bases. Regarding \( s = 13 \), we deduce that \( k(13, 13) = 26 \), because Algorithm 1 finds no solutions with 25 elements, but we can construct a basis with 26 elements (see Figure 1d).

We also observe that in small even-sided instances \( s = 2, 4, 6, 8, 10, 12 \) one of the minimal solutions is the boundary basis. In small odd-sided instances \( s = 1, 3, 5, 9, 11 \) one of the minimal solutions is the L-shaped basis. Cases \( s = 7, 13 \) stand out as exceptions where the L-shaped basis is not minimal (recall Figures 1c and 1d).
Concerning the restricted case, Table 3 summarizes the results for squares up to \( s = 46 \). For \( s \leq 26 \) we generated and counted the minimal bases. For \( 28 \leq s \leq 46 \) we only determined the value of \( k^*(s, s) \), but did not generate the bases. For example, since we found that there is no restricted 91-basis for the 46-square, we can deduce that \( k^*(46, 46) = 92 \) as the boundary basis has this size. In all even-sided squares with \( 2 \leq s \leq 46 \), we have \( k^*(s, s) = 2s \), which is attained by the boundary basis.

Although the simple L-shaped and boundary bases provide minimal or almost minimal solutions for small squares, having the full collection of minimal solutions can be useful from an application perspective. In some sensor array applications it is beneficial to avoid placing sensor elements near each other, so as to avoid mutual coupling effects that cause degraded performance [16]. This may lead to a secondary optimization goal or constraint, and one may search the collection of minimal-size bases in order to satisfy this constraint.

5.2 Results for rectangles

The situation with rectangles is quite different from that with squares: if the aspect ratio \( \rho = (s_y + 1)/(s_x + 1) \) is not equal to 1, then minimal bases may be much smaller than the L-shaped and boundary bases.
Minimal bases for rectangles are summarized in Table 2, and Tables 4 and 5 for the restricted case. In order to compare the minimal solutions to the L-shaped and boundary bases, the quantity $\Delta k = k - k_t$ is computed. Here $k_t$ is the number of elements in the best applicable trivial solution, which is the boundary basis when $s_x$ and $s_y$ are even, and the L-shaped basis otherwise, except when $s_y = 0$ where the trivial solution is a one-dimensional basis with $\lceil s_x/2 \rceil + 1$ elements at $0, 1, \ldots, \lceil s_x/2 \rceil$.

Table 3: Minimal restricted bases for squares.

<table>
<thead>
<tr>
<th>$s$</th>
<th>$k^*$</th>
<th>$m$</th>
<th>$m_u$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>12</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>16</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>17</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>24</td>
<td>58</td>
<td>16</td>
</tr>
<tr>
<td>14</td>
<td>28</td>
<td>163</td>
<td>28</td>
</tr>
<tr>
<td>16</td>
<td>32</td>
<td>451</td>
<td>72</td>
</tr>
<tr>
<td>18</td>
<td>36</td>
<td>2047</td>
<td>276</td>
</tr>
<tr>
<td>20</td>
<td>40</td>
<td>8451</td>
<td>1133</td>
</tr>
<tr>
<td>22</td>
<td>44</td>
<td>43807</td>
<td>5575</td>
</tr>
<tr>
<td>24</td>
<td>48</td>
<td>213859</td>
<td>27108</td>
</tr>
<tr>
<td>26</td>
<td>52</td>
<td>1273607</td>
<td>159744</td>
</tr>
<tr>
<td>28</td>
<td>56</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>60</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>68</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>72</td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>76</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>80</td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>84</td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>88</td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>92</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$s_x$</td>
<td>$s_y$</td>
<td>$k^*$</td>
<td>$\Delta k$</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>------</td>
<td>----------</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>12</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>4</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>-2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>14</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>16</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>5</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>-2</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>13</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>16</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>18</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>5</td>
<td>-2</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>-4</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>14</td>
<td>-2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>18</td>
<td>0</td>
<td>14</td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>22</td>
<td>0</td>
<td>14</td>
</tr>
<tr>
<td>12</td>
<td>24</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>6</td>
<td>-2</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>-4</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>-2</td>
<td>15</td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>0</td>
<td>91</td>
</tr>
<tr>
<td>8</td>
<td>22</td>
<td>0</td>
<td>47</td>
</tr>
<tr>
<td>10</td>
<td>24</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>12</td>
<td>26</td>
<td>0</td>
<td>37</td>
</tr>
</tbody>
</table>

