SOLVING LOW DENSITY KNAPSACKS†

Ernest F. Brickell
Sandia National Laboratories
Albuquerque, New Mexico 87185

INTRODUCTION

Let $a_1,\ldots,a_n$ and s be a set of integers. The knapsack (or subset sum) problem is to find a 0-1 vector $(\varepsilon_1,\ldots,\varepsilon_n)$ such that $\Sigma\ \varepsilon_i a_i = s$ or to show that such a vector does not exist. The integers $a_1,\ldots,a_n$ are sometimes referred to as weights. The general knapsack problem is known to be NP complete [5,6]. Several cryptosystems based on the knapsack problem have been designed [9,12,16]. In April, 1982, Adi Shamir [14] announced a method for breaking the Merkle-Hellman cryptosystem. Since that time there has been a flurry of activity to extend his results to include all of the proposed knapsack based cryptosystems [1,2,3,7,13].

In a knapsack based cryptosystem, the cryptodesigner publishes a set of integers $a_1,\ldots,a_n$. A 0-1 vector $(\varepsilon_1,\ldots,\varepsilon_n)$ is encrypted by forming the sum $s = \sum_{i=1}^{n} \varepsilon_i a_i$. The cryptodesigner keeps secret certain information about the way the a_i's were chosen. This information allows him to decrypt any message, i.e., solve any knapsack problem where the integers $a_1,\ldots,a_n$ are the set of weights.

In all of the techniques mentioned above for breaking knapsack based cryptosystems, the cryptanalyst, using only the integers

† This work performed at Sandia National Laboratories supported by the U.S. Department of Energy under contract number DE-AC04-76DP00789.

$a_1,\ldots,a_n$, manages to find some of the secret information. In fact, the cryptanalyst can find enough information so that he als can solve any knapsack problem where the integers $a_1,\ldots,a_n$ are the set of weights.

Let $a_1,\ldots,a_n$ be a set of positive integers. Let $A = \max\{a_1,\ldots,a_n\}$. We define the *density* of a knapsack problem with weights $a_1,\ldots,a_n$ to be $n/\log_2 A$.

In this paper, we describe two methods for solving knapsacks of low density. Method 1 is a technique for solving knapsack problems that appears to work for almost all knapsacks of density less than $1/\log_2 n$. Method 2 is a slightly different technique that works on almost all knapsacks of density less than $d(n)$, where $d(n)$ is a function of n that is significantly larger than $1/\log_2 n$. However, our estimates of $d(n)$ come from emperical studies.

J. Lagarias and A. Odlyzko [8] have developed another technique for solving knapsacks of low density. The techniques are quite different. In our method, part of the algorithm works only with the weights $a_1,\ldots,a_n$ and runs in $O(n^4 (\log n)^3)$. If it is successful, then any knapsack problem in which the weights are $a_1,\ldots,a_n$ can be solved in $O(n^3)$. The Lagarias-Odlyzko method requires $O(n(\log A)^3)$ running time for each solution of a knapsack problem. They prove that their technique is expected to succeed if the density $\leq 1/n$ and show emperically that it is expected to succeed for much higher densities.

All of the above techniques for solving knapsack problems use the Lenstra-Lenstra-Lovasz (L^3) basis reduction algorithm for lattices [11]. A subset of points, L, of $\mathbf{R}^n$ is a lattice of rank n if $L = \{\sum_{i=1}^{n} z_i v_i : z_i \in \mathbf{Z}\}$ where $v_1,\ldots,v_n$ is some set of independent vectors in $\mathbf{R}^n$. The vectors $v_1,\ldots,v_n$ are said to be a basis of L. The L^3 algorithm finds a "short" or reduced basis for a given lattice. In [11], there are worst case bounds given on the lengths of vectors in a reduced basis. However, in emperi-

cal tests of the L^3 algorithm, it found a reduced basis that was much shorter than the worst case bounds indicate. Since the value of d(n) depends on the length of vectors in a reduced basis in a lattice of dimension n, we have not extrapolated from our tests on small n to predict the value of d(n) for arbitrary n.

SMALL SUM MODULAR MAPPINGS

We use the notation $b = a \bmod M$ (or just $a \bmod M$) to indicate that b (or a mod M) is the smallest integer in absolute value such that $b \equiv a \bmod M$.

