JOURNAL OF ALGORITHMS 9, 343–353 (1988)

Solution to a Linear Diophantine Equation for Nonnegative Integers

HAROLD GREENBERG

Department of Statistics and Computer Information Systems, Baruch College, City University of New York, New York, New York 10010

Received October 6, 1986; accepted September 21, 1987

We solve the 3-variable problem: find integers $x \geq 0$, $y \geq 0$, $z \geq 0$ that satisfy $ax + by + cz = L$ for given integers a, b, c, L, where $1 < a < b < c < L$. The method of solution is related to the one for the Frobenius problem in three variables, which has been solved by Selmer and Beyer and by Rödseth (*J. Reine Angew. Math.* **301** (1978), 161–178). These methods take $O(a)$ steps, in the worst case, to find the Frobenius value. The method here, for the Frobenius value, is shown to be rapid, requiring less than $O(\log a)$ steps. The diophantine equation is then solved with little extra effort to result in an $O(\log a)$ method overall.

1. Introduction

We solve the 3-variable problem: find integers $x \geq 0$, $y \geq 0$, $z \geq 0$ that satisfy

$$ax + by + cz = L \tag{1}$$

or determine that no solution exists (not all L values lead to a solution). The a, b, c, L are given integers with $1 < a < b < c < L$.

In related work, Rödseth [1] solves the Frobenius problem in 3 variables. Given the basis $\{a, b, c\}$, he shows how to find a value L^* with the property that $L = L^*$ does not allow for a solution of (1), but any $L > L^*$ does permit a solution. In finding the Frobenius value, L^*, he uses a form of the Euclidean algorithm with negative remainders to find convergents of continued fractions. The algorithm may take as many as $a - 2$ steps. Rödseth then obtains L^* by finding the minimum of two numbers.

0196-6774/88 $3.00

The first published method for solution of the Frobenius problem, that of Selmer and Beyer [2], uses a form of the Euclidean algorithm with positive remainders to find convergents of continued fractions. The algorithm takes less than $O(\log a)$ steps. Instead of Rödseth's minimum of two numbers, Selmer and Beyer have to use a complicated function that may have as many as $2a$ arguments and, thus, would need $2a$ additional steps to find L^*.

While both Rödseth's and Selmer and Beyer's methods have advantages, they each may take $O(a)$ steps to obtain L^* and, therefore, are not computationally practical. In contrast to their work, we are able to solve the 3-variable problem easily. In solving (1), we follow an approach that leads to the Frobenius value. We use the Euclidean algorithm with positive remainders in the same way as Selmer and Beyer, but are able to finish the method as simply as Rödseth did with his negative remainders. Thus, we develop an efficient method for producing the Frobenius value that takes less than $O(\log a)$ steps. We then go further to efficiently solve (1) when L is given.

Rödseth determines the boundary of a set of pairs (y, z) and finds an extreme point of the set (as the minimum of the two numbers) that leads to the Frobenius value. We shall use this same set of (y, z) values to solve (1).

The Rödseth set is in the shape of a nonconvex hexagon having adjacent perpendicular sides. The set of values, in this fundamental hexagon, will be seen to have the property that if there is any solution to (1), then there is a unique pair (y_0, z_0) in the set that satisfies (1). We find the particular pair (y_0, z_0) that allows for a solution of (1) for a given value of L. We determine whether a solution exists and, if it does, obtain $x = x_0$ to complete the solution.

2. The Fundamental Hexagon

To solve (1), we treat L as an integer variable and define the function t_L as

$$t_L = \min(by + cz) \tag{2}$$

subject to

$$\begin{gathered} by + cz \equiv L \bmod a \\ \text{integer } y \geq 0, \qquad \text{integer } z \geq 0. \end{gathered} \tag{3}$$

