TR1087

COINS AND CONES

DMITRY N. KOZLOV AND VAN H. VU

ABSTRACT. We discuss the problem of maximizing the number of coins, for which, using just n weighings, one can tell whether all of them are of the same weight or not. The first purpose of the paper is to show the connection between this problem and a problem in lattice geometry. Using this approach, we are able to establish an upper bound on the number of coins and also to disprove the conjecture that the maximal number of coins is 2^n by giving some quick algorithms for the original problem. We also conjecture that the upper bound is asymptotically tight.

1. INTRODUCTION

Let us begin with a description of the general counterfeit coin problem. We start with a set S of m coins. We know that at most two different weights can occur among them, but we do not know what these weights are. Those coins, which have a weight different from the majority are called the **counterfeit** coins. We are allowed to do an operation which we call a **weighing**. Each weighing is a comparison of the weights of two chosen groups of coins. A weighing can have three different outcomes: "the first group is lighter", "the first group is heavier" and "the groups have the same weight". Since we do not know the weights in advance, it may happen that different coins differ very little (if at all) in weight, thus comparing any two unequal groups will always show that the group with more coins is heavier, which does not give us any information about the weights of the coins. For that reason we will always compare groups of equal sizes. We are now ready to state the promised problem.

The general counterfeit problem. *Given a set of m coins of at most two different weights. Determine the set C of the counterfeit coins. The solution is called* **optimal** *if it uses as few weighings as possible.*

The case when there is exactly one counterfeit coin has been known as a mathematical puzzle for a long time. As soon as $|C| > 1$ the problem turns out to be hard and the minimum number of weighings is still unknown. However, a lower bound can be easily achieved by the following information-theoretic argument. Let c denote the cardinality of C. Then C can be each of $\binom{m}{c}$ subsets of size c of S. Observe that each weighing has 3 outcomes, it is clear that we need at least $\log_3 \binom{m}{c}$ weighings. This lower bound turns out to be not far from the optimum. In the case c is known, there is an algorithm which detects all the counterfeit in $\log_3 \binom{m}{c} + 15c$ steps [LP]. More natural is the case when c is not known, for this authors in [HH] and [CHH] provided an algorithm which needs $a \log_3 \binom{m}{c}$ steps, where a is a constant (the best constant known is $2\log_2 3$ [CHH]).

Date: February 29, 1996.

This paper was written while the first author was visiting the Departments of Computer Science and Mathematics at Yale University, supported by a KTH-Yale joint research program.

The following problem is closely related to the general counterfeit problem.

The all-equal problem. *Given a set S of m coins, decide if all the coins have the same weight or not.*

It can be shown that in general this problem cannot be solved faster than using $m-1$ weighings, see [K]. However if one imposes some very natural condition, this bound can be significantly improved. The purpose of this paper is to consider the all-equal problem under the assumption that the weights of the coins are generic. Technically speaking we have the following condition.

Condition (*). If $w_1, \dots, w_t$ are the different weights occuring among the coins, then there are no integers $\lambda_1, \dots, \lambda_t$ such that not all λ_i's are equal to 0 and the following is true:

(1) $\sum_{i=1}^{t} \lambda_i w_i = 0$,
(2) $\sum_{i=1}^{t} \lambda_i = 0$.

If the condition (*) is satisfied for some set of coins, we will say that these coins have **generic weights.** We will refer to the all-equal problem with this additional condition imposed on the occuring weights as the **generic all-equal problem**.

Although the definition may seem somewhat technical it has a very natural meaning. Imagine that we have compared two groups of coins, A and B, and that the outcome says that they are of equal weight. Then the condition (*) simply means that for any weight w, the number of coins of weight w in A is the same as that in B. Observe also that if the coins have at most two different weights then (*) is satisfied.