A *modular mapping* by W mod M of a set of integers $a_1,\ldots,a_n$ is a mapping of $a_1,\ldots,a_n$ to $b_1,\ldots,b_n$ where $b_i \equiv a_i W \bmod M$.

A modular mapping by W mod M of $a_1,\ldots,a_n$ into $b_1,\ldots,b_n$ is said to have the *small sum property* iff $\sum_{i=1}^{n} |b_i| < M$. We will use SSMM to refer to a modular mapping with the small sum property.

Theorem 1: Suppose a modular mapping by W mod M of $a_1,\ldots,a_n$ into $b_1,\ldots,b_n$ has the small sum property. Let

$$B = \sum_{\substack{1\leqslant i\leqslant n \\ b_i>0}} b_i \quad .$$

Suppose $\sum_{i=1}^{n} \varepsilon_i a_i = s$ for some 0-1 vector $(\varepsilon_1,\ldots,\varepsilon_n)$. Let $s' = sW \bmod M$. Let

$$s'' = \begin{cases} s' & \text{if} \quad s' \leqslant B \\ s'-M & \text{if} \quad s' > B \end{cases} \quad .$$

Then $s'' = \sum_{i=1}^{n} \varepsilon_i b_i$.

Proof:

$$\sum_{i=1}^{n} \varepsilon_i b_i \equiv \sum_{i=1}^{n} \varepsilon_i a_i W \equiv sW \equiv s'' \bmod M \quad .$$

$$B-M < \sum_{i=1}^{n} \varepsilon_i b_i \leq B \quad .$$

The result follows. ∎

We say that a set $(W_1,M_1),\ldots,(W_k,M_k)$ of SSMMs for $a_1,\ldots,a_n$ are independent if the vectors $(a_1,\ldots,a_n)$, $(a_1W_1 \bmod M_1,\ldots, a_nW_1 \bmod M_1),\ldots,(a_1W_k \bmod M_k,\ldots,a_nW_k \bmod M_k)$ are independent.

Theorem 2: If a set of n-1 independent SSMM for $a_1,\ldots,a_n$ can be found, then the knapsack problem for weights $a_1,\ldots,a_n$ can be solved for any sum s by computing n modular multiplications and by multiplying an $n \times n$ matrix by a vector.

Proof: Let $(W_1,M_1),\ldots,(W_{n-1},M_{n-1})$ be a set of n-1 independent SSMM.

$$Y = \begin{pmatrix} a_1 & a_2 & & a_n \\ a_1W_1 \bmod M_1 & a_2W_1 \bmod M_1 & \cdots & a_nW_1 \bmod M_1 \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ a_1W_{n-1} \bmod M_{n-1} & a_2W_{n-1} \bmod M_{n-1} & & a_nW_{n-1} \bmod M_{n-1} \end{pmatrix}$$

Given an integer s, we want to find a 0-1 vector $\boldsymbol{\alpha} = (\alpha_1,\ldots,\alpha_n)$ such that

$$\sum_{i=1}^{n} \alpha_i a_i = s \quad , \qquad (1)$$

or show that such a 0-1 vector does not exist.

Assume that a 0-1 vector satisfying (1) exists. By Theorem 1, we can find $s_1,\ldots,s_{n-1}$ such that

$$\sum_{i=1}^{n} \alpha_i (a_i W_j \bmod M_j) = s_j \qquad 1 \leq j \leq n-1 \quad .$$

Let $\mathbf{s} = (s, s_1, \ldots, s_{n-1})$.

$$Y\boldsymbol{\alpha} = \mathbf{s} \quad .$$
$$Y^{-1}\mathbf{s} = \boldsymbol{\alpha} \quad .$$

Hence to solve the knapsack problem for sum s, we form the vector $\mathbf{s}$ as above. If $Y^{-1}\mathbf{s}$ is not a 0-1 vector, then the problem has no solution. If $Y^{-1}\mathbf{s} = \boldsymbol{\alpha}$ is a 0-1 vector, then compute $\boldsymbol{\alpha} \cdot a$. If $\boldsymbol{\alpha} \cdot a = s$, then $\boldsymbol{\alpha}$ is a solution, and if $\boldsymbol{\alpha} \cdot a \neq s$, then the problem has no solution. ∎

Since we are only interested in whether or not $Y^{-1}\mathbf{s}$ is a 0-1 vector, we can do the multiplication of $Y^{-1}\mathbf{s}$ modulo p, where p is the smallest prime such that Y is irreducible mod p.