For the moment, we assume that $\gcd(a, b) = 1$ so that the congruence $by + cz \equiv L \bmod a$ is solvable for any integer L. If $\gcd(a, b) > 1$, then, as shown below, we will easily convert (1) to a new form where it suffices to assume that $\gcd(a, b) = 1$. Hence, t_L will be defined for all L. Moreover, $t_L = t_{L'}$, where L and L' are in the same residue class modulo a. Thus, we need to solve (2) for $L = 1, 2, \ldots, a-1$. Clearly, L (not restricted to $L \le a-1$) and t_L are in the same residue class. Note that $\min(ax + by + cz)$ subject to (3) is also t_L. Hence, (1) is solvable for a particular value of L if and only if $L \ge t_L$. We have

$$L^* = \max(t_L | L = 1, 2, \ldots, a-1) - a.$$

Each of the values $L = 1, 2, \ldots, a-1$ yields a pair (y, z) that produces t_L. We shall find the region in which the complete set of (y, z) values lies. For a given value of L for (1), we shall next find the y and z values in the set that produce the corresponding t_L value. If $L \ge t_L$, then $x = (L - t_L)/a$; the x, y, z values solve (1). If $L < t_L$, then (1) has no solution.

Given any two integers r_{-1}, r_0, we shall need the continued fraction expansion of r_{-1}/r_0. We use the Euclidean algorithm

$$r_{i-1} = q_{i+1} r_i + r_{i+1}, \qquad 0 < r_{i+1} < r_i,\ i = 0, 1, \ldots, m-1,$$
$$r_{m-1} = q_{m+1} r_m + r_{m+1}, \qquad 0 = r_{m+1} < r_m, \qquad r_m = \gcd(r_{-1}, r_0).$$

The continued fraction convergents P_i/Q_i to r_{-1}/r_0 are

$$\begin{array}{llll} P_{-1} = 0, & P_0 = 1, & P_1 = q_1; & P_{i+1} = q_{i+1} P_i + P_{i-1} \\ Q_{-1} = 1, & Q_0 = 0, & Q_1 = 1; & Q_{i+1} = q_{i+1} Q_i + Q_{i-1} \end{array}$$

with

$$r_0 P_i - r_{-1} Q_i = (-1)^i r_i;$$

hence,

$$r_{-1} Q_i \equiv (-1)^{i-1} r_i \bmod r_0, \qquad r_0 P_i \equiv (-1)^i r_i \bmod r_{-1}. \tag{4}$$

The Euclidean algorithm in this form is known to take less than $O(\log a)$ steps. Refer, for example, to the theory in [3].

First, we try to find v, where $bv \equiv 1 \bmod a$ and $1 < v < a$. For ease of implementation, we extract the needed parts of the continued fraction expansion into algorithmic language. We are able to find v in

ALGORITHM 1. Initialization: $v = 1$, $e = 0$, $f = b - [b/a]a$, $g = a$.

1. Set $e \leftarrow e + [g/f]v$, $g \leftarrow g - [g/f]f$.
   If $g > 1$, go to 2.
   If $g = 1$, set $v \leftarrow a - e$; $bv \equiv 1 \bmod a$. Stop.
   If $g = 0$, $(b/f)v \equiv 1 \bmod (a/f)$. Stop.
2. Set $v \leftarrow v + [f/g]e$, $f \leftarrow f - [f/g]g$.
   If $f > 1$, go to 1.
   If $f = 1$, $bv \equiv 1 \bmod a$. Stop.
   If $f = 0$, set $v \leftarrow (a/g) - e$; $(b/g)v \equiv 1 \bmod (a/g)$. Stop.

From the continued fraction expansion of $b/a = r_{-1}/r_0$, we identify the values of Algorithm 1 as follows: the r_i, $i = 0, 1, \ldots,$ are g and f, alternating; the Q_i, $i = 0, 1, \ldots,$ are e and v, alternating. Algorithm 1 is seen to solve for v in $bv \equiv 1 \bmod a$ since, for the successive values calculated in steps 1 and 2 of Algorithm 1, we have from (4)