It is clear that the algorithm which solves the generic all-equal problem should stop as soon as the weighing is not balanced. Thus its objective is to perform a certain number of weighings, such that: *if all of them are balanced then all the coins must have the same weight.*

The reader has probably already recognized that the "$\log_3 \binom{m}{c}$" argument above does not work any more, and we do not get any lower bound for the minimum number of weighings. On the other hand, there is a very natural "doubling" algorithm: in the first step compare two coins, in the $(k+1)^{\text{th}}$ step, weigh the set of coins used in the first k steps with a set of new coins of the same cardinality. If every weighing was balanced, it is trivial that all the coins have the same weight, and the algorithm solves the problem in $\lceil \log_2 m \rceil$ steps (i.e. in n steps we can solve the generic all-equal problem for up to 2^n coins).

For a while it has been believed that $\log_2 m$ is the best one can do. Authors in [HH] have also attempted to prove this in Theorem 1, their proof however was incomplete. In section 4 we will show that this is actually false.

In section 2 we shall discuss the background of this problem, which is, surprisingly, related to lattice geometry, and has nothing to do with information theory. This new approach will allow us to find an algorithm using essentially less steps than $\log_2 m$, and hence to disprove the conjecture. Furthermore, we prove a lower bound $O(\log m / \log\log m)$, i.e. we prove that in n steps we can solve the problem for at most $\exp(O(n \log n))$ coins (it does not matter here which base the $\log n$ has).

Finally, in section 4, we formulate a conjecture (Conjecture 4.1) claiming the existence of an algorithm, which in n steps solves the problem for $\exp(cn \log n)$ coins, for some positive constant c. This conjecture is equivalent to an interesting question concerning arithmetics of the determinants of integer matrices.

So, the general question we will try to answer is:

What is the maximum size of a set of coins, the weight-uniformness of which can be decided by n weighings ?

To start let us now state the problem in a more mathematical way. Let A_i and B_i, $i = 1, 2, ..., n$, be the sets of coins we weigh in the i^{th} step, $|A_i| = |B_i|$. If every weighing was balanced and it is still possible that coins can have different weights, then we take a set C, consisting of all the coins of some certain weight (not an empty set). Because of the condition (*) we get $|C \cap A_i| = |C \cap B_i|$ for every i. So we end up with the following question:

Let S be a set of m elements. Consider n pairs A_i, B_i, $i = 1, 2, ..., n$, of subsets of S, such that $|A_i| = |B_i|$ and there is no proper subset C of S, the intersections of which with A_i and B_i have the same cardinality for all i. What is the largest value of m, for which one can find such a family of pairs, assuming that the value of n is fixed ?

2. Translation into Linear Algebra Language

In this section we will introduce a new approach to the problem. The idea is to apply linear algebra.

Assume that as above we have a set S, $|S| = m$, and pairs of subsets of S, (A_i, B_i), $i = 1, \ldots, n$, such that $|A_i| = |B_i|$. We will construct a set of m vectors in $\mathbb{R}^n$ associated to this data. Let x be an element of S then we define $v_x \in \mathbb{R}^n$ by the rule that the i^{th} coordinate of v_x is equal to 1 if x belongs to A_i, -1 if x belongs to B_i and 0 if x lies outside both A_i and B_i. Since we choose A_i and B_i non-intersecting, v_x is well-defined.

The condition that $|A_i| = |B_i|$ will then simply translate into

$$\sum_{x \in S} v_x = 0. \tag{2.1}$$

Let W be the set of all the vectors in $\mathbb{R}^n$ with coordinates from the set $\{0, 1, -1\}$. Obviously $|W| = 3^n$ and for any $x \in S$ we have $v_x \in W$. Having any vector w from W, let λ_w count the number of occurencies of w among $\{v_x | x \in S\}$. Then the equation 2.1 translates into

$$\sum_{w \in W} \lambda_w w = 0, \tag{2.2}$$

where obviously λ_w is a non-negative integer for all $w \in W$.