Method I

Given weights $a_1, \ldots, a_n$, we perform an L^3 basis reduction on the lattice L in $\mathbf{R}^n$ generated by the vectors

$$\begin{pmatrix} 1 \\ na_2 \\ na_3 \\ \cdot \\ \cdot \\ \cdot \\ na_n \end{pmatrix}, \quad \begin{pmatrix} 0 \\ na_1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{pmatrix}, \quad \ldots, \quad \begin{pmatrix} 0 \\ 0 \\ \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ na_1 \end{pmatrix} . \qquad (2)$$

Proposition 3: Let $\mathbf{x} = (x_1,\ldots,x_n)$ be a vector in L. If

$$\sum_{i=2}^{n} |x_i| < na_1 \quad ,$$

then the modular mapping of $a_1,\ldots,a_n$ by $x_1 \bmod a_1$ has the small sum property.

Proof: Since $\mathbf{x}$ is an integral linear combination of the vectors in (2), there exist integers $(y_1,\ldots,y_n)$ such that

$$x_1 = y_1$$

$$x_i = y_i na_1 + y_1 na_i \qquad \text{for} \qquad 2 \leq i \leq n \quad .$$

Since $n|x_i$, let $x_i' = x_i/n$ for $2 \leq i \leq n$.

$$x_i' \equiv a_i y_1 \bmod a_1 \qquad \text{for} \qquad 2 \leq i \leq n$$

$$0 \equiv a_1 y_1 \bmod a_1 \quad .$$

Since $\sum_{i=1}^{n} |x_i'| < a_1$, the modular mapping by $x_1 \bmod a_1$ has the small sum property. ∎

For a vector $\mathbf{x} = (x_1,\ldots,x_n)$ in the lattice L, define $x_1' = 0$, $x_i' = x_i/n$ for $2 \leq i \leq n$, and $\mathbf{x}' = (x_1',\ldots,x_n')$. $\mathbf{x}$ is said to be short enough if $\sum_{i=2}^{n} |x_i'| < a_1$.

Proposition 4: If all vectors in the reduced basis for L are short enough, then we can find n-1 independent SSMM for $a_1,\ldots,a_n$.

Proof: Let $\mathbf{v}_1,\ldots,\mathbf{v}_n$ be the vectors in the reduced basis. Since $\mathbf{v}_1,\ldots,\mathbf{v}_n$ are independent, there exists a set of n-1 of the vectors $\mathbf{v}_1',\ldots,\mathbf{v}_n'$ that are independent. Renumber so that $\mathbf{v}_1',\ldots,\mathbf{v}_{n-1}'$ are independent. Let z_i be the first coordinate of $\mathbf{v}_i$. Then (z_1,a_1), $\ldots,(z_{n-1},a_1)$ is a set of n-1 independent SSMM for $a_1,\ldots,a_n$. ∎

To evaluate the probability of success of Method 1, we want to estimate the number of integral solutions $(y_1,\ldots,y_n)$ to

$$\begin{aligned} |a_i y_1 + y_i a_1| &\leq a_1/n \qquad 2 \leq i \leq n \\ 0 &< y_1 < a_1 \quad . \end{aligned} \tag{3}$$

Let K be the region in $\mathbf{R}^n$ defined by (3). The vertices of K are the 2^n points

$$(0, \pm 1/n, \ldots, \pm 1/n)$$

$$(a_1, a_2 \pm 1/n, \ldots, a_n \pm 1/n) \quad .$$

K is a parallelepiped of volume $a_1(2/n)^{n-1}$. In [14], it is shown that in a region of volume V, the expected number of integral points is V. Thus we will estimate the number of integral solutions to (3) by $a_1(2/n)^{n-1}$.

Hence if $a_1 \approx n^n$, the expected number of solutions is $n2^{n-1}$. So if the density if $\leq 1/\log_2 n$, then we expect Method 1 to work. Tests of this method up to $n = 30$ have been successful.

Method 2

If the density of a knapsack is $> 1/\log_2 n$, then we cannot expect Method 1 to work. What we will do is to artificially lower the density. We cannot expect this to work in all cases, but it will be successful under certain conditions. We will explain this later.