THEOREM 1. *$be \equiv -g \bmod a$ and $bv \equiv f \bmod a$.*

We proceed in Algorithm 1 not knowing the value of $\gcd(a, b)$. If f or g is one at some step, then $\gcd(a, b) = 1$; we keep Eq. (1) as is. If f or g is zero, then $\gcd(a, b) = d > 1$ and we must change (1). If $f = 0$, then $g = d$; if $g = 0$, then $f = d$. As seen from the Euclidean algorithm, d divides every f and g value calculated in Algorithm 1, while the e and v values remain the same. Clearly, $f = 0$ or $g = 0$ is reached in less than $O(\log(a/d))$ steps. Moreover, for there to be a solution to (1), z must then satisfy $cz \equiv L \bmod d$. We try to solve for z in this congruence (in the usual way with the Euclidean algorithm). If $\gcd(a, b, c) = \gcd(d, c) = 1$, we obtain generally $z = z_0 + dz'$ for integer unknown $z' \geq 0$, where z_0, $0 < z_0 < d$, satisfies $cz \equiv L \bmod d$; z_0 is obtained in less than $O(\log d)$ additional steps. Substituting for z in (1), we get the reduced basis $\{a/d, b/d, c\}$, a new form of (1) with L replaced by $(L - cz_0)/d$, and a v value with the property that $(b/d)v \equiv 1 \bmod(a/d)$. Suppose the reduced problem $(a/d)x + (b/d)y + cz = M$ has no solution for given value M and the Frobenius value is M^*. Thus, (1) has no solution for $L = dM + cz_0$. The largest possible L with no solution for (1) is given by $M = M^*$ and $z_0 = d - 1$ when $L \equiv -c \bmod d$. Thus, the Frobenius value for the basis $\{a, b, c\}$ is $L^* = dM^* + c(d - 1)$, which was first given by Johnson [4].

If, in solving $cz \equiv L \bmod d$, we discover that $\gcd(d, c) = d' > 1$, then d' must divide L for a solution; we proceed as above to obtain $z = z_0 + (d/d')z'$, where z_0, $0 < z_0 < d/d'$, here satisfies $(c/d')z \equiv (L/d') \bmod(d/d')$. We get the reduced basis $\{a/d, b/d, c/d'\}$ and a new form of (1) with L replaced by $(L - cz_0)/d$. Suppose M^* is the Frobenius value for the reduced basis. As above, we obtain $L^* = dM^* + c(d/d' - 1)$ for the basis $\{a, b, c\}$.

In all the above cases, we achieve the desired v value and possibly a new form of (1) in less that $O(\log a)$ steps. Clearly, it suffices to assume that (1) holds with $\gcd(a, b, c) = 1$ and $\gcd(a, b) = 1$ and the assumption that (3) is solvable for any L is fulfilled.

When v is found from Algorithm 1, we multiply through in (3) by v to obtain $y + s_0 z \equiv vL \bmod a$, where $s_0 = cv - [cv/a]a$. We then have

$$t_L = \min(by + cz) \tag{5}$$

subject to

$$\begin{gathered} y + s_0 z \equiv vL \bmod a, \\ \text{integer } y \geq 0, \qquad \text{integer } z \geq 0. \end{gathered} \tag{6}$$

We shall work now with (6) instead of (3) to find the region for the complete set of (y, z) values that produces t_L.

We use the s_0 value to test whether or not c is independent of a and b. We suppose that $s_0 > 0$, for otherwise c is a multiple of a, and use

THEOREM 2. *The basis element c is dependent on elements a and b if, and only if,* $s_0 \leq c/b$.

Proof. Since $vb \equiv 1 \bmod a$ and $vc \equiv s_0 \bmod a$, then $s_0 b \equiv c \bmod a$. If $s_0 \leq c/b$, then $c = ta + s_0 b$ for some $t \geq 0$. Hence, c is dependent. If $c = ma + nb$, for values $m \geq 0$, $n > 0$, then $nb \equiv c \equiv s_0 b \bmod a$. Since $\gcd(a, b) = 1$, we obtain $s_0 \equiv n \bmod a$; thus, with $s_0 < a$, we get $n = s_0 + ka$, $k \geq 0$, and $c = ma + s_0 b + kab$. Clearly, $s_0 \leq c/b$.