Furthermore, what does it mean that there exists a subset C of S, such that $|C \cap A_i| = |C \cap B_i|$? It means exactly that

$$\sum_{x \in C} v_x = 0$$

or in terms of vectors from W we get

$$\lambda_w = \alpha_w + \beta_w \quad \forall\, w \in W, \tag{2.3}$$

such that $\sum_{w \in W} \alpha_w w = \sum_{w \in W} \beta_w w = 0$ and α_w, β_w are non-negative integers for all $w \in W$. The fact that C is a proper subset of S means that not all of α's are equal to 0 and not all of β's are equal to 0.

Let us now impose an order on the 3^n vectors from W and consider all the vectors $\lambda \in \mathbb{R}^{3^n}$ such that equation 2.2 is satisfied. These vectors obviously form a subspace which we will call T. Form a matrix M of size $n \times 3^n$ by taking the vectors from W as columns, then $\text{rk}M = n$ and T is the kernel of M, hence $\dim T = 3^n - n$.

Let furthermore $\mathbb{R}_+^{3^n}$ denote the positive cone of $\mathbb{R}^{3^n}$, i.e. the cone defined by the equations: $x_i \geq 0$, $i = 1, 2, \ldots, 3^n$. We denote $K = T \cap \mathbb{R}_+^{3^n}$ and let Z be all the vectors from K with integer coordinates. K is obviously a polyhedral cone and the vectors λ, α and β in equation 2.3 are all from Z. To restate the existence of a subset C of S with the properties mentioned above we need the notion of an integral Hilbert basis. This notion was first introduced by [GP]. The following definition is a slight reformulation of the one in Chapter 16 of [Sc].

Definition 2.1. *Given a polyhedral cone K, let Z be the set of all the integer vectors in K. A finite set of vectors $\{a_1, a_2, \ldots, a_t\}$ from Z is called an* **integral Hilbert basis** *if each integral vector b in K is a nonnegative integral combination of $a_1, a_2, \ldots, a_t$.*

In general an integral Hilbert basis does not have to exist (i.e. the set of generators described above does not have to be finite), hence the set of the sums of the coordinates of its vectors does not have to be bounded. However this is true in our case, because the polyhedral cone K above is defined by rational equations, hence it is a rational cone. The following appears as Theorem 16.4 in [Sc].

Theorem 2.2. *Each rational polyhedral cone K has an integral Hilbert basis. If K is pointed there is a unique minimal integral Hilbert basis (minimal relative to taking subsets).*

It is easy to see that the unique minimal integral Hilbert basis of K consists of exactly those vectors from Z, which cannot be written as a sum of two other vectors from Z. Let us denote this set by H. Every algorithm (i.e. a family of pairs of sets (A_i, B_i)), which in n steps decides whether m given coins are all of the same weight or not, gives rise to a vector v, in H, such that the sum of its coordinates is equal to m. In fact this correspondence is a bijection, because starting from such a vector we can easily read off the vectors $\{v_x | x \in S\}$ and hence see which pairs of sets (A_i, B_i) we are to take in the set S.

To clarify what we are doing, let us shortly summarize the discussion above. We have W - the set consisting of 3^n vectors in $\mathbb{R}^n$ with coordinates $\pm 1, 0$. We consider the polyhedral cone $K = T \cap \mathbb{R}_+^{3^n}$, where T is the set of all linear dependencies of vectors from W (T is a subspace of $\mathbb{R}^{3^n}$). Because of Theorem 2.2 K has a unique minimal integral Hilbert basis, which we denote by H. The question now is: *What is the maximum of the sum of the coordinates of vectors in H?*

From the fact that H is finite we immediately derive that there is a function $f(n)$, such that if $m > f(n)$, then there is no algorithm which in n steps decides whether all the given m coins are of the same weight or not. The more rigorous bound $f(n) = \exp(O(n \log n))$ will be proven in the next section (Corollary 3.5).

3. The upper bound

For the sake of brevity, we call the sum of the coordinates of a vector x the **weight** of x, and denote it by $w(x)$. A direct approach to estimate the maximum weight of vectors in a minimal Hilbert basis H might be to determine the basis explicitly, and then to optimize the function w on that. Although we know all the boundary hyperplanes of the cone, this approach seems to be very difficult because of the high dimension of the space. Our idea here is to estimate the weights of the vectors in H via the weights of the so-called **generator vectors** of K, which we can count directly from the matrix M.