Given weights $a_1,\ldots,a_n$ of density $> 1/\log_2 n$, pick M and $W < M$ such that $n^n < M < 2n^n$ and $(W,M) = 1$. Let

$$b_i \equiv a_i W \bmod M, \qquad 1 \leq i \leq n \quad ,$$

such that b_i is the least nonnegative residue.

Now the density of $b_1,\ldots,b_n$ is $< 1/\log_2 n$, so we can use Method 1 on $b_1,\ldots,b_n$. Since the weights $b_1,\ldots,b_n$ are not randomly chosen, we cannot show that Method 1 is expected to work. However, if Method 1 is successful for the weights $b_1,\ldots,b_n$ then we can solve any knapsack problem with weights $a_1,\ldots,a_n$ and any sum s. If there exists a 0-1 vector $\boldsymbol{\alpha}$ such that

$$\sum_{i=1}^{n} \alpha_i a_i = s \quad , \tag{4}$$

then we have that

$$\sum_{i=1}^{n} \alpha_i b_i \equiv sW \bmod M \quad .$$

But

$$\sum_{i=1}^{n} b_i < nM \quad .$$

Let $s' \equiv sW \bmod M$, such that s' is the least nonnegative residue.

$$\sum_{i=1}^{n} \alpha_i b_i \in \{s', s'+M, \ldots, s'+(n-1)M\} \quad . \tag{5}$$

We use Method 1 to find all solutions $\boldsymbol{\alpha}$ to (5), and test each one to see if it is a solution to (4). If we find no solutions, then we can be certain that there are no 0-1 vectors $\boldsymbol{\alpha}$ satisfying (4).

It is difficult to analyze Method 2 to determine exactly when it will work. However there is something we can say about when it will not work.

For an integral vector $\mathbf{x} = (x_1,\ldots,x_n)$, let

$$\|\mathbf{x}\| = \sum_{i=1}^{n} |x_i| \quad .$$

Theorem 5: For $a_1,\ldots,a_n$, W, M $\in \mathbf{Z}^+$, let $\mathbf{a} = (a_1,\ldots,a_n)$. For $1 \leq i \leq n$, let

$$c_i = a_i W \bmod M \quad .$$

Let $\mathbf{c} = (c_1,\ldots,c_n)$. Let $\mathbf{x}$ be an integral vector with $\|\mathbf{x}\| \leq n$. If $|c_i| < M/n$ for $1 \leq i \leq n$, and $\mathbf{x} \cdot \mathbf{a} = 0$, then $\mathbf{x} \cdot \mathbf{c} = 0$.

Proof: $\mathbf{x} \cdot \mathbf{c} \equiv 0 \bmod M$.

$$|\mathbf{x}\cdot\mathbf{c}| = \Big|\sum_{i=1}^{n} x_i c_i\Big| \leq \sum_{i=1}^{n} |x_i||c_i| < M/n \sum_{i=1}^{n} |x_i| \leq M \quad .$$

So $\mathbf{x} \cdot \mathbf{c} = 0$. ∎

Proposition 6: Let $a_1,\ldots,a_n \in \mathbf{Z}^+$. Let $\mathbf{x} \neq 0$ be an integral vector such that $\mathbf{x} \cdot \mathbf{a} = 0$. Suppose that if (W,M) is any SSMM of $a_1,\ldots,a_n$, then $\mathbf{x} \cdot \mathbf{c} = 0$, where

$$c_i = a_i W \bmod M \quad .$$

Then there do not exist n-1 independent SSMMs for $a_1,\ldots,a_n$.

Proof: There can be at most n-1 independent vectors perpendicular to $\mathbf{x}$. ∎

Suppose there are a lot of small linear dependencies satisfied by $a_1,\ldots,a_n$ (i.e., there are a lot of vectors $\mathbf{x}$ such that $\|\mathbf{x}\| \leq n$ and $\mathbf{x} \cdot \mathbf{a} = 0$). In Method 2, we form $b_1,\ldots,b_n$ where $b_i \equiv a_i W \bmod M$. The number of small linear dependencies satisfied by $b_1,\ldots,b_n$ will be about $n^{-1/2}$ times the number satisfied by $a_1,\ldots,a_n$. If $b_1,\ldots,b_n$ satisfy a lot of small linear dependencies, then Method 1 is not likely to work, and thus Method 2 will fail for the weights $a_1,\ldots,a_n$.