Table 4: Minimal restricted bases for rectangles.
Tables 2 and 4 show that minimal (unrestricted and restricted) bases use increasingly fewer elements than the trivial solutions as the aspect ratio deviates further from 1. This is also apparent from Figure 5, which shows the efficiency (1) of the minimal bases as a function of aspect ratio, along with the asymptotical efficiencies of the L-shaped basis, and two parametric bases that are introduced in section 6.2 (Definitions 11 and 13). Specifically, the L-shaped basis has efficiency $c \to \rho/(1 + \rho)^2$, as $s_x \to \infty$, since it requires $s_x + s_y + 1 = (1 + \rho)s_x + \rho$ elements for its sumset to cover the $[0, s_x] \times [0, s_y] = [0, s_x] \times [0, \rho(s_x + 1) - 1]$ rectangle. Similarly, the asymptotical efficiency of the dense-sparse and short-bars bases is $c = 1/4$, as shown later in Corollary 15 of section 6.2. The efficiency of the minimal bases in Figure 5 seem to approach 1/4 as $s_x$ and $\rho$ increase.

A peculiarity is illustrated in Figure 6, which shows two minimal restricted bases for which the number of elements actually decreases as the target width increases. Not only is $k^*(62, 2) = 28 > k^*(64, 2) = 26$, but the number of solutions for the two cases is also drastically different. The former has 125247 unique solutions, whereas the latter has only 1. The solutions for $s_y = 2$ listed in Table 5 reveal that a similar effect also occurs for $s_x = 104$ and 116. The same also applies to $s_y = 0$, since $k^*(s_x, 2) = 2k^*(s_x, 0)$ by Lemma 3.

An overview of currently known minimal restricted bases is shown in Figure 7. The colors of the pixels correspond to the minimal number of elements. At present, bases up to about $k = 50$ are practical to list exhaustively. For clarity of presentation, restricted one-dimensional bases are not plotted for $s_x > 120$.