From Theorem 2, the basis $\{a, b, c\}$ is reduced to $\{a, b\}$ when $s_0 \leq c/b$. The solution for t_L in (5) is then given by $z = 0$, $y = vL - [vL/a]a$ resulting in $t_L = by$. If $L \geq t_L$, the solution for (1) is completed with $x = (L - t_L)/a$. It is also instructive to show L^*; if $L \equiv -b \bmod a$, then $\max y = a - 1$. Thus, with $\max t_L = b(a-1)$, we achieve the well-known $L^* = b(a-1) - a$. We can assume $s_0 > c/b$ from now on.

For $s_0 > c/b$, we consider (6) with $y = 0$ and find z, s pairs that satisfy

$$s_0 z \equiv s \bmod a \tag{7}$$

for decreasing values of s starting at s_0, skipping values of s that can be produced when needed. With each $z = z_0$, s pair calculated, $y + s_0 z \equiv s \bmod a$ has as solutions $y = s$, $z = 0$ or $y = 0$, $z = z_0$ and, as will be proven, t_L is produced by one of these solutions. We use

ALGORITHM 2. Initialization: $z = 1$, $e = 0$, $s = s_0$, $g = a$.

1. Set $e \leftarrow e + [g/s]z$, $g \leftarrow g - [g/s]s$.
   If $g > 0$, go to 2. Otherwise, $g = 0$; stop.
2. Set $z \leftarrow z + [s/g]e$, $s \leftarrow s - [s/g]g$.
   If $bs > cz$, go to 1. Otherwise, go to 3.
3. Calculate $\gamma = [(cz - bs)/(ce + bg)] + 1$ and then

$$\begin{aligned} z' &= z - \gamma e, & s' &= s + \gamma g, \\ z^* &= z' + e, & s^* &= s' - g. \end{aligned}$$

From the continued fraction expansion of $a/s_0 = r_{-1}/r_0$, we identify the values of Algorithm 2 as follows: the r_i, $i = -1, 0, \ldots$, are g and s, alternating; the P_i, $i = -1, 0, \ldots$, are e and z, alternating. The algorithm takes less than $O(\log a)$ steps.

For the corresponding successive values calculated in steps 1 and 2 of Algorithm 2, we have from (4)

THEOREM 3. $s_0 e \equiv -g \bmod a$, $s_0 z \equiv s \bmod a$.

After $bs > cz$ in step 2 of the algorithm, it may happen that $g = 0$ on the return to step 1; in that case, the algorithm cannot yield the desired results of step 3 (see [2]). The $g = 0$ case is handled separately. We keep the current z and s values at the point in the algorithm when $g = 0$. Clearly $z \geq 1$; hence, with $bs > cz$, we have $s > c/b > 1$. When $g = 0$, then $s = \gcd(s_0, a)$ and, since $s_0 b \equiv c \bmod a$, we see that $s = \gcd(a, c)$. Thus, $(s_0/s)b \equiv (c/s) \bmod (a/s)$ combined with $(s_0/s)z \equiv 1 \bmod (a/s)$, from Theorem 3, results in $b \equiv (c/s)z \bmod (a/s)$. Therefore, with $b > (c/s)z$, we get $b = (c/s)z + t(a/s)$, $t > 0$; b is dependent on c/s and a/s.

Continuing in the $g = 0$ case, we turn to (1) and notice that, since s divides both a and c, a solution for y must satisfy $by \equiv L \bmod s$. Since $\gcd(a, b, c) = 1$, then $\gcd(b, s) = 1$ and $by \equiv L \bmod s$ is solvable for y. We obtain generally $y = y_0 + y's$, $y' \geq 0$, where y_0, $0 < y_0 < s$, satisfies $by \equiv L \bmod s$. Substituting for y in (1), we obtain the basis $\{a/s, b, c/s\}$ with L replaced by $(L - by_0)/s$. Since b depends on a/s and c/s, we obtain the reduced basis $\{a/s, c/s\}$ and can easily solve $(a/s)x + (c/s)z = (L - by_0)/s$ for x and z, when possible, and complete the solution to (1) with $y = y_0$.