We now need to determine when the weights $a_1,\ldots,a_n$ are expected to satisfy a lot of small linear dependencies.

Let $\phi(n) = \{(x_1,\ldots,x_n): x_i \in \mathbf{Z} \text{ and } 0 < \sum_{i=1}^{n} |x_i| \leq n\}$.
For $1 \leq k \leq n$, let

$$\theta(n,k) = \{(x_1,\ldots,x_n) \in \phi(n): \text{exactly } k \text{ of the } x_i\text{'s are nonzero}\}$$

To calculate $|\theta(n,k)|$, we can choose which k of the x_i's are nonzero in $\binom{n}{k}$ ways. Suppose we have chosen $x_{i_1},\ldots,x_{i_k}$ to be nonzero. To decide what each x_{i_j} should be, select k integers $1 \leq y_1 < y_2 < \ldots < y_k \leq n$ in $\binom{n}{k}$ ways. Then let $x_{i_1} = y_1$, $x_{i_j} = y_j - y_{j-1}$ for $2 \leq j \leq k$. Finally we can assign + or - to the x_{i_j} in 2^k ways.

$$|\theta(n,k)| = 2^k \binom{n}{k} \binom{n}{k}$$

$$|\theta(n)| = \sum_{k=1}^{n} 2^k \binom{n}{k} \binom{n}{k} .$$

The maximum of $\{2^k \binom{n}{k} \binom{n}{k} : 1 \leq k \leq n\}$ occurs at $k \approx (2 - \sqrt{2})n$, and for this value of k, $|\theta(n,k)| \approx 2^{2.54n}$. Since $|\phi(n)| \leq n|\phi(n,k)|$ and we are only interested in the $\log_2|\phi(n)|$, we will approximate $|\theta(n)|$ by $2^{2.54n}$.

Theorem 7: Let $\mathbf{a} = (a_1,\ldots,a_n)$ where each a_i is randomly chosen from the interval [1,N], where $N \geq 2^{2.54n}$. Then the expected number of small linear dependencies satisfied by $\mathbf{a}$ is less than 1, i.e., $E(C(\mathbf{a})) < 1$.

Proof: Let $\mathcal{A}(n) = \{(a_1,\ldots,a_n): a_i \in \mathbf{Z}, 1 \leq N \text{ for } 1 \leq i \leq n\}$. $|\mathcal{A}(n)| = N^n$. We want to get an upper bound on P(n) = the number of pairs of vectors $(\mathbf{x},\mathbf{a})$ where $\mathbf{x} \in \phi(n)$, $\mathbf{a} \in \mathcal{A}(n)$, and $\mathbf{x}\cdot\mathbf{a} = 0$. For each $\mathbf{x} = (\alpha_1,\ldots,\alpha_n) \in \phi(n)$, there exists i, $1 \leq i \leq n$, such that $\alpha_i \neq 0$. If $a_1,\ldots,a_{i-1},a_{i+1},\ldots,a_n$ are integers $\in [1,N]$, then there is at most one choice for a_i such that $(\alpha_1,\ldots,\alpha_n)\cdot(a_1,\ldots,a_n) = 0$. Thus there are at most $(N)^{n-1}$ vectors in $\mathcal{A}(n)$ that are perpen-

dicular to $\mathbf{x}$. Since $|\phi(n)| < 2^{2.54n}$, $P(n) < N^{n-1}2^{2.54n}$. Since $P(n) < |\mathcal{A}(n)|$, for a random vector $\mathbf{a} \in \mathcal{A}(n)$, the expected number of $\mathbf{x} \in \phi(n)$ perpendicular to $\mathbf{a}$ is less than 1. ∎

Thus if the density of the weights $a_1,\ldots,a_n$ is greater than .39 = 1/2.54, then there will probably be many small linear dependencies satisfied by $a_1,\ldots,a_n$.

MODIFICATIONS AND TESTS

Consider the following problem, which we will call the half knapsack problem.

Given a set of integers $a_1,\ldots,a_n$, and s, find a 0-1 vector $\mathbf{a} = (\alpha_1,\ldots,\alpha_n)$ with less than half ones, i.e., $|\{i:\alpha_i = 1\}| \leq \lceil \frac{n}{2} \rceil$, such that $\sum_{i=1}^{n} \alpha_i a_i = s$, or show that such a 0-1 vector does not exist.