<table>
<thead>
<tr>
<th>$s_x$</th>
<th>$k^*$</th>
<th>$m_u$</th>
<th>$s_x$</th>
<th>$k^*$</th>
<th>$m_u$</th>
<th>$s_x$</th>
<th>$k^*$</th>
<th>$m_u$</th>
<th>$s_x$</th>
<th>$k^*$</th>
<th>$m_u$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>4</td>
<td>1</td>
<td>32</td>
<td>18</td>
<td>1</td>
<td>62</td>
<td>28</td>
<td>125247</td>
<td>92</td>
<td>32</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>1</td>
<td>34</td>
<td>20</td>
<td>777</td>
<td>64</td>
<td>26</td>
<td>1</td>
<td>94</td>
<td>34</td>
<td>1284</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>1</td>
<td>36</td>
<td>20</td>
<td>50</td>
<td>66</td>
<td>28</td>
<td>654</td>
<td>96</td>
<td>34</td>
<td>222</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>1</td>
<td>38</td>
<td>20</td>
<td>8</td>
<td>68</td>
<td>28</td>
<td>62</td>
<td>98</td>
<td>34</td>
<td>88</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>1</td>
<td>40</td>
<td>20</td>
<td>1</td>
<td>70</td>
<td>28</td>
<td>3</td>
<td>100</td>
<td>34</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>10</td>
<td>1</td>
<td>42</td>
<td>22</td>
<td>412</td>
<td>72</td>
<td>28</td>
<td>1</td>
<td>102</td>
<td>36</td>
<td>74170</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>7</td>
<td>44</td>
<td>22</td>
<td>20</td>
<td>74</td>
<td>30</td>
<td>2415</td>
<td>104</td>
<td>34</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>12</td>
<td>1</td>
<td>46</td>
<td>24</td>
<td>32931</td>
<td>76</td>
<td>30</td>
<td>97</td>
<td>106</td>
<td>36</td>
<td>945</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
<td>20</td>
<td>48</td>
<td>24</td>
<td>3126</td>
<td>78</td>
<td>30</td>
<td>6</td>
<td>108</td>
<td>36</td>
<td>242</td>
</tr>
<tr>
<td>20</td>
<td>14</td>
<td>3</td>
<td>50</td>
<td>24</td>
<td>369</td>
<td>80</td>
<td>30</td>
<td>1</td>
<td>110</td>
<td>36</td>
<td>104</td>
</tr>
<tr>
<td>22</td>
<td>16</td>
<td>113</td>
<td>52</td>
<td>24</td>
<td>37</td>
<td>82</td>
<td>32</td>
<td>18937</td>
<td>112</td>
<td>38</td>
<td>283716</td>
</tr>
<tr>
<td>24</td>
<td>16</td>
<td>10</td>
<td>54</td>
<td>24</td>
<td>2</td>
<td>84</td>
<td>32</td>
<td>1561</td>
<td>114</td>
<td>38</td>
<td>42971</td>
</tr>
<tr>
<td>26</td>
<td>16</td>
<td>2</td>
<td>56</td>
<td>26</td>
<td>4337</td>
<td>86</td>
<td>32</td>
<td>193</td>
<td>116</td>
<td>36</td>
<td>1</td>
</tr>
<tr>
<td>28</td>
<td>18</td>
<td>162</td>
<td>58</td>
<td>26</td>
<td>239</td>
<td>88</td>
<td>32</td>
<td>8</td>
<td>118</td>
<td>38</td>
<td>454</td>
</tr>
<tr>
<td>30</td>
<td>18</td>
<td>22</td>
<td>60</td>
<td>26</td>
<td>36</td>
<td>90</td>
<td>32</td>
<td>2</td>
<td>120</td>
<td>38</td>
<td>202</td>
</tr>
</tbody>
</table>

Table 5: Minimal restricted bases for $s_y = 2$. 

An overview of currently known minimal restricted bases is shown in Figure 7. The colors of the pixels correspond to the minimal number of elements. At present, bases up to about $k = 50$ are practical to list exhaustively. For clarity of presentation, restricted one-dimensional bases are not plotted for $s_x > 120$. 


Figure 5: Efficiency of minimal bases, and asymptotical efficiency of L-shaped basis (dotted red line) and dense-sparse/short-bars bases (dashed blue line). The L-shaped basis is suboptimal when $\rho \neq 1$ and $s_x \to \infty$, whereas the dense-sparse and short-bars bases asymptotically achieve $c = 1/4$ for any $\rho$. The asymptotic efficiency of minimal bases is unknown.

Figure 6: Two restricted bases for $s_y = 2$, for which the minimal number of elements decreases as the rectangle width increases.

Figure 7: Minimal number of elements in restricted bases.
6 Bounds for large-scale behaviour

For very large rectangles it seems difficult to determine the minimum basis size exactly. Towards understanding the large-scale behaviour, we establish some upper and lower bounds on the efficiency \( c = N/k^2 \) (recall (1)) of such bases.

6.1 Upper bounds

A crude upper bound on efficiency is obtained by observing that from \( k \) elements at most \((k + 1)k/2\) different pairwise sums can be formed, considering that \( a + b = b + a \) and that sums of the form \( a + a \) are allowed. It follows that \( N \leq (k + 1)k/2 \), so for any planar basis we have

\[
c \leq 0.5 + O(1/\sqrt{N}).
\]

In one dimension, upper bounds tighter than 0.5 have been established by analytic and combinatorial methods. For all \( s_x \) large enough, by Yu’s Theorem 1.1 [25] we have

\[
s_x/k(s_x,0)^2 \leq 0.45851 = \alpha,
\]

and by Yu’s Theorem 1.2 [24] we have

\[
s_x/k^*(s_x,0)^2 \leq 0.41983 = \beta.
\]

Combining Yu’s theorems with simple counting, we obtain the following bounds with rectangles of small constant height. For brevity, if \( P \) is a set of points, we denote \( P_y = \{x : (x, y) \in P\} \) and call this the row \( y \) of \( P \).

