Acta Cybernetica, Tom. 9, Fasc. 4, Szeged, 1990

Decision problems arising from knapsack transformations

Arto Salomaa*

Abstract

The transformations of knapsack vectors by modular multiplications give rise to various intriguing decidability questions. While the most important applications of the resulting algorithms belong to cryptanalysis, the algorithms are certainly of interest on their own right. The basic problem we are considering is whether or not a given vector is obtained from a super-increasing vector by one or several modular multiplications. Various formulations of this problem, as well as various other problems connected with it, will also be discussed. Some interesting problems remain open.

1 Introduction. Connection with cryptography

It is well known that the *knapsack problem* is NP-complete. It is an especially lucid example of an NP- complete problem - easily explainable even for a layman. We are given a vector $B = (b_1, \ldots, b_n)$ consisting of distinct positive integers, as well as another positive integer α. If possible, we have to find a subset of the set $\{b_1, \ldots, b_n\}$ such that the elements in this subset sum up to α. Equivalently, we have to find a column vector C consisting of 0's and 1's such that $BC = \alpha$.

Knapsack vectors B can be used to encrypt messages as follows. A message is divided into blocks C consisting of n bits. Such a block is encrypted as the number BC. Even if one knows the vector B, decryption still amounts to solving the NP-complete knapsack problem. However, decryption is equally difficult for the legal recipient of the message and an illegal eavesdropper, usually called a cryptanalyst. To obtain a public-key cryptosystem, the legal recipient must be given some secret trapdoor information about the publicized vector B. In the earliest public-key cryptosystem, [1], this is done in terms of super-increasing vectors.

A vector $A = (a_1, \ldots, a_n)$ is termed *super-increasing* iff each entry in A exceeds the sum of the preceding entries. The resulting knapsack problems (A, α) are easy and can be solved as follows by just scanning A once from right to left. Thus, we want to determine a bit vector $C = (c_1, \ldots, c_n)^T$ such that $AC = \alpha$. Clearly, $c_n = 1$ iff $\alpha \geq a_n$. (If we do not include a_n in the sum, we cannot reach α because $\sum_{i=1}^{n-1} a_i < a_n$.) Next we compare a_{n-1} with $\alpha - a_n$ or α, depending whether $c_n = 1$ or $c_n = 0$. And so forth, until we reach a_1. As a consequence we observe that every knapsack problem (A, α), where A is super-increasing, possesses at most

*Mathematics Department, University of Turku, SF-20500 Turku, Finland

one solution. This is a property of knapsack vectors referred to as injectivity in the sequel.

A super-increasing knapsack vector A can be scrambled by *modular multiplications*. One chooses a multiplier t and modulus m and reduces, for each i, the product ta_i modulo m. The resulting vector $B = (b_1, \ldots, b_n)$ is publicized. For technical reasons, it is assumed that the greatest common division $(t, m) = 1$ and that $m > \sum_{i=1}^{n} a_i$. Consequently, t possesses an inverse $t^{-1} = u \pmod{m}$. The pair (t, m) constitutes the trapdoor information known to the legal recipient who can form the superincreasing A and decrypt a cryptotext β by using A and the smallest positive remainder α of $u\beta$ modulo m. More details and examples can be found in [2] and [3].

A cryptanalyst has to solve the knapsack problem determined by A' and α' that looks like an arbitrary knapsack problem. However, it only looks like it because very few knapsack vectors are reachable by a modular multiplication from a superincreasing vector. Indeed, Shamir, [5], gave an algorithm working in random polynomial time for finding a super-increasing vector A' (not necessarily the same as the original A) such that the given B results from A' by a modular multiplication. A deterministic polynomial-time algorithm, based on different considerations, was given in [4].

The present paper discusses decision problems and algorithms arising in this set-up. In what follows, the interconnections with cryptography will not any more be important. All definitions and results will be given in terms of ordered n-tuples of positive integers. The exposition is self-contained. A brief outline of the contents of the paper follows.

In Section 2, the basic definitions and notations are given. They include also concepts needed for our technical apparatus. Section 2 contains also some basic results. Our technical tools will be developed in Sections 3 and 4 which include also illustrative examples.

By definition, a vector B is *super-reachable* iff B results from *some* super-increasing vector by a modular multiplication. The two algorithms given in Section 5 decide whether or not the given vector is super-reachable and, in the positive case, produce the corresponding multiplier and modulus. Consequences, related decision problems and modifications are discussed in Section 6. Section 7 deals with a variant, where also permutations of given vectors are taken into account.

A vector is *hyper-reachable* iff it results from some super-increasing vector by a *sequence* of modular multiplications. The wellknown example presented in the basic paper [1] starts with the super-increasing vector $A = (5,10,20)$. Using the multiplier 17 and modulus 47, we obtain $A' = (38,29,11)$. Another multiplier 3 and modulus 89 are applied to A', yielding $B = (25,87,33)$. Thus, B is hyper-reachable. It is easy to show that B cannot be reached from A by one modular multiplication. However, B is still super-reachable. It results, for instance, from $(2,3,66)$ using the multiplier 62 and modulus 99. (This result is obtained by the initial part of the algorithm presented in Section 5.)

It is shown in Section 5 that there are properly hyper-reachable vectors, that is, hyper-reachable vectors which are not super-reachable. Two algorithms for hyper-reachability with a bounded number of modular multiplications are given in Section 8. Section 8 also discusses some other decision problems dealing with hyper-reachability. The concluding Section 9 contains some open problems.

2 Definitions and notations

An ordered n-tuple of distinct positive integers $A = (a_1, \ldots, a_n), n \geq 3$, is referred to as a *knapsack vector* of dimension n. A knapsack vector A is *increasing* (resp. *super-increasing*) iff

$$a_j > a_{j-1} \quad (\text{resp.} \quad a_j > \sum_{i=1}^{j-1} a_i)$$

holds for all $j = 2, \ldots, n$. Clearly, every super-increasing vector is inreasing. For a knapsack vector A, we define

$$\max A = \max\{a_j \mid 1 \leq j \leq n\}.$$

For a positive number x, we denote by $[x]$ the integer part of x, that is, the greatest integer $\leq x$. For integers x and $m \geq 2$, we denote by $(x, \bmod m)$ the *least nonnegative remainder* of x modulo m. Clearly,

$$(x, \bmod m) = x - [x/m] \cdot m$$

This equation will be often written in the form

$$x = (x, \bmod m) + [x/m] \cdot m. \tag{1}$$

We now define two different notions of modular multiplication. Consider a knapsack vector A, an integer $m > \max A$ and a positive integer $t < m$ such that the greatest common divisor $(t, m) = 1$. If $B = (b_1, \ldots, b_n)$ is a vector such that

$$b_i = (ta_i, \bmod m), \text{ for } i = 1, \ldots, n,$$

we write

$$A \underset{(t,m)}{\longrightarrow} B. \tag{2}$$

The integers t and m are referred to as the *multiplier* and the *modulus*, respectively.

Since every knapsack vector A satisfies $\max A \geq 3$, we have always $m \geq 4$. The condition $t < m$ is no loss of generality because if $t > m$, we can subtract $[t/m] \cdot m$ from t without affecting the result of modular multiplication. The equation $t = m$ is not possible because $(t, m) = 1$. The latter condition guarantees also that t possesses an inverse $t^{-1} = u$ such that

$$tu \equiv 1 \,(\bmod m)$$

and $1 \leq u < m$. Since clearly $m > \max B$, we have also

$$B \underset{(u,m)}{\longrightarrow} A. \tag{3}$$

We now come to the other notion of modular multiplication. It means simply the strengthening of our previous notion, where we make the additional assumption that $m > \sum_{i=1}^{n} a_i$. If this condition is satisfied and (2) holds, we write

$$A \underset{(M,t,m)}{\longrightarrow} B. \tag{4}$$

Observe that now a condition analogous to (3) does not necessarily hold because we cannot conclude that $m > \sum_{i=1}^{n} b_i$. Clearly, (4) implies (2) but not vice versa.

