Algorithms for Computing the $h$-Range of the Postage Stamp Problem

By Svein Mossige

Abstract. New algorithms, based on a very efficient method to compute the $h$-range, have been used to extend known tables of the extremal $h$-range, to complete the solution in the case $k = 3$, and to find a lower bound for the extremal 2-range.

1. Introduction. The postage stamp problem consists of choosing, for given $h$ and $k$, a set of $k$ positive integers such that

(a) sums of $h$ (or fewer) of these integers can realize the numbers $1, 2, \ldots, n$;
(b) the value of $n$ in (a) is as large as possible.

Let $h$ and $k$ be given positive integers and $A_k = \{a_1, a_2, \ldots, a_k\}$ a set of distinct integers such that $1 = a_1 < a_2 < \cdots < a_k$. We form the set of linear combinations

$$S = \left\{ \sum_{i=1}^{k} x_i a_i; x_i > 0, \sum_{i=1}^{k} x_i < h \right\}.$$

The number $n$ is called the $h$-range of the basis $A_k$ if the set $S$ contains the integers $1, 2, \ldots, n$, that is, $[1, n] \subseteq S$, and $n + 1 \notin S$. The $h$-range of $A_k$ is denoted by $n_h(A_k)$.

For given $h$ and $k$, an extremal basis is a basis of $k$ elements for which $n$ is as large as possible. The corresponding extremal $h$-range is denoted by $n_h(k)$.

The set $A_k$ is called an admissible basis if

$$a_i < a_{i+1} < n_h(A_i) + 2, \quad i = 1, 2, \ldots, k - 1,$$

that is, $n_h(A_k) > a_k$.

In what follows, only such bases $A_k$ will be considered. Let $h_0$ denote the smallest possible $h$ such that $A_k$ is an admissible basis.

For all $k$ and $h > h_0$, we trivially have

$$n_{h+1}(A_k) > n_h(A_k) + a_k.$$  

Further, Selmer [10] proved that, for arbitrary $k$ and $h > h_0$,

$$n_h(A_k) > (h + 1)a_{k-1} - a_k$$

implies

$$n_{h+1}(A_k) = n_h(A_k) + a_k.$$  

If $h$ is increased by 1, the right-hand side of (3) increases with $a_{k-1}$, while the left-hand side increases with at least $a_k$. There is consequently an $h_1 (\geq h_0)$ such that (3) and hence (4) are satisfied for all $h > h_1$. This means that for given $h$, 

---

Received October 10, 1979; revised July 10, 1980.
1980 Mathematics Subject Classification. Primary 10J79, 10-04.
2. Recursive Definition of the Set $S$. To simplify the notation, we introduce the artificial basis element $a_0 = 0$. For given $h$, $k$, and $A_k$, the set $S = s(h, k)$ can then be defined recursively by

$$
\begin{align*}
\cases{
  s(1, k) = A_k \cup \{0\}, \\
  s(r, k) = \bigcup_{i=0}^{r-1} (a_i + s(r-1, k)), & r = 2, 3, \ldots, h,
}\end{align*}
$$

where, as usual,

$$
a_i + s(r - 1, k) = \{a_i + b; b \in s(r - 1, k)\}.
$$

We further need the obvious relation

$$
s(h, k) = \bigcup_{i=0}^{h} (i a_k + s(h - i, k - 1)).
$$

The argument $k - 1$ indicates that $a_k$ is removed from the basis. We define $s(0, k) = s(h, 0) = \{0\}$.

From (6) and (7), we see that $s(r, k)$ is a union of the set $s(r - 1, k)$ and the sets given by adding $a_i$ ($i > 1$) to each element in $s(r - 1, k)$. The following interpretation of the process may be enlightening:

Let $b_0 = 0, b_1, \ldots, b_v$, be all the elements in the set $s(r - 1, k)$, and mark a ruler at distances $b_0, b_1, \ldots, b_v$ from its starting point. Place this point at the origin of an axis and transfer to the axis the marks on the ruler. Then translate the ruler along the axis, first $a_1$ units and transfer the marks to the axis, then $a_2$ units, etc., up to $a_k$ units and transfer each time the marks on the ruler to the axis. The numbers corresponding to the marks on the axis are then exactly the elements of the set $s(r, k)$.