In addition, suppose the reduced problem $(a/s)x + (c/s)z = M$ has no solution for given value M. Hence, (1) has no solution for $L = sM + by_0$. The largest possible L with no solution for (1) is given by $M = (c/s)(a/s - 1) - a/s$ and $y_0 = s - 1$, when $L \equiv -b \bmod s$. Thus, the Frobenius value for the basis $\{a, b, c\}$ is $L^* = c(a/s - 1) - a + (s - 1)b$. The $g = 0$ case is now complete. From now on, we assume that Algorithm 2 ends in step 3.

In Algorithm 2, when $bs > cz$, the z and s values at that point will be seen to have the property that the pair $y = 0$, z produces t_L, while the pair $y = s$, $z = 0$ does not. If $bs \leq cz$, then the pair $y = s$, $z = 0$ will be seen to produce t_L, while the pair $y = 0$, z does not. In addition, using the results of step 3 of Algorithm 2 and Theorem 3, we obtain

THEOREM 4. $s_0 z' \equiv s' \bmod a$, $s_0 z^* \equiv s^* \bmod a$.

On the basis of Theorem 4 and Algorithm 2, we have produced particular solutions of (7) having the property

$$\frac{s^*}{z^*} \leq \frac{c}{b} < \frac{s'}{z'}. \tag{8}$$

These solutions will be seen to have the property that $y = s^*$, $z = 0$ produces t_L, while $y = 0$, $z = z^*$ does not. Also $y = 0$, $z = z'$ produces t_L, while $y = s'$, $z = 0$ does not. Moreover, these solutions enable us to find properties for the actual y, z values that solve (6). These will be sharp bounds on the y, z values that define the region for the complete residue class of solutions modulo a for t_L.

We multiply $s_0 = cv - [cv/a]a$ through by bz^* to obtain $bs^* \equiv cz^* \bmod a$. Similarly, $bs' \equiv cz' \bmod a$. Hence, with (8), there exist integers $R^* \geq 0$ and $R' < 0$, where $cz^* - bs^* = aR^*$ and $cz' - bs' = aR'$. For any L in (1), let $y = y_0$, $z = z_0$ give t_L as the minimum for (5). We have

$$t_L - aR^* = b(y_0 + s^*) + c(z_0 - z^*).$$

If $R^* > 0$, then $z_0 < z^*$; otherwise, t_L is not the minimum. If $R^* = 0$, suppose $z_0 \geq z^*$. The pair $y = y_0 + s^*$, $z = z_0 - z^*$ produces the same t_L with a smaller z value than z_0. Hence, we can take y_0 and z_0, among all those pairs giving the same t_L value, as the pair with minimum z value. Thus, t_L, although having nonunique solutions for y and z, also occurs for $z_0 < z^*$; otherwise, z_0 is not the minimum. We have

$$t_L + aR' = b(y_0 - s') + c(z_0 + z').$$

Since $R' < 0$, then $y_0 < s'$. We also have

$$t_L - a(R^* - R') = b(y_0 + s^* - s') + c(z_0 - z^* + z'),$$

where $R^* - R' > 0$. Hence, if $z_0 \geq z^* - z'$, then $y_0 < s' - s^*$. If $y_0 \geq s' - s^*$, then $z_0 < z^* - z'$.

From Algorithm 2, it is easy to see that $z^*s' - z's^* = se + zg \leftarrow se + zg$. Since $se + zg = a$ initially, we obtain $z^*s' - z's^* = a$; thus, the bounds for (y_0, z_0) are bounds on the complete system of residues modulo a. Hence, $(y_0, z_0) \in A \cup B$, where A and B are disjoint, contiguous sets of pairs of integers given by

$$A = \{(y, z) | 0 \leq y < s' - s^*, 0 \leq z < z^*\}$$
$$B = \{(y, z) | s' - s^* \leq y < s', 0 \leq z < z^* - z'\}$$

so that

$$\{t_L | L = 0, 1, \ldots, a - 1\} = \{by + cz | (y, z) \in A \cup B\} \qquad (9)$$

for the complete system.