The vector B is (A,t,m)-*reachable* (resp. (A,t,m)-*super-reachable*) iff (2) holds and A is increasing (resp. (4) holds and A is super-increasing). B is *super-reachable* iff B is (A,t,m)-super-reachable, for some triple (A,t,m).

Observe that a notion of reachability, defined analogously to that of super-reachability, does not make much sense because apparently every vector would be reachable.

Let $r \geq 1$ be an integer. A knapsack-vector B is *r-hyper-reachable* iff there is a sequence of vectors $A_0, A_1, \ldots, A_r = B$ such that A_0 is super-increasing and, for each $i = 0, \ldots, r-1$, there are t, m such that

$$A_i \underset{(M,t,m)}{\longrightarrow} A_{i+1}.$$

Observe that t and m may be different for different values of i. The notions of 1-hyper-reachability and super-reachability coincide.

In the Introduction the vector $B = (25,87,33)$ was defined in way which showed that it is 2-hyper-reachable. It was also observed that B is, in fact, super-reachable. Similarly, the derivation chains

$$(1,2,4) \underset{(M,5,8)}{\longrightarrow} (5,2,4) \underset{(M,5,12)}{\longrightarrow} (1,10,8) = B_1$$

and

$$(2,4,7) \underset{(M,7,24)}{\longrightarrow} (14,4,1) \underset{(M,3,20)}{\longrightarrow} (2,12,3) \underset{(M,2,23)}{\longrightarrow} (4,1,6) = B_2$$

show that the vectors B_1 and B_2 are 2-hyper-reachable and 3-hyper-reachable, respectively. It will be seen in Example 2 of Section 5 that B_2 is super-reachable, whereas B_1 is not super-reachable.

We would like to emphasize that all our examples only illustrate some points in the theory and are, therefore, "small". Cryptographically interesting vectors would have to be much bigger, say $n = 200$.

A vector is *hyper-reachable* iff it is r-hyper-reachable, for some r. We now define a notion that enables us to construct easily examples of non-hyper-reachable vectors.

A knapsack vector $A = (a_1, \ldots, a_n)$ is *injective* iff the function $f(C) = AC$, defined for n-dimensional bit column vectors C, is injective. Equivalently, the injectivity of A means that, for every α, the knapsack problem determined by A and α possesses at most one solution.

Theorem 1 *Every hyper-reachable vector is injective. Hence, every super-reachable vector is injective.*

Proof. The theorem is a consequence of the following two facts (i) and (ii).

(i) Every super-increasing vector A is injective. Indeed, assume that $AC = AC'$ holds for some bit column vectors C and C'. Then the last components of C and C' coincide, because if one of c_n and c'_n equals 0 and the other equals 1, the numbers AC and AC' cannot be the same. Similarly we conclude by descending induction that, for all $i = 1, \ldots, n$, $c_i = c'_i$. Consequently, $C = C'$.

(ii) The relation (4) preserves injectivity. Assume the contrary: A is injective and there are two distinct vectors C and C' such that $BC = BC'$. We obtain by (4)

$$B \underset{(u,m)}{\longrightarrow} A,$$

where u is the inverse of t modulo m. This implies that $uBC = uBC'$ and, consequently, $AC \equiv AC' \pmod{m}$. Since $m > \sum_{i=1}^{n} a_i$, we conclude that $AC = AC'$, contradicting the injectivity of A. □

Theorem 1 shows that if a knapsack vector is not injective, it cannot be hyper-reachable. For instance, every vector, where some component equals the sum of some other components, is noninjective. We now come to notions that are quite essential in the subsequent proofs.

Consider a knapsack vector $A = (a_1, \ldots, a_n)$, an integer $m > \max A$ and positive integer $t < m$ such that $(t, m) = 1$. The *growing sequence associated with the triple* (A, t, m) is the sequence of triples $(A(k), t, m + kt), k = 0, 1, 2, \ldots$, where

$$A(k) = (a_1 + k \cdot [ta_1/m], \ldots, a_n + k \cdot [ta_n/m]).$$

Thus, the growing sequence associated with (A, t, m) begins with (A, t, m). The terms *multiplier* and *modulus* refer also to the numbers t and $m + kt$ in the triple $(A(k), t, m + kt)$.

For instance, if $A = (1, 2, 3), t = 4, m = 5$, then the growing sequence begins with the triples

$$((1, 2, 3), 4, 5), ((1, 3, 5), 4, 9) \text{ and } ((1, 4, 7), 4, 13).$$

If $A = (1, 4, 7), t = 3, m = 8$, then the growing sequence is

$$((1, 4 + k, 7 + 2k), 3, 8 + 3k), k = 0, 1, 2, \ldots.$$

As the third example, take $A = (1, 5, 6), t = 7, m = 8$. Then the associated growing sequence is

$$((1, 5 + 4k, 6 + 5k), 7, 8 + 7k), k = 0, 1, 2, \ldots.$$

A number $i, 2 \leq i \leq n$, is termed a *violation point* in a knapsack vector A iff $a_i \leq \sum_{j=1}^{i-1} a_j$. Thus, the ith componenet of A violates the requirement of A being super-increasing. If A is increasing, every violation point i in A satisfies $i \geq 3$.

The *goal* of a triple (A, t, m) (defined as above) is the first triple $(A(k), t, m + kt)$ in the growing sequence such that $A(k)$ is super-increasing and $m + kt$ is greater than the sum of the components of $A(k)$, provided such triples exist. Clearly, a triple can be its own goal and some triples have no goal. In particular, if A is not increasing, then (A, t, m) cannot possess a goal. This follows because $a_i > a_{i+1}$ implies that $[ta_i/m] \geq [ta_{i+1}/m]$ and consequently, for all k,

$$a_i + k \cdot [ta_i/m] > a_{i+1} + k \cdot [ta_{i+1}/m].$$

Returning to the three examples considered above, $i = 3$ is a violation point in the initial vector of the first and third example. In the second example the initial vector, as well as all vectors in the growing sequence are super-increasing. The goal in the first example is the third triple in the growing sequence, although in the first triple neither the vector is super-increasing nor the modulus big enough. The sequences in the second and third examples possess no goals. In the second example the modulus will never become big enough. The same holds true for the third example, although the violation point $i = 3$ in the initial vector is "rescued" already by the second vector (1,9,11). (The formal definition of a rescuer will be given below.)

The following more general result concerning super-reachable vectors can be established already at this stage.

Theorem 2 *The vector* $(i, i-1, i-2, \ldots, i-j), i-j \geq 1$, *is super-reachable exactly in case if both* $j = 2$ *and* $i \geq 4$.