(8) suggests a method to compute $n_h(A_{k-1} \cup \{a_k\})$ based on the information achieved by the computation of $n_h(A_{k-1})$.

3. The $h$-Range (Basic Algorithm). In the computer, we represent the set $s(r, k) \setminus \{0\}$, $1 < r < h$, as a bit string $B_r$ of length $ra_k$:

$$
B_r: \text{bit } t = 1 \iff t \in s(r, k) \setminus \{0\}.
$$

In particular, $B_1$ corresponds to the basis $A_k$.

Let $a_i B_{r-1}$ denote the bit string $B_{r-1}$ shifted $a_i$ places to the right (filling in with zeros to the left). Then the recursive definition (6) is equivalent to the following set of OR operations:

$$
B_r = \bigvee_{i=0}^{k} a_i B_{r-1}, \quad r = 2, 3, \ldots, h.
$$

Each time a new $B_r$ is constructed, we check if the $a_k$ left-most bits in $B_r$ are 1 bits. When this is the case for the first time, we have $n_r(A_k) > a_k$ and hence $h_0 = r$. For each $r > h_0$, $n_r(A_k)$ is determined by scanning the bit string $B_r$. The first zero is then in position $n_r(A_k) + 1$.

If $h_0 < h$, we check for each $r$, $h_0 < r < h$, whether (3) is satisfied. If this occurs for the first time for $r = h_1$, we use (5) to determine $n_h(A_k)$. For large $h$, this simple device yields a significant reduction of the computing time.
Another device, also very effective for large $h$, is to reduce the number of registers needed to hold the bit strings $B_r$. For $r > h_0$, we can delete the first $n_r(A_k)$ 1 bits. From (2), it follows that no bit string length for $r > h_0$ will then exceed $h_0a - n_h(A_k)$.

When computing $n_h(k)$ by the method of Section 4, we may also save a substantial amount of time by chopping off the bit strings $B_r$ at the other end. Keeping $k$ fixed, the extremal range $n_h(k)$ turns out to increase fairly regularly with $h$. We can therefore estimate a safe upper bound for $n_h(k)$ and delete those parts of the bit strings $B_r$ which exceed this bound.

4. **The Extremal $h$-Range** $n_h(k)$, $h > 2$. For given $h$ and $k$, let $U_h$ be the universe of all admissible sets $A_k$, defined by the conditions (1). To find the set(s) with the extremal $h$-range, we scan the universe $U_h$. For each $A_k \in U_h$, the $h$-range $n_h(A_k)$ is computed by the basic algorithm.

An alternative approach (not used in our calculations) is to extract from the basic algorithm also the ranges $n_r(A_k)$ for $r < h$. This would allow for a simultaneous computation of extremal ranges for all $r < h$, at the cost of larger storage requirements.

If we know a lower bound $L$ such that $n_h(k) > L$, we can skip all $A_k \in U_h$ with $ha < L$. Having already calculated $n_{h-1}(k)$ and a corresponding extremal basis $A^*_k$, we may start the scanning of $U_h$ with $L = n_{h-1}(k) + q^*_k$. Whenever we find a larger $n_h(A_k)$, this can replace the previous bound $L$. To get large bounds as quickly as possible, we scan the intervals (1) for $a_i$ downwards. This also simplifies the exclusion of all $A_k$ with $ha < L$.

5. **The Extremal 2-Range** $n_2(k)$. For $h = 2$, the devices mentioned at the end of Section 3 are of no use. We may instead utilize (8), which now takes the form

$$s(2, k) = s(2, k - 1) \cup (a_k + (A_k - 1) \cup \{0\}) \cup \{2a_k\}.$$  

We scan $U_2$ for $k - 1$, corresponding to $s(2, k - 1)$. Each $A_k - 1$ is then extended to $A_k$, where $a_{k-1} < a_k < n_2(A_{k-1}) + 2$ by (1). In the computer representation of (10) as a bit string, we may clearly delete $\{a_k\}$ (since $A_k$ is admissible). We may also delete $\{2a_k\}$, if we note that a calculated 2-range $2a_k - 1$ then really corresponds to the maximal range $2a_k$. As a result, we get the extremal ranges $n_2(k - 1)$ and $n_2(k)$ simultaneously.