**Theorem 7.** For all \( s_x \) large enough, any basis for \([0, s_x] \times [0, 1]\) has efficiency \( c < 0.4311 \).

**Proof.** Assume that \( s_x \) is large enough that (4) holds. Without loss of generality let \( A \) be admissible, and let its rows \( A_0, A_1 \) contain \( k_0, k_1 \) elements, respectively. Now \( A_0 + A_0 \) must cover \( R_0 = [0, s_x] \), and \( A_0 + A_1 \) must cover \( R_1 = [0, s_x] \). By applying (4) on row 0, and by counting sums on row 1, we obtain

\[
s_x \leq \alpha k_0^2, \quad s_x \leq k_0 k_1.
\]

For any \( k \), the minimum of these two bounds is maximized at \( k_1 = \alpha k_0 \), implying that \( k = (1 + \alpha)k_0 \) and

\[
s_x/k^2 \leq \frac{\alpha}{(1 + \alpha)^2} < 0.215542.
\]

Since \( N = |R| = 2(s_x + 1) \), we have \( N/k^2 < 0.4311 \) for \( s_x \) large enough.

**Theorem 8.** For all \( s_x \) large enough, any basis for \([0, s_x] \times [0, 2]\) has efficiency \( c < 0.4190 \).
Proof. Assume that $s_x$ is large enough that (4) holds. Without loss of generality let $A$ be admissible, and let its rows $A_0, A_1, A_2$ contain $k_0, k_1, k_2$ elements, respectively. Now $A_0 + A_0$ must cover $R_0 = [0, s_x]$, and $A_0 + A_1$ must cover $R_1 = [0, s_x]$, and finally $(A_0 + A_2) \cup (A_1 + A_1)$ must cover $R_2 = [0, s_x]$. By applying (4) on row 0, and by counting sums on rows 1 and 2, we obtain

$$s_x \leq \alpha k_0^2,$$
$$s_x \leq k_0 k_1,$$
$$s_x \leq k_0 k_2 + k_1^2/2 + k_1/2.$$

For any $k$, the minimum of these three bounds is maximized at their intersection, and by routine manipulations we obtain

$$s_x / k^2 \leq \frac{\alpha}{(1 + 2\alpha - \alpha^2/2)^2} + o_{s_x}(1) < 0.139663$$

for $s_x$ large enough. Since $N = |R| = 3(s_x + 1)$, we have $c = N/k^2 < 0.4190$ for $s_x$ large enough.

Any improvements to the one-dimensional bound (4) will imply corresponding improvements to Theorems 7 and 8. One could also apply the same proof technique with larger constant values of $s_y$, but it then becomes more complicated to maximize the simultaneous upper bounds of $s_x$. Numerical maximization suggests decreasing upper bounds as $s_y$ increases, for example, around 0.4126 with $s_y = 3$, and around 0.4087 with $s_y = 4$. This begs the question: what happens when $s_y$ goes to infinity?

Turning our attention to the restricted case we obtain the following bounds.

**Theorem 9.** For all $s_x$ large enough, any restricted basis for $[0, s_x] \times [0, 2]$ has efficiency $c < 0.3149$.

*Proof. Combine Lemma 3 with the bound (5) and the fact that $|R| = 3(s_x + 1)$.*

**Theorem 10.** For all $s_x$ large enough, any restricted basis for $[0, s_x] \times [0, 4]$ has efficiency $c < 0.3585$.

*Proof. Assume that $s_x$ is large enough that (5) holds. Let $A$ be a restricted basis for $R$, and let $k_0, k_1, k_2$ be the cardinalities of its rows. By applying (5) on rows 0 and 4 of the target, and by counting sums on rows 1 and 3, we obtain

$$s_x \leq \beta k_0^2,$$
$$s_x \leq k_0 k_1,$$
$$s_x \leq k_1 k_2,$$
$$s_x \leq \beta k_2^2.$$
The minimum of these four bounds is maximized at their intersection, where \( k_0 = k_2 \) and \( k_1 = \beta k_0 \), thus \( k = (2 + \beta)k_0 \). Then we obtain

\[
\frac{s_x}{k^2} \leq \frac{\beta}{(2 + \beta)^2} < 0.071698.
\]

Since \( N = |R| = 5(s_x + 1) \), we have \( c = N/k^2 < 0.3585 \) for \( s_x \) large enough.