Proof. If $j < 2$, the vector is not at all a knapsack vector. If $j > 2$, then the vector is not injective and hence, by Theorem 1, cannot be super-reachable. The same conclusion can be made if $j = 2$ and $i = 3$, because (3,2,1) is not injective. If $j = 2$ and $i > 4$, we have

$$(1,3,5) \underset{(M,i,2i+1)}{\longrightarrow} (i, i-1, i-2). \tag{5}$$

If $j = 2$ and $i = 4$, we have

$$(1,4,7) \underset{(M,4,13)}{\longrightarrow} (4,3,2).$$

In this case (5) does not hold because the modulus is too small but we may use the second triple in the growing sequence. □

We define finally a notion in some sense dual to that of a growing sequence. Let A, B, t, m be such that (2) is satisfied, $t \leq \max B$ and $m > 2 \max B$. Then the *diminishing sequence associated with the triple* (A, t, m) is the sequence of triples

$$(A(-k), t, m - kt), 0 \leq k \leq s,$$

where s is the smallest integer such that $m - st \leq 2 \max B$ and the vectors $A(-k)$ are defined by descending induction as follows. $A(-0) = A$. Assume that $A(k) = (d_1, \ldots, d_n)$ has been defined, and that we still have $m - kt > 2 \max B$. (By the choice of m, this condition holds for $k = 0$.) Then

$$A(-k-1) = (d_1 - [td_1/(m-kt)], \ldots, d_n - [td_n/(m-kt)]).$$

Observe that $t < m - st$. Diminishing sequences are always finite, whereas growing sequences are infinite. However, in the sequel only finite initial segments of growing sequences will be of interest.

3 Fundamental lemmas I

We will now develop the technical tools needed. As will be seen, most of the technical apparatus deals with growing and diminishing sequences. We begin with properties of growing sequences. In Lemmas 1-3, the notation A, t, m and $A(k)$ is the same as in the definition of a growing sequence.

Lemma 1 *If A is increasing or super-increasing, then each vector in the growing sequence associated with (A, t, m) is increasing or super-increasing, respectively.*

Proof. The inequality $a_{i-1} < a_i$ implies the inequality $[ta_{i-1}/m] \leq [ta_i/m]$. Hence, if A is increasing then so is every $A(k)$.

Assume, next, that

$$\sum_{j=1}^{i-1} a_j < a_i.$$

Consequently,

$$\sum_{j=1}^{i-1} [ta_j/m] \leq [(t \sum_{j=1}^{i-1} a_j)/m] \leq [ta_i/m].$$

This implies that, whenever A is super-increasing, then so is every $A(k)$.

Lemma 2 *Assume that* $A \underset{(t,m)}{\longrightarrow} B$ *holds for some* B. *Then* $A(k) \underset{(t,m+kt)}{\longrightarrow} B$ *holds for all* k. *If* B *is* (A,t,m)*-reachable (resp.* (A,t,m)*-super-reachable), then, for all* k, B *is also* $(A(k),t,m+kt)$*-reachable (resp.* $(A(k),t,m+kt)$*-super-reachable).*

Proof. Denoting $B = (b_1,\ldots,b_n)$, we infer by the assumption:

$$b_i = (ta_i, \bmod m), \text{ for } 1 \le i \le n.$$

Clearly, $(t, m+kt) = 1$. By (1), for all k,

$$t(a_i + k\cdot[ta_i/m]) = b_i + [ta_i/m]\cdot m + [ta_i/m]\cdot kt = b_i + [ta_i/m](m+kt).$$

Since by the definition of b_i we have $b_i < m+kt$, we conclude that

$$(t(a_i + k\cdot[ta_i/m]), \bmod (m+kt)) = b_i.$$

Therefore,

$$A(k) \underset{(t,m+kt)}{\longrightarrow} B. \tag{6}$$

By Lemma 1, $A(k)$ is increasing if A is and hence, by (6), the claim concerning reachability follows.

Assume that B is (A,t,m)-super-reachable. By Lemma 1, each $A(k)$ is super-increasing. Moreover,

$$\sum_{i=1}^{n} a_i < m.$$

This implies that

$$\sum_{i=1}^{n}(a_i + k[ta_i/m]) < m + \sum_{i=1}^{n} k\cdot[ta_i/m] \le m + k[t(a_1+\ldots+a_n)/m]$$

$$\le m + k\cdot[t] = m + kt.$$

Consequently, B is $(A(k),t,m+kt)$-super-reachable. □

It is an immediate consequence of Lemma 2 that every super-reachable vector can be obtained from infinitely many super-increasing vectors by modular multiplication with a big enough modulus. The special case, where $[ta_i/m] = 0$ for all i, can be easily handled separately.

We now investigate the question of which triples (A,t,m) possess goals. Recall that every violation point i of A satisfies, by definition,

$$a_i \le \sum_{j=1}^{i-1} a_j. \tag{7}$$

Assume that also

$$[ta_1/m] + \ldots + [ta_{i-1}/m] < [ta_i/m]. \tag{8}$$

(Observe that (7) and (8) are by no means contradictory.) Then the smallest integer x such that

$$\sum_{j=1}^{i-1} a_j + x\sum_{j=1}^{i-1}[ta_j/m] < a_i + x\cdot[ta_i/m] \tag{9}$$

is called the *rescuer* of i. Explicitly,

$$x = [((\sum_{j=1}^{i-1} a_j) - a_i)/([ta_i/m] - \sum_{j=1}^{i-1}[ta_j/m])] + 1.$$

By (7) and (8), x is a positive integer.

If (8) holds for every violation point i, then the *rescuer* of A is defined to be the maximum of the rescuers of all violation points i.

We consider, next, the situation where the modulus is not big enough:

$$m \leq \sum_{i=1}^{n} a_i. \tag{10}$$

Assume that also

$$\sum_{i=1}^{n}[ta_i/m] < t. \tag{11}$$

Then the smallest integer y such that

$$\sum_{i=1}^{n} a_i + y\sum_{i=1}^{n}[ta_i/m] < m + yt \tag{12}$$

is called the *rescuer* of m. Explicitly,

$$y = [((\sum_{i=1}^{n} a_i) - m)/(t - \sum_{i=1}^{n}[ta_i/m])] + 1.$$

We infer by (10) and (11) that the rescuer of m is a positive integer. It is important to notice that if (9) (resp. (12)) holds for some x (resp. y) then it holds for all integers $> x$ (resp. $> y$) as well. This means that if i' is rescued by k', that is, i' is not a violation point in $A(k')$, then i' is not a violation point in any $A(k), k > k'$. Hence, if we have to rescue several numbers (possibly including m), then we may go further in the growing sequence until all of them have been rescued (if ever). For the sake of completeness, we say that 0 is the rescuer of i (resp. m) if (7) resp. (10)) does not hold.

Lemma 3 *A triple (A, t, m) possesses a goal iff (8) holds whenever (7) holds and, moreover, (11) holds in case (10) holds. If these conditions are satisfied, the goal is $(A(k_0), t, m + k_0 t)$, where k_0 is the maximum of the rescuers of A and m.*

Proof. If k_0 is defined as in the statement of the lemma, then $A(k_0)$ is superincreasing (because it has no violation points) and $m + k_0 t$ is greater than the sum of the components of $A(k_0)$. The definition of k_0 guarantees that we obtain the smallest number satisfying these conditions. On the other hand, if some i satisfies (7) but in (8) we have $\geq$ instead of $<$, then i is a violation point in every $A(k)$. Similarly, if (10) holds but (11) does not hold, then for all k,

$$\sum_{i=1}^{n}(a_i + k[ta_i/m]) \geq m + kt.$$

Hence, the modulus is too small in every triple of the growing sequence. □

We now give some illustrations. The examples are given in terms of tables, where A, t, m, B and the goal are listed. Here B is the result of modular multiplication (that is, the vector satisfying (2)). By the second sentence of Lemma 3, the goal gives items showing that B is super-reachable. If no goal exists, we use the abbreviations NR($i = i'$) and NR(m) to mean that a violation point $i = i'$ or too small a modulus m does not possess a rescuer (that is, (8) or (11) is not satisfied). In some examples there may be several such failures. The existence of one failure already shows that there is no goal.