Because of the simplicity of (10), this method leads to a surprisingly large saving in computing time. A simultaneous computation of $n_2(9)$ and $n_2(10)$ used only 25% of the time needed to determine these extremal ranges individually by the method of Section 4.

Also for small $h > 2$, the above method may represent an interesting alternative to such individual computations.

Riddell and Chan [7] have another algorithm for $h = 2$. Their method can be used for $h > 2$, but then their universe of sets is larger than necessary, including also nonadmissible sets.

Lunnon’s algorithm [4] utilizes (8) for all $h$. For fixed $k$, this requires the storing of all bit strings $s(h - i, k - 1)$, $i = 0, 1, \ldots, h$. 

License or copyright restrictions may apply to redistribution; see http://www.ams.org/journal-terms-of-use
6. The Extremal 2-Range for Symmetric Bases. A set \( A_k \cup \{a_0 = 0\} \) is symmetric if \( a_i + a_{k-i} = a_k \) for \( i = 0, 1, \ldots, k \). The 2-range of a symmetric basis is \( 2a_k \); cf. [8].

To compute the extremal 2-range for symmetric bases, it is sufficient to scan all admissible sets \( A_{k/2} \) for \( k \) even and all sets \( A_{(k+1)/2} \) for \( k \) odd. The computing time can be reduced by the methods described at the end of Section 4. The amount of work is comparable to that of computing \( n_2((k + 1)/2) \).

7. Results of the Calculations. The algorithms were written in assembler language and Fortran and performed on the Univac computer at the University of Bergen.

For \( k = 2 \), explicit formulas for the extremal bases and ranges are well known; cf. [11].

In 1968, Hofmeister [2] solved the corresponding problem for \( k = 3 \) almost completely, giving formulas which are valid for sufficiently large \( h \). He also gave a table for \( h < 34 \) (where the "anticipated" extremal basis \( \{1, 19, 102\} \) for \( h = 22 \) is missing). Recently, Hofmeister [3] has shown that it suffices to check separately the cases with \( h < 200 \). Using results from Selmer's paper [10], we have performed this check. It turns out that for \( h > 22 \), Hofmeister's formulas for \( k = 3 \) cover all extremal bases.

For \( k > 4 \), the standard table of reference is that of Lunnon [4]. In addition, Seldon [9] computed \( n_5(10) \), Phillips [6] \( n_4(8) \), and Riddell and Chan [7] \( n_2(13) \).

Our programs used only 4.5 hours to verify Lunnon's tables. This figure reflects the speed of Univac 1100/82, but even more the efficiency of our basic bit string algorithm.

We have extended earlier tables in four directions:

For \( k = 4 \), Table 1 below lists \( n_h(4) \) and the corresponding extremal bases for \( 2 < h < 28 \).