Sets A and B are in the form of a hexagon that is pictured in Fig. 1. The (y, z) values in the hexagon are fundamental for solving (1) and for finding the Frobenius value. Note that the parts of the boundary included for $\{t_L\}$ are those given by the heavy lines along the y and z axes. Given a value of L for (1), there result a unique (y, z) pair in the fundamental hexagon and

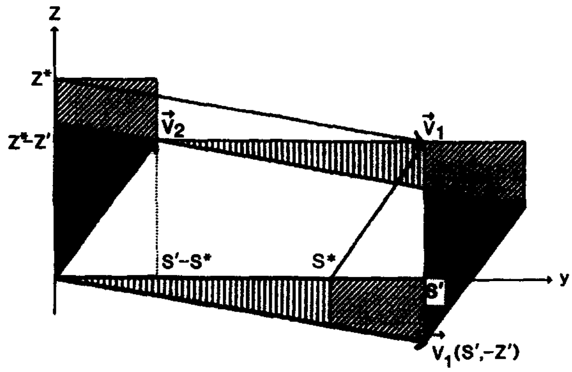

FIG. 1. The fundamental hexagon is given by the points $(y, z) \in A \cup B$, where $A = \{(y, z) | 0 \le y < s' - s^*,\ 0 \le z < z^*\}$ and $B = \{(y, z) | s' - s^* \le y < s',\ 0 \le z < z^* - z'\}$. The fundamental parallelogram is spanned by the vectors $\mathbf{v}_1$ and $\mathbf{v}_2$, which start at the origin and end at the points $(s', -z')$ and $(s' - s^*, z^* - z')$, respectively. The regions of the same shadings are mapped into one another.

a corresponding t_L value. If $L \ge t_L$, then $x = (L - t_L)/a$ and the (y, z) pair solve (1). If $L < t_L$, there is no solution. Moreover, the Frobenius value L^* is obtained from a (y, z) pair occurring at an extreme point of the hexagon. In the next section, we show how to solve (1) and how to find L^*.

Remarks. We use the division algorithm with positive remainders in Algorithm 2. With the use of γ in step 3, we are able to obtain the same s^*, s', z^*, and z' values as found by Rödseth, who obtains them by using the division algorithm with negative remainders, but who uses $a - 2$ division steps in the worst case when $s_0 = a - 1$. Sets A and B found here are the same sets in [1] that are used to find the Frobenius value L^*. We have essentially followed the proof of Rödseth, who uses R^* and R' to show that t_L is given by (9). We will use R^* and R' to produce mappings of the (y, z) plane that enable us to achieve a solution to (1).

Selmer and Beyer also use the division algorithm with positive remainders. They end the algorithm with the same stopping rule as in step 2 of Algorithm 2. They then rely on the complicated M-function of a multiple number of arguments to find L^*; their M-function, moreover, has $2a$ arguments in the worst case when $s_0 = a - 1$. After the stopping rule of step 2, we deviate from Selmer and Beyer by introducing the use of γ; by doing so, we are able to proceed just as simply as Rödseth does after his use of negative remainders.

Algorithm 2 takes less than $1.672 + 1.44 \log a$ steps in the worst case as a gets large; see the theory in [3] on the Euclidean algorithm with positive remainders. In summary, the methods of [1, 2], being $O(a)$ methods, are inefficient for finding L^*. In contrast to [1, 2], our method, being an

$O(\log a)$ method, is efficient for finding L^* and for solving (1), as will be shown.

3 The Solution

We shall now use the fundamental hexagon to solve (1) and to find the Frobenius value L^*. To solve (1), given any value of L, we need to find the corresponding t_L value and the (y, z) pair in the hexagon that produces t_L.