### 6.2 Lower bounds

As with one-dimensional bases, also in planar bases it is relatively easy to obtain an efficiency of approximately 1/4 for large rectangles. For squares this is particularly easy: the L-shaped basis for an \( s \)-square has \( k = 2s + 1 \), so \( c = 0.25 + O(1/s) \). The boundary basis has \( k = 2s \), so its efficiency has the same asymptotic form.

However, for non-square rectangles, the efficiency of the L-shaped and boundary bases falls below 1/4. Indeed, consider rectangles \([0, s_x] \times [0, s_y]\) with a constant aspect ratio \( \rho \neq 1 \). The L-shaped basis has

\[
k = s_x + s_y + 1 = (1 + \rho)s_x + \rho,
\]

so

\[
c \rightarrow \frac{\rho}{(1 + \rho)^2} < 1/4
\]

as \( s_x \rightarrow \infty \). The case with the boundary basis is similar. For example, if the aspect ratio is \( \rho = 1/9 \), then both the L-shaped and boundary bases have only \( c \rightarrow 0.09 \) in the limit.

The following parametric constructions demonstrate that an asymptotic efficiency of 1/4 can be achieved with rectangles of essentially any constant aspect ratio, as both \( s_x \) and \( s_y \) go to infinity. The first construction, a **dense-sparse basis**, is the union of a dense part (a filled rectangle) and a sparse part (regularly spaced single elements). The second construction, a **short-bars basis**, consists of short, regularly spaced horizontal and vertical bars. Both constructions are illustrated in Figure 8. We use here the notation

\[
[a, (t), b] = \{a, a + t, a + 2t, \ldots, b\}
\]

for a finite arithmetic progression from \( a \) to \( b \) with step length \( t \), with the provision that \( b - a \) is divisible by \( t \).

**Definition 11.** The **dense-sparse basis** with parameters \( t_x, t_y \geq 1 \) is the set \( A = B \cup C \), where \( B = [0, t_x - 1] \times [0, t_y - 1] \) and \( C = [0, (t_x), t_x^2 - t_x] \times [0, (t_y), t_y^2 - t_y] \).

**Theorem 12.** The dense-sparse basis has \(|A| = 2t_x t_y - 1\) and \( A + A \supseteq [0, t_x^2 - 1] \times [0, t_y^2 - 1] \).

**Proof.** Since \(|B| = |C| = t_x t_y \) and \( B \cap C = \{(0, 0)\} \), the claim on \(|A|\) follows. For any point \((x, y) \in R\), let \( x = b_x + c_x \) with \( b_x \in [0, t_x - 1] \) and \( c_x \in [0, (t_x), t_x^2 - t_x] \). Similarly let \( y = b_y + c_y \) with \( b_y \in [0, t_y - 1] \) and \( c_y \in [0, (t_y), t_y^2 - t_y] \). Now \((x, y) = (b_x, b_y) + (c_x, c_y) \) with \((b_x, b_y) \in B \) and \((c_x, c_y) \in C \). Thus \((x, y) \in B + C \subseteq A + A \). \( \square \)
Figure 8: Two bases for the rectangle $[0, 24] \times [0, 8]$: (a) a dense-sparse basis (Definition 11), (b) a short-bars basis (Definition 13), both with parameters $t_x = 5$, $t_y = 3$. Both bases have only 29 elements, while an L-shaped basis for the same rectangle has 33 elements.

**Definition 13.** The short-bars basis with parameters $t_x, t_y \geq 1$ is the set $A = B \cup C$, where $B = [0, t_x - 1] \times [0, (t_y^2 - t_y)]$ and $C = [0, (t_x), t_x^2 - t_x] \times [0, t_y - 1]$.

**Theorem 14.** The short-bars basis has $|A| = 2t_x t_y - 1$ and $A + A \supseteq [0, t_x^2 - 1] \times [0, t_y^2 - 1]$.