Note that we can solve the knapsack problem for the weights $a_1,\ldots,a_n$ and sum s by solving the half knapsack problem for the weights a_i and sum s and also for weights a_i and sum $\sum_{i=1}^{n} a_i - s$.

A modular mapping by W mod M of $a_1,\ldots,a_n$ into $b_1,\ldots,b_n$ is said to have the *small half sum property* iff $\sum_{i\in B} |b_i| < M$ for any $B \subseteq \{1,\ldots,n\}$ with $|B| = \lceil \frac{n}{2} \rceil$.

We can prove an analog to Theorem 2 for solving the half knapsack problem if we have n-1 independent SHSMMs. Then we can modify Methods 1 and 2 to search for n-1 independent SHSMMs. We can then show that for knapsacks of density less than .44, Method 2 would probably succeed if the L^3 algorithm actually found the shortest basis in the lattice.

In our tests of this modified Method 2, we were successful on most knapsacks of density less than .44 for $n \leq 26$, but for n = 28, we had to drop the density below .40 to achieve success. The reason the density drops as the dimension increases is due to the performance of the L^3 algorithm. As the dimension increases, the ratio of the length of vectors in the reduced basis to the

length of vectors in the shortest basis in the lattice also increases. This property makes it difficult to predict for large values of n what the cutoff density for success of this algorithm will be. However, we can say that there is some function d(n) such that this algorithm will be successful on almost all n-weight knapsacks of density < d(n).

ACKNOWLEDGEMENT

I would like to thank Jeff Lagarias and Andrew Odlyzko for their helpful comments on this paper.

REFERENCES

1. L. M. Adleman, "On Breaking the Generalized Knapsack Public Key Cryptosystems," Proceedings of the 15th Annual Symposium on Theory of Computing (1983), 402-412.

2. E. F. Brickell, "Are Most Low Density Knapsacks Solvable in Polynomial Time?," to appear in Congressus Numerantium (1983).

3. E. F. Brickell and G. J. Simmons, "A Status Report on Knapsack Based Public Key Cryptosystems," Congressus Numerantium, Vol. 37 (1983), 3-72.

4. W. Diffie and M. E. Hellman, "New Directions in Cryptography," IEEE Trans. Inform. Theory IT-22, 6 (Nov. 1976), 644-654.

5. M. R. Garey and D. S. Johnson, Computers and Intractability, A Guide to the Theory of NP-Completeness, W. H. Freeman and Company, San Francisco (1979).

6. R. M. Karp, "Reducibility Among Combinatorial Problems," in Complexity of Computer Computations, R. E. Miller and J. W. Thatcher (Eds.), Plenum Press, New York (1972), 85-104.

7. J. C. Lagarias, "Knapsack Public Key Cryptosystems and Diophantine Approximation," to appear Advances in Cryptography (1983).

8. J. C. Lagarias and A. M. Odlyzko, "Solving 'Low-Density' Subset Sum Problems," to appear.

9. A. Lempel, "Cryptology in Transition: A Survey," Comput. Surv. 11, 4 (Dec. 1979), 285-304.

10. H. W. Lenstra, Jr., "Integer Programming with a Fixed Number of Variables," Univ. of Amsterdam Tech. Report 81-03 (April 1981); to appear, Math. of Operations Research.

11. A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovasz, "Factoring Polynomials with Rational Coefficients," Mathematische Annalen, Vol. 261, No. 4 (1982), 515-534.

12. R. C. Merkle and M. Hellman, "Hiding Information and Signatures in Trapdoor Knapsacks," IEEE Trans. Inform. Theory IT-24, 5 (Sept. 1978), 525-530.

13. A. M. Odlyzko, "Cryptanalytic Attacks on the Multiplicative Knapsack Cryptosystem and on Shamir's Fast Signature System," to appear.

14. L. A. Santaló, Integral Geometry and Geometric Probability, Addison-Wesley Publishing Company (1976).

15. A. Shamir, "A Polynomial Time Algorithm for Breaking the Basic Merkle-Hellman Cryptosystem," Proc. 23rd Annual Symposium on Foundations of Computer Science (1982), 145-152.

16. A. Shamir, "The Strongest Knapsack-Based Cyrptosystem?," (extended abstract) paper presented at Crypto'82, Santa Barbara, CA (August 1982).