Example 1 We begin with some vectors considered above in Section 2.

A	t	m	B	Goal
(1,2,3)	4	5	(4,3,2)	$k = 2$, (1,4,7),4,13
(1,4,7)	3	8	(3,4,5)	NR (m):0 + 1 + 2 $\geq$ 3
(1,5,6)	7	8	(7,3,2)	$k = 1$ rescuer of $i = 3$, NR (m)
(1,3,5)	4	9	(4,3,2)	$k = 1$, (1,4,7),4,13

We continue with some other illustrations. Different cases concerning which numbers can be rescued will be included.

A	t	m	B	Goal
(1,3,6)	3	7	(3,2,4)	NR (m)
(2,3,4)	5	6	(4,3,2)	NR ($i = 3$), NR (m)
(1,2,3)	5	6	(5,4,3)	$k = 1$, (1,3,5),5,11
(1,5,12)	8	13	(8,1,5)	NR (m)
(1,2,10)	8	15	(8,1,5)	Own goal
(1,8,13,36,57)	87	200	(87,96,131,132,159)	$k = 2$,(1,14,23,66,105),87,374
(1,34,67)	97	100	(97,98,99)	$k = 3$,(1,130,259,),97,391
(1,15,29,44)	93	100	(93,95,97,92)	$k = 2$,(1,41,81,124),93,286
(2,3,5,8)	4	9	(8,3,2,5)	NR ($i = 4$), NR (m), $k = 1$ rescuer of $i = 3$.

4 Fundamental lemmas II

The first lemma in this section deals with an interplay between the multiplier and the modulus. We then discuss properties of diminishing sequences. Finally, growing and diminishing sequences are tied together.

Lemma 4 *Assume that* $max\ B < t < m$ *and*

$$A \underset{(t,m)}{\longrightarrow} B \text{ (resp. } A \underset{(M,t,m)}{\longrightarrow} B) \tag{13}$$

holds. Then the items $A', t' \leq max\ B$ *and* m' *defined below satisfy*

$$A' \underset{(t',m')}{\longrightarrow} B' \text{ (resp. } A' \underset{(M',t',m')}{\longrightarrow} B'). \tag{14}$$

If B *is super-reachable, then* B *is* (A', t', m')*- super-reachable with* $t' \leq max\ B$.

Proof. Assume that in (13) $\max B < t < m$. We define another triple (A_1, t_1, m_1) such that $t_1 < t$ and (14) holds with (A', t', m') replaced by (A_1, t_1, m_1). (14) with $t' \leq \max B$ is then established by repeating this construction as many times as necessary.

Our definition interchanges the multiplier and the modulus as follows:

$$m_1 = t, t_1 = (-m, \bmod t), A_1 = ([ta_1/m], \ldots, [ta_n/m]).$$

Clearly, $t_1 < t$ and $(t_1, m_1) = 1$. By (1) and our assumption (13) we obtain, for $1 \leq i \leq n$,

$$t_1[ta_i/m] \equiv b_i - ta_i \equiv b_i \,(\bmod t).$$

Since $b_i \leq \max B < t$, we may write further

$$(t_1[ta_i/m], \bmod t) = b_i.$$

Corresponding to the "resp."-statement in parentheses in (13) and (14), we still have to show that if m exceeds the sum of the components of A, then t exceeds the sum of the components of A_1. But this is clear. If $m > \sum_{i=1}^{n} a_i$, then also

$$t > \sum_{i=1}^{n} ta_i/m \geq \sum_{i=1}^{n} [ta_i/m].$$

To prove the last sentence of Lemma 4, we show that if A is super-increasing then so is A_1. If A is super-increasing, we have for $2 \leq i \leq n$,

$$\sum_{j=1}^{i-1} ta_j/m > ta_i/m.$$

(The original inequality is multiplied by t/m.) Hence,

$$\sum_{j=1}^{i-1} [ta_j/m] \leq [ta_i/m]. \tag{15}$$

Assume that we have equality in (15). Then

$$\sum_{j=1}^{i-1} m[ta_j/m] = m[ta_i/m].$$

Applying again (1) we obtain

$$\sum_{j=1}^{i-1} (ta_j - b_j) = ta_i - b_i$$

and, hence,

$$b_i - \sum_{j=1}^{i-1} b_j = t(a_i - \sum_{j=1}^{i-1} a_j).$$

The coefficient of t on the right side is positive and, consequently,

$$t \leq b_i - \sum_{j=1}^{i-1} b_j < b_i \leq \max B,$$

which contradicts the assumption $t > \max B$. This implies that we must have strict inequality in (15). Hence, A_1 is super-increasing. □

As an illustration of the technique of Lemma 4, observe first that the vector (7,3,2) is ((7,15,38), 73, 84)-super- reachable. Here the multiplier 73 is much too big. The technique yields, successively, the following triples.

$$((6,13,33),62,73),((5,11,28),51,62),$$

$$((4,9,23),40,51),((3,7,18),29,40),$$

$$((2,5,13),18,29),((1,3,8),7,18).$$

The vector (7,3,2) is super-reachable with respect to all of these triples. In the last triple the multiplier is sufficiently small.

Similarly, the vector (46,45,40,30) is ((4,5,10,20), 49,50)-super-reachable. It is also super-reachable for each of the triples

$$((3,4,9,19),48,49),((2,3,8,18),47,48),((1,2,7,17),46,47).$$

In case of the ((2,5,8,17),32,33)-super-reachability of the vector (31,28,25,16) only one interchange between multiplier and modulus makes the new multiplier sufficiently small. The vector (31,28,25,16) is also ((1,4,7,16),31,32)- super-reachable.

In the following lemma we use the same notation as in the definition of diminishing sequences.

Lemma 5 *Every triple* $(A(-k), t, m - kt), 0 \leq k \leq s$, *in the diminishing sequence associated with the triple* (A, t, m) *satisfies*

$$A(-k) \underset{(t,m-kt)}{\longrightarrow} B. \tag{16}$$

Moreover, if A is increasing, then so is every vector $A(-k), 0 \leq k \leq s$.

Proof. We prove the first sentence by induction on k. For $k = 0$, (16) holds by the definition of the diminishing sequence. Assume that (16) holds and we still have

$$m - kt > 2 \max B. \tag{17}$$

We will show that

$$A(-k-1) \underset{(t,m-(k+1)t)}{\longrightarrow} B. \tag{18}$$

Denote $A(-k) = (d_1, \ldots, d_n)$. Then the ith component of $A(-k-1), 1 \leq i \leq n$, is

$$d_i - [td_i/(m - kt)].$$

Multiplying this by t and using (1) and (16), we obtain

$$td_i - t[td_i/(m - kt)] = b_i + (m - kt)[td_i/(m - kt)] - t[td_i/(m - kt)] =$$

$$= b_i + (m - (k+1)t)[td_i/(m - kt)] \equiv b_i (\bmod m - (k+1)t).$$

By (17) and the assumption $t \leq \max B$ made in the definition of the diminishing sequence,

$$m - (k+1)t > \max B \geq b_i.$$

This implies that

$$(t(d_i - [td_i/(m-kt)]), \bmod m - (k+1)t) = b_i$$

and, consequently, (18) holds.