<table>
<thead>
<tr>
<th>( h )</th>
<th>( n_h(4) )</th>
<th>( a_2 )</th>
<th>( a_3 )</th>
<th>( a_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>12</td>
<td>3</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>24</td>
<td>4</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>44</td>
<td>3</td>
<td>11</td>
<td>18</td>
</tr>
<tr>
<td>5</td>
<td>71</td>
<td>4</td>
<td>12</td>
<td>21</td>
</tr>
<tr>
<td>( 5 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>114</td>
<td>4</td>
<td>19</td>
<td>33</td>
</tr>
<tr>
<td>7</td>
<td>165</td>
<td>5</td>
<td>24</td>
<td>37</td>
</tr>
<tr>
<td>8</td>
<td>234</td>
<td>6</td>
<td>25</td>
<td>65</td>
</tr>
<tr>
<td>9</td>
<td>326</td>
<td>5</td>
<td>34</td>
<td>60</td>
</tr>
<tr>
<td>10</td>
<td>427</td>
<td>6</td>
<td>41</td>
<td>67</td>
</tr>
<tr>
<td>11</td>
<td>547</td>
<td>7</td>
<td>48</td>
<td>85</td>
</tr>
<tr>
<td>12</td>
<td>708</td>
<td>7</td>
<td>48</td>
<td>126</td>
</tr>
<tr>
<td>13</td>
<td>873</td>
<td>9</td>
<td>56</td>
<td>155</td>
</tr>
<tr>
<td>14</td>
<td>1094</td>
<td>8</td>
<td>61</td>
<td>164</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( h )</th>
<th>( n_h(4) )</th>
<th>( a_2 )</th>
<th>( a_3 )</th>
<th>( a_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>1383</td>
<td>12</td>
<td>65</td>
<td>240</td>
</tr>
<tr>
<td>16</td>
<td>1650</td>
<td>11</td>
<td>78</td>
<td>216</td>
</tr>
<tr>
<td>17</td>
<td>1935</td>
<td>11</td>
<td>90</td>
<td>252</td>
</tr>
<tr>
<td>18</td>
<td>2304</td>
<td>16</td>
<td>73</td>
<td>338</td>
</tr>
<tr>
<td>19</td>
<td>2782</td>
<td>10</td>
<td>99</td>
<td>360</td>
</tr>
<tr>
<td>20</td>
<td>3324</td>
<td>16</td>
<td>103</td>
<td>488</td>
</tr>
<tr>
<td>21</td>
<td>3812</td>
<td>16</td>
<td>103</td>
<td>488</td>
</tr>
<tr>
<td>22</td>
<td>4368</td>
<td>12</td>
<td>121</td>
<td>561</td>
</tr>
<tr>
<td>23</td>
<td>5130</td>
<td>14</td>
<td>142</td>
<td>659</td>
</tr>
<tr>
<td>24</td>
<td>5892</td>
<td>16</td>
<td>163</td>
<td>757</td>
</tr>
<tr>
<td>25</td>
<td>6745</td>
<td>20</td>
<td>149</td>
<td>860</td>
</tr>
<tr>
<td>26</td>
<td>7880</td>
<td>16</td>
<td>194</td>
<td>734</td>
</tr>
<tr>
<td>27</td>
<td>8913</td>
<td>21</td>
<td>177</td>
<td>1006</td>
</tr>
<tr>
<td>28</td>
<td>9919</td>
<td>21</td>
<td>177</td>
<td>1006</td>
</tr>
</tbody>
</table>

Table 2 gives some extremal ranges for \( k = 5 \) and \( k = 6 \), with bases.
Table 2

<table>
<thead>
<tr>
<th>( k )</th>
<th>( h )</th>
<th>( n_h(k) )</th>
<th>( a_2 )</th>
<th>( a_3 )</th>
<th>( a_4 )</th>
<th>( a_5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>10</td>
<td>1055</td>
<td>8</td>
<td>27</td>
<td>119</td>
<td>194</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>1475</td>
<td>10</td>
<td>34</td>
<td>165</td>
<td>270</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>2047</td>
<td>10</td>
<td>26</td>
<td>195</td>
<td>320</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>664</td>
<td>7</td>
<td>12</td>
<td>64</td>
<td>113</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>1045</td>
<td>9</td>
<td>14</td>
<td>65</td>
<td>170</td>
</tr>
</tbody>
</table>

Table 3 shows the three extremal bases corresponding to \( n_2(14) = 80 \). The first one of these bases inspired the appendix below.*

Table 3

<table>
<thead>
<tr>
<th>( a_2 )</th>
<th>( a_3 )</th>
<th>( a_4 )</th>
<th>( a_5 )</th>
<th>( a_6 )</th>
<th>( a_7 )</th>
<th>( a_8 )</th>
<th>( a_9 )</th>
<th>( a_{10} )</th>
<th>( a_{11} )</th>
<th>( a_{12} )</th>
<th>( a_{13} )</th>
<th>( a_{14} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5</td>
<td>8</td>
<td>11</td>
<td>14</td>
<td>17</td>
<td>20</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>51</td>
<td>53</td>
<td>55</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>5</td>
<td>8</td>
<td>14</td>
<td>20</td>
<td>26</td>
<td>32</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>39</td>
<td>40</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>9</td>
<td>10</td>
<td>15</td>
<td>16</td>
<td>21</td>
<td>22</td>
<td>24</td>
<td>25</td>
<td>51</td>
<td>53</td>
<td>55</td>
</tr>
</tbody>
</table>