We work geometrically. We take the (y, z)-plane and draw two vectors, $\mathbf{v}_1$ from the origin to the point $(s', -z')$ and $\mathbf{v}_2$ from the origin to the point $(s' - s^*, z^* - z')$. We also draw the parallelogram spanned by the vectors $\mathbf{v}_1$ and $\mathbf{v}_2$. See Fig. 1 for the construction. The number of (y, z) points in the parallelogram is given by $s'(z^* - z') - (s' - s^*)(-z') = a$, the same as in the hexagon. Moreover, all points in the hexagon are in the parallelogram initially, or are mapped into points in the parallelogram by either (a) a $\mathbf{v}_1$ translation, (b) a $\mathbf{v}_1$ translation and then a $-\mathbf{v}_2$ translation, or, (c) a $-\mathbf{v}_2$ translation. These alternatives are readily seen by the corresponding shaded regions of Fig. 1.

In addition, if we translate any (y, z) point in the plane by the vector $\mathbf{v}_1$, the residue $by + cz \bmod a$ is unchanged, since $bs' + c(-z') = -aR'$. Similarly, if we translate (y, z) by $\mathbf{v}_2$, the residue $by + cz \bmod a$ is unchanged, because $b(s' - s^*) + c(z^* - z') = a(R^* - R')$. This result of constancy of the residue is true for points in the hexagon when they are mapped into the parallelogram and for the reverse mapping of points in the parallelogram mapped into the hexagon. Hence, the parallelogram becomes fundamental for the solution to (1), since we will be able to map a point (y, z) in the plane, with desired residue, into the parallelogram and then into the fundamental hexagon.

Now suppose we know L from (1). We take as the original point the solution of $y + s_0 z \equiv vL \bmod a$ given by $y = vL - [vL/a]a$, $z = 0$. If $y < s'$, then $(y, 0)$ is in the fundamental hexagon and $t_L = by$. If $L < t_L$, there is no solution to (1). If $L \geq t_L$, we have the solution $x = (L - t_L)/a$, $y = vL - [vL/a]a$, $z = 0$.

If $y \geq s'$, for $y = vL - [vL/a]a$, let us decompose the vector $(y, 0)$ along $\mathbf{v}_1$ and $\mathbf{v}_2$ as $(y, 0) = \alpha\mathbf{v}_1 + \beta\mathbf{v}_2$. This gives $\alpha = y(z^* - z')/a$, $\beta = yz'/a$. The vector $(y_1, z_1) = (\alpha - [\alpha])\mathbf{v}_1 + (\beta - [\beta])\mathbf{v}_2$ then falls inside the fundamental parallelogram. We have

$$y_1 = y - [y(z^* - z')/a]s' - [yz'/a](s' - s^*),$$
$$z_1 = [y(z^* - z')/a]z' - [yz'/a](z^* - z'),$$

and $by + c \cdot 0 \equiv by_1 + cz_1 \equiv L \bmod a$.

We need the corresponding point (y_0, z_0) inside the fundamental hexagon. If (y_1, z_1) is in the hexagon, we have the desired point already. Otherwise, we map (y_1, z_1) in the fundamental parallelogram into the fundamental hexagon by the reverse of the translations described above. Specifically, the following translations are made to (y_1, z_1):

If $y_1 \geq s'$, we translate $-\mathbf{v}_1$.

If $s^* \leq y_1 < s'$, $z_1 < 0$, we first translate $+\mathbf{v}_2$ and then $-\mathbf{v}_1$.

If $y_1 < s^*$, $z_1 < 0$, we translate $+\mathbf{v}_2$.

The (y_0, z_0) values inside the fundamental hexagon are given by the alternatives

$$\begin{aligned} s' \leq y_1 &: \quad y_0 = y_1 - s',\ z_0 = z_1 + z', \\ s^* \leq y_1 < s',\ z_1 < 0 &: \quad y_0 = y_1 - s^*,\ z_0 = z_1 + z^*, \\ y_1 < s^*,\ z_1 < 0 &: \quad y_0 = y_1 + s' - s^*,\ z_0 = z_1 + z^* - z', \\ y_1 < s',\ z_1 \geq 0 &: \quad y_0 = y_1,\ z_0 = z_1. \end{aligned}$$

We now have $t_L = by_0 + cz_0$, where $by_0 + cz_0 \equiv L \bmod a$. If $L < t_L$, there is no solution to (1). If $L \geq t_L$, we have a solution given by $x_0 = (L - t_L)/a$, y_0, z_0.