The second sentence of Lemma 5 is established also by induction on k. Assume that $A = A(-0)$ is increasing. We make the inductive hypothesis that $A(-k) = (d_1, \ldots, d_n)$ is increasing and (17) holds. Denote $A(-k-1) = (e_1, \ldots, e_n)$. Because of (17) and the inequality $t \leq \max B$, we obtain

$$m - kt > 2t. \tag{19}$$

Consider now an arbitrary i, $1 \leq i \leq n-1$. Since $A(-k)$ is increasing,

$$d_{i+1} = d_i + \alpha \text{ for some } \alpha \geq 1.$$

Assume first that $\alpha > 1$. Then

$$e_{i+1} = d_i + \alpha - [t(d_i + \alpha)/m - kt)] \geq d_i + \alpha - (1 + [td_i/(m-kt)] + [t\alpha/(m-kt)])$$

$$= e_i + (\alpha - 1) - [t\alpha/(m-kt)] > e_i.$$

Here the first inequality follows because always $[x+y] \leq [x] + [y] + 1$, and the second because, by (19),

$$[t\alpha/(m-kt)] \leq t\alpha/(m-kt) < \alpha/2.$$

Assume, secondly, that $\alpha = 1$. In this case $[t\alpha/(m-kt)] = 0$. If

$$[t(d_i+1)/(m-kt)] = [td_i/(m-kt)],$$

we obtain $e_{i+1} > e_i$. Hence, suppose that

$$[t(d_i+1)/(m-kt)] = [td_i/(m-kt)] + 1. \tag{20}$$

(By the above estimate for e_{i+1} there are no other possibilities. (20) would imply that $e_{i+1} = e_i$.) Denote the right side of (20) by $\beta + 1$. Hence,

$$(m-kt)\beta \leq td_i < (m-kt)(\beta+1) \leq t(d_i+1).$$

Assume that $td_i < (m-kt)(\beta + \frac{1}{2})$. Hence, by (19),

$$td_i + t < (m-kt)(\beta + \frac{1}{2}) + t = (m-kt)(\beta+1) + t - \frac{m-kt}{2} < (m-kt)(\beta+1),$$

a contradiction. Hence, $td_i \geq (m-kt)(\beta + \frac{1}{2})$. But now, by (16),

$$b_i = td_i - \beta(m-kt) \geq (m-kt)/2.$$

This implies that $m - kt \leq 2b_i \leq 2 \max B$, contradicting (17). This shows that (20) cannot hold. □

It is important to note that certain properties preserved by the growing sequences are not preserved by the diminishing sequences. A may be super-increasing although the other vectors in the diminishing sequence are not. For instance, choose

$$A = (1, 14, 23, 66, 105), t = 87, m = 374,$$

implying that $B = (87, 96, 131, 132, 159)$ and, hence $t \leq \max B$ and $m > 2 \max B$. Now

$$A(-1) = (1, 11, 18, 51, 81),$$

which is not super-increasing. Similarly, we see that

$$(1, 4, 7) \underset{(M,4,13)}{\longrightarrow} (4, 3, 2)$$

but when we go to the first triple in the diminishing sequence, we observe that not

$$(1, 3, 5) \underset{(M,4,9)}{\longrightarrow} (4, 3, 2)$$

because $9 = 1 + 3 + 5$. Thus, the M-relation is not preserved. In the second triple ((1,2,3),4,5,) of the same diminishing sequence neither is the M-relation satisfied nor is the vector super-increasing. Such negative results are natural in view of the following lemma and reflect the fact that some properties are rescued from a certain point on in the growing sequence. The same properties are lost at this point in the diminishing sequence.

In the statement of the following lemma, the notation A, t, m, s, B is the same as in the definition of the diminishing sequence.

Lemma 6 *Assume that* $(A(-k), t, m - kt), 0 \leq k \leq s$, *is the diminishing sequence associated with the triple* (A, t, m). *Denote* $A(-s) = C$. *Consider the initial segment*

$$(C(k), t, m - st + kt), 0 \leq k \leq s,$$

of the growing sequence associated with the triple $(C = C(0), t, m - st)$. *Then, for each* k *such that* $0 \leq k \leq s$,

$$C(k) = A(-(s - k)). \tag{21}$$

Proof. Our intention is to use induction on k. For this purpose, it is useful to denote

$$A(-(s - k)) = (a_1^k, \ldots, a_n^k), C(k) = (c_1^k, \ldots, c_n^k),$$

for $0 \leq k \leq s$. Clearly, $A(-(s - s)) = (a_1, \ldots, a_n) = A$. We consider an arbitrary k and i in their respective ranges. To simplify notation, we write $a^k = a_i^k$ and $c^k = c_i^k$. By the definitions of growing and diminishing sequences,

$$c^{k+1} = c^k + [tc^0/\alpha] \text{ and } a^{k+1} = a^k + [ta^{k+1}/(\alpha + (k + 1)t)], \tag{22}$$

where $\alpha = m - st$. We have to establish $c^k = a^k$, for all k with $0 \leq k \leq s$, in order to establish (21). By the choice of $C(0)$, we have $a^0 = c^0$. Using (22), we show that $c^1 = a^1$. Thus, we have to prove that

$$[ta/\alpha] = [ta^1/(\alpha + t)], \tag{23}$$

where we denote $c^0 = a^0 = a$. By Lemmas 2 and 5,

$$ta^1 \equiv tc^1 \text{ and, hence, } a^1 \equiv c^1 \pmod{\alpha + t}.$$

Because $a^0 = c^0$, we infer that

$$[ta/\alpha] \equiv [ta^1/(\alpha+t)] \,(\mathrm{mod}\, \alpha+t). \tag{24}$$

(24) can hold without (23) holding only in case that the absolute value of the difference between the two bracket expressions is a positive multiple of $\alpha + t$. We prove that this is impossible by showing that both of the bracket expressions (which clearly are nonnegative) are less than $\alpha + t$. Since $\alpha > \max C(0) = \max A(-s) \geq a$, we obtain

$$[ta/\alpha] < t < \alpha + t.$$

The bracket expression on the right side of (24) is estimated by repeated use of the principle $[x] \leq x$, yielding when we denote $t/(\alpha+t) = x$

$$[ta^1/(\alpha+t)] \leq ta^1/(\alpha+t) = \frac{t}{\alpha+t}(a + [ta^1/(\alpha+t)])$$

$$\leq \frac{t}{\alpha+t}(a + \frac{t}{\alpha+t}(a + [ta^1/(\alpha+t)])) \ldots$$

$$\leq a(x + x^2 + \ldots + x^p) + x^{p+1}a^1$$

$$\leq a/(1-x) + x^{p+1}a^1 = a + at/\alpha + x^{p+1}a^1$$

$$< \alpha + t + x^{p+1}a^1.$$

This holds for arbitrarily large p, which means that the term $x^{p+1}a^1$ can be made arbitrarily small. Consequently,

$$[ta^1/(\alpha+t)] < \alpha + t.$$

By (24), (23) holds. We have shown that $a^1 = c^1$.

The inductive step from $a^k = c^k$ to $a^{k+1} = c^{k+1}$ is now very easy. We consider only the initial segment of the diminishing sequence to the triple $(A(-(s-k)), t, m-(s-k)t)$. We start the growing sequence from this triple. Also now we have to establish (23), where now $\alpha = m - (s-k)t, a = a^k = c^k$ and $a^1 = a^{k+1}$. The proof is exactly the same as above. This completes the induction and, hence, (21) holds.