Table 4 gives the extremal 2-ranges for symmetric bases with \( 15 < k < 30 \). The set of differences \( d_i = a_i - a_{i-1} \) is also symmetric, and is only listed up to the middle of the set.

Table 4

<table>
<thead>
<tr>
<th>( k )</th>
<th>( n_2(A_k) )</th>
<th>( d_i = a_i - a_{i-1}, ; i = 1, 2, \ldots, \left\lceil \frac{k+1}{2} \right\rceil )</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>92</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>16</td>
<td>104</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>17</td>
<td>116</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>18</td>
<td>128</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>19</td>
<td>140</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>20</td>
<td>152</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>21</td>
<td>164</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>22</td>
<td>180</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>23</td>
<td>196</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>24</td>
<td>212</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>25</td>
<td>228</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>26</td>
<td>244</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>27</td>
<td>262</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>28</td>
<td>280</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>29</td>
<td>298</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
<tr>
<td>30</td>
<td>316</td>
<td>1 2 1 1 3 6 6 6</td>
</tr>
</tbody>
</table>

---

*\( n_2(15) = 92 \) and \( n_2(16) = 104 \) with the extremal bases given in Table 4.
Appendix, by Torleiv Kløve and Svein Mossige. Rohrback [8] proved that
\[ n_2(k) > \frac{1}{4} k^2 + O(k) \]
and conjectured that this is best possible. However, Hämmerer and Hofmeister [1] proved that
\[ n_2(k) > \frac{5}{18} k^2 + O(k). \]
We shall prove that
\[ n_2(k) > \frac{2}{7} k^2 + O(k). \]
The same result has been proved independently by Mrose [5] by a different and more complicated construction.

We use the following notation: Let a, b, and c be positive integers such that \( b > a \) and c divides \( b - a \), and let d be a nonnegative integer. Then
\[
\begin{align*}
\begin{bmatrix} a(c)b \end{bmatrix} &= \{ a + ic | 0 < i < (b - a)/c \}, \\
\begin{bmatrix} a,b \end{bmatrix} &= \begin{bmatrix} a(1)b \end{bmatrix}, \\
B(a,d) &= \begin{bmatrix} a(d)a + d^2 \end{bmatrix}.
\end{align*}
\]

**Lemma.** Let \( x, y, \) and \( z \) be positive integers where \( y > x > 2 \), and let
\[
\begin{align*}
S_1 &= [0, x - 1], \\
S_2 &= [x - 1(x)yx - 1] \cup \{0\}, \\
S_3 &= [yx - 1, yx + x - 2] \cup \{0\}, \\
T &= S_1 \cup S_2 \cup S_3.
\end{align*}
\]
Then
(I) \( T + T = [0, 2^x - 2] \),
(II) \( T + B(z, x - 1) = [z, z + yx + x^2 - x - 1] \).

**Proof.** The proof of (I) is divided into five parts:
\[
\begin{align*}
[0, 2x - 2] &= S_1 + S_1, \\
[2x - 1, yx + x - 2] &\subset S_1 + S_2, \\
[yx, yx + 2x - 3] &\subset S_1 + S_3, \\
[yx + x - 2, 2yx + x - 3] &\subset S_2 + S_3, \\
[2yx, 2yx + 2x - 4] &\subset S_3 + S_3,
\end{align*}
\]
the verification of which is straightforward. To prove (II), we first note that
\[
\begin{bmatrix} z, z + x(x - 1) \end{bmatrix} \subset S_1 + B(z, x - 1).
\]
Next, if \( u \in [z + x^2 - x, z + yx - 1] \), then \( u = z + lx - m \), where \( 1 < m < x \) and \( x < l < y \). Hence,
\[
u = ((l - m + 1)x - 1) + (z + (m - 1)(x - 1)) \in S_2 + B(z, x - 1),
\]
and so
\[
\begin{bmatrix} z + x^2 - x, z + yx - 1 \end{bmatrix} \subset S_2 + B(z, x - 1).
\]
Further,
\[ \left[ z + yx, z + yx + x^2 - x - 1 \right] \subset S_3 + B(z, x - 1). \]