**Proof.** Since $|B| = |C| = t_x t_y$ and $B \cap C = \{(0, 0)\}$, the claim on $|A|$ follows. For any point $(x, y) \in R$, let $x = b_x + c_x$ with $b_x \in [0, t_x - 1]$ and $c_x \in [0, (t_x), t_x^2 - t_x]$. Similarly let $y = b_y + c_y$ with $b_y \in [0, (t_y), t_y^2 - t_y]$ and $c_y \in [0, t_y - 1]$. Now $(x, y) = (b_x, b_y) + (c_x, c_y)$ with $(b_x, b_y) \in B$ and $(c_x, c_y) \in C$. Thus $(x, y) \in B + C \subseteq A + A$. 

**Corollary 15.** Let $\rho = p^2/q^2$ be a fixed aspect ratio, where $p$ and $q$ are integers, and let $h \geq 1$ be an integer. Then both the dense-sparse basis and the short-bars basis, with parameters $t_x = qh$ and $t_y = ph$, are bases for the rectangle $[0, t_x^2 - 1] \times [0, t_y^2 - 1]$, which has the said aspect ratio. The efficiency of either basis is

$$c = \frac{t_x^2 t_y^2}{(2t_x t_y - 1)^2} = 0.25 + O(1/h^2).$$

**Corollary 16.** For any fixed aspect ratio $\rho = p^2/q^2 \neq 1$, with $p$ and $q$ integers, the L-shaped basis and the boundary bases are asymptotically suboptimal as $s_x \to \infty$.

For arbitrarily wide rectangles of any constant height we present a basis construction whose asymptotic efficiency exceeds $1/4$. The construction is somewhat analogous to Mrose’s one-dimensional basis [18], hence the name. As a mnemonic for our symbols here, note that in $I_1, I_2, I_3$ the set of $x$ coordinates is an interval; in $T$ it is a $t$-step arithmetic progression; and in $S$ it is a “sparse” $(t + 1)$-step arithmetic progression.
Definition 17. The stacked Mrose basis with parameters $s_y \geq 0$ and $t \geq 1$ is the set $I_1 \cup I_2 \cup I_3 \cup T \cup S$, where

$$I_1 = [0, t] \times Y,$$
$$T = [0, (t), at^2 - t] \times \{0\},$$
$$S = [at^2, (t + 1), (a + 1)t^2 - 1] \times Y,$$
$$I_2 = [2at^2, 2at^2 + t] \times Y,$$
$$I_3 = [(3a + 1)t^2, (3a + 1)t^2 + t] \times Y,$$
and $Y = [0, s_y]$ and $a = 4s_y + 3$.

Theorem 18. If $A$ is a stacked Mrose basis, then $|A| = (8s_y + 7)t + (3s_y + 1)$ and $A + A \supseteq [0, (16s_y + 14)t^2 - 1] \times [0, s_y]$.

Proof. Let us first determine the size of the basis. We observe that $|I_1| = |I_2| = |I_3| = (t + 1)(s_y + 1)$, $|T| = at$, and $|S| = t(s_y + 1)$. Because the parts are otherwise disjoint except that $I_1 \cap T = \{(0, 0), (t, 0)\}$, the claim on $|A|$ follows.

Let us next verify that $A + A$ covers the desired target rectangle. We check seven consecutive subrectangles in turn.

1. $[0, at^2 - 1] \times Y$ is covered by $I_1 + T$.
2. $[at^2, (a + 1)t^2 - 1] \times Y$ is covered by $I_1 + S$.
3. $[(a + 1)t^2, 2at^2 - 1] \times Y$ is covered by $T + S$.
4. $[2at^2, 3at^2 - 1] \times Y$ is covered by $I_2 + T$.
5. $[3at^2, (3a + 1)t^2 - 1] \times Y$ is covered by $I_2 + S$.
6. $[(3a + 1)t^2, (4a + 1)t^2 - 1] \times Y$ is covered by $I_3 + T$.
7. $[(4a + 1)t^2, (4a + 2)t^2 - 1] \times Y$ is covered by $I_3 + S$.