5 Super-reachability

We are now in the position to establish one of our main results.

Theorem 3 *A knapsack vector B is super-reachable iff B is (A, t, m)-reachable, where $t \leq \max B, m \leq 2\max B$ and the triple (A, t, m) possesses a goal.*

Proof. We already have developed all the necessary technical apparatus. The "if"-part follows by Lemma 2 and the definition of the goal. Lemma 3 gives a simple method for deciding whether or not a given triple possesses a goal.

For the "only if"-part, assume that B is super-reachable. By Lemma 4, B is (A, t, m)-super-reachable with $t \leq \max B$. If $m \leq 2\max B$, we are finished. Otherwise, we form the diminishing sequence

$$(A(-k), t, m - kt), 0 \leq k \leq s,$$

where $m - st \leq 2 \max B$. Since A is increasing, we conclude, by Lemma 5, that B is $(A(-s), t, m - st)$-reachable and, by Lemma 6, that the triple $(A(-s), t, m - st)$ possesses a goal. □

The algorithm due to Theorem 3 can be described as follows. Given B, choose $\max B < m \leq 2 \max B$ and $u < m$ with $(u, m) = 1$. Check whether the vector A satisfying

$$B \underset{(u,m)}{\longrightarrow} A$$

is increasing and $u^{-1} = t \leq \max B$. If not, choose another pair (u, m). Else check whether the triple (A, t, m) possesses a goal. If not, choose another pair (u, m). Otherwise, B is super-increasing. The goal also gives a super-increasing vector, multiplier and modulus showing this.

The time complexity of the algorithm is estimated in [4] . Complexity in terms of $\max B$ is at most cubic. Complexity in terms of n depends on the upper bound for $\max B$ in terms of n. Such upper bounds are given, for instance, in [1] and [5]. They are always arbitrary and leave out most of the instances, whereas the algorithm of Theorem 3 works independently of any bounds, for $\max B$. Reductions in the estimates can possibly be made by a more detailed analysis of the number of successful pairs (u, m).

Example 2 We now give some illustrations of the algorithm of Theorem 3. Again, for the sake of readability, the illustrations are very small in size. We consider first the vectors (1,10,8) and (4,1,6) shown 2- and 3-hyper-reachable in Section 2. Consider first the vector (4,1,6). The pairs (u, m) to be investigated are listed in the following table.

m	12	11	10	9	8	7
u	5,7,11	2,3,...,10	3,7,9	2,4,5,7,8	3,5,7	2,3,4,5,6

The next table shows the actual application of the algorithm. The leftmost column lists all the pairs (u, m) which might lead to success, that is, $u^{-1} = t \leq \max B = 6$ (inverse is taken modulo m), and the vector A obtained from $B = (4, 1, 6)$ by modular multiplication due to (u, m) is increasing. The items t and A are listed in the next two columns. If A is not super- increasing or $m \leq a_1 + a_2 + a_3$, we investigate whether or not the violation point i' (here only $i' = 3$ is possible) and the modulus m can be rescued. If they can, the last column indicates the value of k for which the goal is reached in the growing sequence associated with the triple (A, t, m). The last column also indicates the three items of the goal. If at least one of the numbers cannot be rescued, we use the abbreviations NR($i = i'$) and NR(m) as before.

u, m	$t = u^{-1}$	A	Goal
3,11	4	(1,3,7)	$k = 1$, (1,4,9),4,15
9,11	5	(3,9,10)	NR($i = 3$), NR(m)
5,8	5	(4,5,6)	NR($i = 3$), NR(m)
2,7	4	(1,2,5)	$k = 2$, (1,4,9),4,15

It is interesting to note that in both cases leading to success we obtain the same triple ((1,4,9),4,15). Thus, this triple can be visualized as the minimal or prime triple for which (4,1,6) is super-reachable. More specifically, whenever (4,1,6) is (A, t, m)-super-reachable, then $t \geq 4$ and $m \geq 15$. This follows because the algorithm would produce any smaller values of t and m. Of course, m can be made

arbitrarily large in the growing sequence. Also t can be made larger by applying an argument similar to that used in Lemma 4 in the reverse order.

The vector $(4,1,6) = B$ shows also that it is in general not sufficient to investigate candidates $m \leq 2 \max B$, without taking into account the growing sequence. If this would be done for (4,1,6), we would never find the solution. However, it is possible to obtain the following general result.

Theorem 4 *A knapsack vector B is super-reachable iff B is (A, t, m')-super-reachable where $t \leq \max B$ and $m' \leq 2 \max B(1 + \max B)$.*

Proof. The upper bound for t is obtained by Lemma 4 exactly as before. To obtain an upper bound for the modulus, we have to deduce an upper bound for the moduli $m + kt$ in the initial segment of the growing sequence consisting of triples up to the goal. We know that $t \leq \max B$ and $m \leq 2 \max B$. Since a goal is reached, the difference between the sum of the components of the vector and the modulus decreases at least by one in every step from a triple to the next triple in the growing sequence. This holds true also as regards the difference defined by any violation point. The goal is reached when all of these differences are negative. Hence, the goal is reached latest in m (= the original modulus) steps, implying that $k \leq 2 \max B$. Consequently, $m' \leq 2 \max B + (2 \max B) \max B$. □

The statement of Theorem 4 is simpler than that of Theorem 3. However, the resulting algorithm is considerably less efficient, as shown even by examples of small size.

Example 3 Consider now the vector (1,10,8), shown 2-hyper-reachable in Section 2. We have to consider moduli $m \leq 20$. For each m, we must have $u < m$ and $(u, m) = 1$. The following table of pairs (u, m) that may lead to success is obtained exactly as in Example 2.

u, m	$t = u^{-1}$	A	Goal
7,20	3	(7,10,16)	NR $(i = 3)$, NR (m)
9,20	9	(9,10,12)	NR $(i = 3)$, NR (m)
2,17	9	(2,3,16)	NR (m)
6,17	3	(6,9,14)	NR $(i = 3)$, NR (m)
5,14	3	(5,8,12)	NR $(i = 3)$, NR (m)
3,13	9	(3,4,11)	NR (m)
4,11	3	(4,7,10)	NR $(i = 3)$, NR (m)
5,11	9	(5,6,7)	NR $(i = 3)$, NR (m)

We conclude that (1,10,8) is not super-reachable. Hence, we have established the following result.

Theorem 5 *There are 2-hyper-reachable knapsack vectors that are not super-reachable.*

It is an open problem whether or not r-hyper-reachable vectors form a strictly increasing hierarchy with r increasing. Other examples of strictly 2-hyper-reachable vectors are easy to construct.

Example 4 We now give the table for each permutation of the vector (2,3,4). In each case only values $m \leq 8$ have to be considered.

(2,3,4): No candidates (u, m)

	u, m	$t = u^{-1}$	A	Goal
(3,4,2):	3,8	3	(1,4,6)	NR (m)
	2,5	3	(1,3,4)	$k = 1$ rescues $i = 3$, NR (m)
(4,2,3):	2,7	4	(1,4,6)	NR (m)
(2,4,3):	4,7	2	(1,2,5)	$k = 2$, (1,2,7),2,11
	3,5	2	(1,2,4)	$k = 3$, (1,2,7), 2,11
(3,2,4):	(5,7)	3	(1,3,6)	NR (m)
(4,3,2)	(4,5)	4	(1,2,3)	$k = 2$, (1,4,7),4,13

The study of (4,3,2) is interesting because it shows that we cannot reject non-injective candidates A in spite of Theorem 1. This is due to the fact that injectivity can be gained later on in the growing sequence.