If \( w \in T + B(z, x - 1) \), then
\[ z = 0 + z < w < yx + x - 2 + z + (x - 1)^2 = z + yx + x^2 - x - 1, \]
and (II) follows.

**Theorem.** Let \( k > 8 \) be an integer, and put \( x = \left\lfloor k/7 \right\rfloor, y = k - 4x + 3 \). Further, let \( T \) have the same meaning as in the lemma, and let \( u = n_2(T) + 1, v = n_2(T \cup B(u, x - 1)) + 1 \) and
\[ A = (T \setminus \{0\}) \cup B(u, x - 1) \cup B(v, x - 1). \]
Then, \( |A| = k \) and
\[ n_2(A) = 4kx - 14x^2 + 12x - 4 = \frac{2}{7} k^2 + \frac{12}{7} k + O(1). \]

**Proof.** By the lemma,
\[ u = 2yx + 2x - 3, \]
\[ v = u + yx + x^2 - x = 3yx + x^2 + x - 3. \]
Hence,
\[ T + A = \left[ 1, v + yx + x^2 - x - 1 \right] = \left[ 1, 4yx + 2x^2 - 4 \right]. \]

Let \( w = 4yx + 2x^2 - 3 \). Suppose \( w = a + b \) where \( a, b \in A, a < b \). Then \( w < 2b \), and so \( b > 2yx + x^2 - 2 \). Hence, \( b \in B(v, x - 1) \). Similarly, \( w > 2a \), and so \( a < 2yx + x^2 - 1 \). Hence, \( a \notin B(v, x - 1) \). Since \( w \notin T + A, a \in B(u, x - 1) \).
Let
\[ a = (2yx + 2x - 3) + l(x - 1), \quad 0 < l < x - 1, \]
\[ b = (3yx + x^2 + x - 3) + m(x - 1), \quad 0 < m < x - 1. \]
Then,
\[ w = 4yx + 2x^2 - 3 = a + b = 5yx + x^2 + 3x - 6 + (l + m)(x - 1). \]
Hence, \((l + m + 3)(x - 1) + x(y - x) = 0\). However, since \( k > 8 \), \( x - 1 > 0 \) and \( y - x > 0 \), and so we have a contradiction. Therefore,
\[ n_2(A) = 4yx + 2x^2 - 4 = 4kx - 14x^2 + 12x - 4. \]

If we put \( x = k/7 + \theta(k) \), then \( 0 < \theta(k) < 1 \), and \( \theta(k) \) depends only on \( k \) modulo 7. Further,
\[ n_2(A) = \frac{2}{7} k^2 + \frac{12}{7} k + (12\theta - 14\theta^2 - 4). \]

**Remark.** One extremal basis for \( k = 14 \) is of the form \( T \cup B(u, x - 1) \) with \( x = 3 \) and \( y = 8 \). For general \( k \), one optimal choice for \( x \) and \( y \) in this case is \( x = \left\lfloor 3k/16 \right\rfloor \) and \( y = k - 3x + 3 \), which gives
\[ n_2(T \cup B(u, x - 1)) = \frac{9}{32} k^2 + O(k). \]
This is asymptotically weaker than the construction above. A more general construction would be

\[ A = (T \setminus \{0\}) \cup \bigcup_{i=1}^{t} B(u_i, x - 1), \]

where \( u_i = n_2(T) + 1 + (i - 1)(yx + x^2 - x). \) The construction above corresponds to \( t = 2, \) which turns out to give the asymptotically best result.

Universitetet i Bergen
Matematisk Institut
5014 Bergen, Norway