Because $I_1, I_2, I_3, T, S \subseteq A$, combining observations (1)–(7) and $4a + 2 = 16s_y + 14$ we have

$$A + A \supseteq [0, (16s_y + 14)t^2 - 1] \times Y$$

as claimed. \qed

Corollary 19. The stacked Mrose basis has efficiency

$$c = \frac{N}{k^2} = \frac{(16s_y + 14)t^2 \cdot (s_y + 1)}{((8s_y + 7)t)^2 + O(t)} \xrightarrow{t \to \infty} \frac{2s_y + 2}{8s_y + 7}.$$

Example 20. With $s_y = 1$, Definition 17 gives a basis of size $k = 15t + 4$ for the rectangle $[0, 30t^2 - 1] \times [0, 1]$, with efficiency tending to $4/15 \approx 0.2666$ as $t \to \infty$. 21
Example 21. With $s_y = 2$, Definition 17 gives a basis of size $k = 23t + 7$ for the rectangle $[0, 46t^2 - 1] \times [0, 2]$, with efficiency tending to $6/23 > 0.2609$ as $t \to \infty$. Figure 9 illustrates this basis in the case of $t = 10$.

Although a stacked Mrose basis can be constructed arbitrarily high, its efficiency tends down to $1/4$ as $s_y$ goes to infinity. We do not know whether $1/4$ can be asymptotically exceeded for rectangles with both dimensions going to infinity (e.g., with a constant aspect ratio).

Figure 9: A schematic illustration of the stacked Mrose basis (Definition 17) with parameters $s_y = 2$ and $t = 10$. In this case $a = 11$ and $s_x = 4599$.

7 Conclusion and open questions

Planar additive bases are a natural generalization of the classical one-dimensional additive bases, and it may be slightly surprising that they have not been studied much. In this paper, some initial results have been provided. For small squares and rectangles, we have determined the minimum cardinalities exactly; and for larger instances, we have established some lower and upper bounds, although the bounds are not very tight.

Apart from the obvious desires of extending the finite results and improving the bounds, we would like to pose some open questions. For squares, the “trivial” L-shaped and boundary bases achieve an asymptotic efficiency of $1/4$. For non-square rectangles, the trivial bases fall below $1/4$, but our parametric constructions still attain $1/4$. Can this be improved at all? We pose this question in two forms, one for squares and one for general rectangles.

**Question 22.** Does any square $[0, s] \times [0, s]$ admit an additive basis of less than $2s$ elements?

**Question 23.** Is there a constant $c > 1/4$ such that there are arbitrarily high and wide rectangles admitting an additive basis of efficiency at least $c$?

We would not be too surprised by a positive answer to our questions above. We recall that in one-dimensional additive bases, contrary to Rohrbach’s conjecture [22, p. 9], parametric constructions with efficiencies over $1/4$ have been found [5, 18, 11]. Also, for constant-height rectangles, our parametric construction (stacked Mrose basis) exceeds $1/4$. But if both dimensions tend to infinity, the question is open.
For other directions of future research, we note that additive bases are conceptually closely related to difference bases, where the object of interest is the difference set \( A - A \). One-dimensional difference bases have been studied by, e.g., Leech [14] and Wichmann [23]. Difference bases find applications in sensor arrays for passive sensing, particularly when second-order statistics of the element outputs are processed [8]. Due to the use of data covariance in many applications, such as direction-of-arrival estimation, both one- and two-dimensional difference bases have received attention recently [15, 20, 16]. We also point out that non-rectangular, for example hexagonal, grids have received some attention in array processing using difference bases [7]. Additive bases for non-rectangular targets would be another interesting direction of research.

8 Acknowledgments

We thank the anonymous referee for insightful suggestions to improve the exposition.

References


2010 Mathematics Subject Classification: Primary 11B13.
Keywords: additive basis, restricted basis, planar basis, rectangular sumset.

(Concerned with sequences A001212, A008574, A295771, A295774.)

Received August 15 2018; revised version received December 17 2018. Published in Journal of Integer Sequences, December 17 2018.

Return to Journal of Integer Sequences home page.