We now investigate similarly all permutations of the vector (1,2,4).

(1,2,4): super-increasing

	u, m	$t = u^{-1}$	A	Goal
(1,4,2):	3,8	3	(3,4,6)	NR ($i = 3$), NR (m)
	2,5	3	(2,3,4)	NR ($i = 3$), NR (m)
(2,1,4):	5,7	3	(3,5,6)	NR ($i = 3$), NR (m)
(2,4,1):	4,7	2	(1,2,4)	$k = 1$, (1,2,5),2,9
	3,5	2	(1,2,3)	$k = 2$, (1,2,5),2,9
(4,1,2):	2,7	4	(1,2,4)	$k = 1$, (1,3,6),4,11
(4,2,1):	4,5	4	(1,3,4)	$k = 1$ rescues $i = 3$, NR (m)

Summarizing we obtain the following result.

Theorem 6 *Consider knapsack vectors with all components ≤ 4. Exactly the following ones are super-reachable:*

$$(2,4,3), (4,3,2), (1,2,4), (2,4,1), (4,1,2)$$

Proof. By Theorem 1, no permutation of any of the vectors (1,3,4), (1,2,3), (1,2,3,4) can be super-reachable. The remaining cases were classified in Example 4. □

6 Consequences and modifications

Several other decidability results can be obtained using our basic technique of growing and diminishing sequences. We mention a minimization result concerning the multiplier and the modulus.

Theorem 7 *Assume that B is super-reachable. Then the smallest m (resp. the smallest t) such that B is (A, t, m)-super-reachable for some A and t (resp. A and m) is effectively computable.*

Proof. By Theorem 3 or Theorem 4, some triple (A, t, m) is obtained. A straightforward way of minimizing the modulus would be a systematic search through all values $m' < m$. For each m', it suffices to test the finitely many triples (A', t', m'), where $t' < m'$ and the sum of the components of the super-increasing A' is less than m'. However, a much more efficient algorithm (running in time at most cubic

in terms of max B) is obtained by Theorem 3: the smallest modulus can be found from the triples produced by Theorem 3. The same holds true as regards the smallest multiplier t. We have presented several examples, where it is necessary to go into the growing sequence in order to find the smallest modulus, as well as examples, where the smallest multiplier is considerably less than max B. □

A consequence of Theorem 3, apparent also in the examples above, is that a vector B is not super-reachable if there are no candidates (A, t, m), where A is increasing, $t = u^{-1} \leq \max B, m \leq 2 \max B$ and A results from B by modular multiplication using u and m. The special case, where B itself is increasing but not super-increasing, is interesting. Considering small examples, one is tempted to conjecture that growing sequences do not at all come into use, that is, one may restrict the attention to vectors A reachable from B by modular multiplication using u and m. However, the following example shows that this conjecture is false. Choose $B = (87, 96, 131, 132, 159), m = 200$ and $n = 23$. Then $t = 87$ and $A = (1, 8, 13, 36, 57)$, which is not super-increasing. However, the triple (A, t, m) possesses the goal

$$((1, 14, 23, 66, 105), 87, 374),$$

reached for $k = 2$. Here $374 > 2 \max B$.

The following result can be obtained along these lines.

Theorem 8 *If $B = (b_1, b_2, b_3)$ is increasing and super-reachable, then B is (A, t, m)-super-reachable, for some $t \leq \max B, m \leq 2 \max B$.*

Proof. There must be an increasing A such that

$$B \underset{(u,m)}{\longrightarrow} A$$

where $m \leq 2 \max B, t = u^{-1} \leq \max B$. Suppose that no such A is super-increasing. Consider an arbitrary $A = (a_1, a_2, a_3)$. Hence, 3 is a violation point: $a_3 \leq a_1 + a_2$. This implies that $ta_3 \leq ta_1 + ta_2$. Denote $ta_i = b_i + \alpha_i m, i = 1, 2, 3$. (Hence, $[ta_i/m] = \alpha_i$.) We obtain

$$b_3 + \alpha_3 m \leq b_1 + b_2 + (\alpha_1 + \alpha_2)m.$$

If now $\alpha_3 > \alpha_1 + \alpha_2$, we obtain further

$$b_3 + m \leq b_3 + (\alpha_3 - \alpha_1 - \alpha_2)m \leq b_1 + b_2 + (\alpha_1 + \alpha_2 - \alpha_3)m + (\alpha_3 - \alpha_1 - \alpha_2)m =$$

$$= b_1 + b_2 < b_1 + m.$$

Consequently, $b_3 < b_1$, which contradicts the assumption of B being increasing. This implies that $\alpha_3 \leq \alpha_1 + \alpha_2$, which shows that the triple (A, t, m) possesses no goal. Since A was arbitrary, we conclude that B is not super-reachable, contrary to the assumption. □

It was seen above that the result of Theorem 8 does not hold true if the number n of the components of the vectors equals 5. It is an open problem whether or not the result holds for $n = 4$.

Our final example in this section is of a somewhat different nature.

Example 5 Shamir's algorithm (see [5] or [2]) is based on the assumption that the given vector B is super-reachable. The algorithm usually produces an interval such that, whenever the number u/m written in lowest terms lies in this interval, then B is (A, t, m)-super-reachable, $t = u^{-1}$ and

$$B \underset{(u,m)}{\longrightarrow} A.$$

Without explaining any details of Shamir's algorithm, we show by a couple of examples how one can go back to the algorithm of Theorem 3.

Consider the vector $B = (7, 3, 2)$. We get an open interval $(5/7, 3/4)$. The number $8/11$ in this interval yields $A = (1, 2, 5)$. This sows that B is $(A, 7, 11)$-super-reachable. Here both 7 and 11 are within the limits of Theorem 3 and, thus, the result is obtained by the "first part" of our algorithm, where one does not use growing sequences. The number $41/56$ in this interval yields $A = (7, 11, 26)$. Since 41 is its own inverse, B is ((7,11,26),41,56)-super-reachable. The multiplier 41 is, however, too big to be reached by the algorithm of Theorem 3. Using Lemma 4, we get successively the following triples:

$$((5, 8, 19), 26, 41), ((3, 5, 12), 11, 26), ((1, 2, 5), 7, 11).$$

Here the last triple (in fact, the same as the one obtained for $8/11$) falls within the size limits of Theorem 3.

As regards the number $61/84$ from the interval in question, the procedure is slightly different. The inverse of 61 is 73 and, hence, B is ((7,15,38),73,84)-super-reachable. Again, the multiplier is too big. Lemma 4 yields, in succession, the triples

$$((6, 13, 33), 62, 73), ((5, 11, 28), 51, 62),$$

$$((4, 9, 23), 40, 51), ((3, 7, 18), 29, 40),$$

$$((2, 5, 13), 18, 29), ((1, 3, 8), 7, 18).$$

In the last triple the multiplier $t = 7$ satisfies $t \leq \max B$. In fact, we already carried out the computation this far after Lemma 4. However, $m > 2 \max B$ and cannot be obtained in the first part of the algorithm. Taking one step in the diminishing sequence we obtain our old friend ((1,2,5),7,11), which completes our argument.

In the example considered in [2],

$$B = (43, 129, 215, 473, 903, 302, 561, 1165, 697, 1523)$$

is the vector to be analyzed. Shamir's algorithm produces the interval $(1/43, 36/1547)$. Choosing $u = 37$ and $m = 1590$, we get the number $37/1590$ in this interval, as well as the vector

$$A = (1, 3, 5, 11, 21, 44, 87, 175, 349, 701)$$

which is super-increasing. Now $u^{-1} = t = 43$ and, thus, both t and u lie within the bounds of the first part of the algorithm of Theorem 3. In fact, the solution obtained equals the one used by the cryptosystem designer in [2] .