The Frobenius value is obtained for $\max(bx + cz)$ at one of the two extreme points of the hexagon produced by $y_1 = s' - 1$, $z_1 = -1$ or $y_1 = s^* - 1$, $z_1 = -1$. Hence,

$$L^* = b(s' - 1) + c(z^* - 1) - \min(bs^*, cz') - a, \tag{10}$$

a result first obtained by Rödseth.

EXAMPLE. Solve $137x + 251y + 256z = 4683$ in nonnegative integers x, y, z. Using Algorithm 1, we obtain $v = 131$. Hence, $s_0 = 108$. We perform the first two steps of Algorithm 2 in Euclidean algorithm format. Initially $e = 0$ and $z = 1$. Thereafter,

$$\begin{aligned} 137 &= 1 \cdot 108 + 29, &\quad g &= 29, &\quad e &= 1 \cdot 1 + 0 = 1, \\ 108 &= 3 \cdot 29 + 21, & s &= 21, & z &= 3 \cdot 1 + 1 = 4, \\ 29 &= 1 \cdot 21 + 8, & g &= 8, & e &= 1 \cdot 4 + 1 = 5, \\ 21 &= 2 \cdot 8 + 5, & s &= 5, & z &= 2 \cdot 5 + 4 = 14. \end{aligned}$$

At this point, with $b = 251$ and $c = 256$, $bs = 251 \cdot 5 \leq 256 \cdot 14 = cz$ for the first time. In step 3, we obtain $\gamma = [(256 \cdot 14 - 251 \cdot 5)/(256 \cdot 5 + 251 \cdot 8)] + 1$. Hence, $\gamma = 1$ and then $z' = 9$, $s' = 13$, $z^* = 14$, $s^* = 5$.

With $L = 4683$, we have $y + 108z \equiv 124 \bmod 137$ and, thus, we take as original point $y = 124$, $z = 0$. Because $y = 124 \geq s' = 13$, we use the vectors $\mathbf{v}_1 = (s', -z') = (13, -9)$, $\mathbf{v}_2 = (s' - s^*, z^* - z') = (8, 5)$; thus, we obtain $[y(z^* - z')/a] = [124 \cdot 5/137] = 4$, $[yz'/a] = [124 \cdot 9/137] = 8$, which results in a point in the fundamental parallelogram given by $y_1 = 8$ and $z_1 = -4$ that is not in the fundamental hexagon.

Continuing, we have $5 = s^* \leq y_1 = 8 < s' = 13$, $z_1 = -4 < 0$; therefore, we obtain $y_0 = y_1 - s^* = 3$, $z_0 = z_1 + z^* = 10$, which results in $t_{4683} = 251 \cdot 3 + 256 \cdot 10 = 3313$; since $L = 4683 \geq 3313 = t_L$, a solution exists and $x_0 = (4683 - 3313)/137 = 10$, $y_0 = 3$, $z_0 = 10$ solves the equation. Note that $L = 4683 \equiv 25 \bmod 137$ and, of course, $251 \cdot 124 + 256 \cdot 0 \equiv 251 \cdot 8 + 256 \cdot (-4) \equiv 251 \cdot 3 + 256 \cdot 10 \equiv 25 \bmod 137$.

For the Frobenius value, we have $\min(bs^*, cz') = \min(1255, 2304) = 1255$; hence, $L^* = 4948$ from (10).

Acknowledgment

The author wishes to thank the referees for their helpful suggestions and insights, which have helped to strengthen the paper.

References

1. Ö. J. Rödseth. On a linear diophantine problem of Frobenius, *J. Reine Angew. Math.* **301** (1978), 171–178.
2. E. S. Selmer and Ö. Beyer, On the linear diophantine problem of Frobenius in three variables, *J. Reine Angew. Math.* **301** (1978), 161–170.
3. D. E. Knuth, "The Art of Computer Programming, Vol. 2, Seminumerical Algorithms," Addison-Wesley, Reading, MA, 1981.
4. S. M. Johnson, A linear diophantine problem, *Canad. J. Math.* **12** (1960), 390–398.