Consider next $u = 72$ and $m = 3095$. We get the vector

$$A = (1, 3, 5, 11, 21, 79, 157, 315, 664, 1331).$$

Also now $t = 43$ but $m > 2 \max B$. When we go two steps back in the diminishing sequence, we get the triple

$$((1, 3, 5, 11, 21, 77, 153, 307, 646, 1295), 43, 3009).$$

Now also m is within the size limits.

7 Permutations

For a cryptanalyst it is certainly sufficient to find a permutation of a publicized vector B that is super-reachable. When such a permutation is known, cryptanalysis works as before - only the inverse permutation has to be applied to the plaintext bit vectors.

Let us call a vector B *permutation-super-reachable* iff some permutation of B is super-reachable. For instance, it was seen in Example 3 that (1,10,8) is not super-reachable. Clearly, it is permutation-super-reachable. By our theory it is easy to see that every injective (a_1, a_2, a_3) is permutation-super-reachable. The following result is established exactly as Theorem 1.

Theorem 9 *Every permutation-super-reachable vector is injective.*

Permutations were investigated already in Example 4. The following example is of a similar nature.

Example 6 We use the same notation as in Example 4 to classify the permutations of (3,4,5).

	u, m	$t = u^{-1}$	A	Goal
(3,4,5):	3,8	3	(1,4,7)	NR (m)
(3,5,4):	7,10	3	(1,5,8)	NR (m)
	5,7	3	(1,4,6)	NR (m)
(4,3,5):	5,9	2	(2,6,7)	NR $(i = 3)$, NR (m)
	7,9	4	(1,3,8)	NR (m)
	4,7	2	(2,5,6)	NR $(i = 3)$, NR (m)
(4,5,3):	2.7	4	(1,3,6)	NR (m)
(5,3,4):	2,9	5	(1,6,8)	NR (m)
	3,7	5	(1,2,5)	$k = 2$, (1,4,11),5,17
(5,4,3):	5,8	5	(1,4,7)	NR (m)
	5,6	5	(1,2,3)	$k = 1$, (1,3,5),5, 11

Thus, only (5,3,4) and (5,4,3) are super-reachable.

8 Hyper-reachability

Various decidability results and polynomial-time algorithms concerning hyper-reachability can be obtained using the techniques developed above. We mention here some such results. All of them concern r-hyper-reachability for a fixed or bounded r. This is basically due to the fact that a characterization of hyper-reachability is missing. Do the r-hyper-reachable sets of vectors form an infinite hierarchy (when r is growing)? It is conceivable that, for some target vectors B, the "derivation chain" is arbitrarily long with irregular fluctuations in the sizes of the intermediate vectors and moduli.

The following Theorems 10-12 correspond to Theorems 3,4 and 7, respectively.

Theorem 10 *It is decidable of a given knapsack vector B and positive integer r whether or not B is r-hyper-reachable.*

Proof. Consider first the case $r = 2$. Then B is 2-hyper-reachable iff there exist t, m, t', m', C and a super-increasing A such that

$$A \underset{(M,t',m')}{\longrightarrow} C \underset{(M,t,m)}{\longrightarrow} B. \tag{25}$$

The method of Theorem 3 is now applicable with the exception that we cannot use Lemma 4 in connection with C. The construction leading from (13) to (14) is valid but does not necessarily preserve the super-reachability of the vectors involved. In (25) C is super-reachable, whereas the vector obtained form C by the construction of Lemma 4 might not be super-reachable. However, the strict separation $t \leq \max B, m > 2 \max B$ is needed only in (19) to prove that diminishing sequences preserve the property of being increasing. We do not need this property in connection with C. The proofs of the Lemmas 2,3 (where the goal is defined only for m), 5 (where the requirement $t \leq \max B$ is omitted from the definition of a diminishing sequence) and 6 remain valid.

We proceed as follows. For arbitrary $u < m \leq 2 \max B$, we form the vector $E = (e_1, \ldots, e_n)$ such that

$$B \underset{(u,m)}{\longrightarrow} E.$$

(E need not be increasing and not necessarily $u^{-1} = t \leq \max B$.) If $m > \sum_{i=1}^{n} e_i$, the vector E qualifies as a candidate for C. Otherwise, we test by Lemma 3 whether or not the modulus can be rescued in the growing sequence associated with (E, t, m). If it can, then the resulting vector E' qualifies as a candidate for C. Then all candidates are obtained according to Theorem 3. If originally in (25) $m > 2 \max B$, a modulus of the right size is obtained in the diminishing sequence. The result need not be super-reachable because the original C is recovered in the growing sequence.

The case of a general r is now obvious by induction. Assuming the validity of the assertion for a fixed r, to test $(r+1)$-hyper-reachability we first form intermediate candidates exactly as above. The only difference is that we are now dealing with candidates for r-hyper-reachability rather than for super-reachability.

Theorem 11 *A vector B is r-hyper-reachable iff it is r-hyper-reachable for a chain of modular multiplications, where each multiplier and modulus is less than $(\max B)^{3^r}$.*

Proof. We replace the upper bound $2 \max B(1+\max B)$ in Theorem 4 by the much ruder upper bound $(\max B)^3$. Theorem 11 now follows because always the components of the vectors are smaller than the modulus.

Theorem 12 *Assume that B is r-hyper-reachable. Then the smallest m such that the r-hyper-reachability of B can be shown using only moduli $\leq m$ is effectively computable.*

Proof. One can use either the algorithm described in Theorem 10 or the more simply stated but less efficient algorithm due to Theorem 11. □

The part of Theorem 7 dealing with multipliers cannot be generalized by this technique. This is due to the fact that Lemma 4 cannot be applied in case of r-hyper-reachability. It is conceivable that a smaller t will work with a big increase in m.

It is clear that, for a fixed r, the algorithms due to Theorems 10-12 work in polynomial time (with respect to $\max B$), where the degree of the polynomial depends on r.

Our final result is an immediate consequence of Theorem 10.

Theorem 13 *If B is known to be hyper-reachable, then the smallest r such that B is r-hyper-reachable can be effectively computed.*

9 Conclusion

The techniques developed here seem to be applicable to a great variety of topics dealing with knapsacks. We have mentioned above many open problems. In our estimation, the following are the most important among them. (i) Present criteria, other than Theorem 1, for constructing classes of vectors that are not super-reachable (resp. not r-hyper-reachable, not hyper-reachable). (ii) Do the r-hyper-reachable vectors form a strictly increasing hierarchy? (iii) Decidability of hyper-reachability?

10 References

[1] R. Merkle and M. Hellman. Hiding information and signatures in trapdoor knapsack. *IEEE Transactions on Information Theory* IT-24 (1978) 525-530.
[2] A. Salomaa. *Computation and Automata.* Cambridge University Press (1985).
[3] A. Salomaa. Knapsacks and superdogs. Formal Language Theory Column in *EATCS Bulletin* (June 1989).
[4] A. Salomaa. A cubic-time deterministic algorithm for modular knapsack problems. To appear in *Theoretical Computer Science.*
[5] A. Shamir. A polynomial time algorithm for breaking the basic Merkle - Hellman cryptosystem. *Proceedings of the 3rd FOCS Symposium* (1982) 145-152.

(Received October 23, 1989)