We call a half-line starting from the origin a **generator half-line** if it is the intersection of K with some hyperplane. A vector $x = OX$, where $X \neq O$ being a point on a generator half-line is called a **generator vector** (or shortly just a **generator**). We quote here some standard results on generator half-lines and vectors.

Lemma 3.1. *If K is a cone determined by a finite number of half-spaces, then there are finitely many generator half-lines, and K is the convex hull of those.*

Clearly it follows that x is a generator vector iff x cannot be written as $x = u+v$, $u, v \in K$, where u and v are independent from x.

Lemma 3.2. *(Caratheodory) If K is the convex hull of p half-lines $l_1, l_2, ..., l_p$ in a k-dimensional space, $p > k$, then for every $x \in K$, there is a set $\{i_1, ..., i_k\} \subset \{1, 2, ..., p\}$, such that x is contained in the convex hull of the k half-lines l_{i_j}.*

In other words we can "triangulate" our cone, i.e. divide it into simplicial cones.

Now we are going to describe the generators of the cone K defined in Chapter 2. For $x \in K, x \neq 0$, let $x_{i_1}, ..., x_{i_{l+1}}$ be the non-zero coordinates of x. Denote by $\bar{x}$ the $(l+1)$-dimensional vector $(x_{i_1}, ..., x_{i_{l+1}})$, and M_x the submatrix of M formed by the columns labeled $i_1, i_2, \ldots, i_{l+1}$. The following Lemma characterizes the generators of K.

Lemma 3.3. *x is a generator of K if and only if*

(a) *$\bar{x}$ is a positive vector and $M_x\bar{x} = 0$*
(b) *$rk(M_x) = l$, where $l+1$ is the length of $\bar{x}$.*

Proof: Let x be a generator. Condition (a) is immediate since K is a non-negative cone and $Mx = 0$. For convenience, suppose that $\bar{x}$ consists of the first $l+1$ coordinates of x, $\bar{x} = (x_1, x_2, \ldots, x_{l+1})$. Assume $\mathrm{rk}(M_x) < l$. It follows that $\dim \mathrm{Ker}(M_x) \geq 2$, hence one can find a vector $\bar{x}' \in \mathbb{R}^{l+1}$ independent from $\bar{x}$ and $M_x\bar{x}' = M_x\bar{x} = 0$. Extend $\bar{x}'$ to a 3^n-dimensional vector $x' = (\bar{x}', 0, 0, 0, \ldots, 0)$, obviously $Mx' = Mx = 0$. Since x_i, $i = 1, \ldots, l+1$ are positive, there are positive numbers α and β such that $u = \alpha x - x'$ and $v = \beta x - \alpha x + x'$ are non-negative vectors. Trivially $u, v \in K$, since they satisfy $Mu = Mv = 0$. Note that the independence of $\bar{x}$ and $\bar{x}'$ implies that of u and v. This is a contradiction because $\beta x = u + v$ and βx itself is also a generator. This proves condition (b).

To prove the converse implication, one just recognizes that if $x = u+v$, $x, u, v \in K$ and $x_k = 0$, then $u_k = v_k = 0$. So if x satisfies (a) and $x = u+v$, then $\bar{x} = \bar{u}+\bar{v}$ (if $\bar{u}$ or $\bar{v}$ has length smaller than that of $\bar{x}$, we extend it by some zeros). Moreover, $Mu = Mv = 0$, so $M_x u = M_x v = M_x x = 0$. But by (b) M_x is an $n \times (l+1)$ matrix of rank l, this means the equation $M_x y = 0$ has only one solution up to scalar multiplication. Thus u and v are scalar multiples of x. This completes the proof. □

We are now particularly interested in the integral generators. We call the integral generator vector with the minimal sum of coordinates on each generator half-line a **minimal generator**. Since K is rational, each generator half-line contains such a vector (it is easily read from the previous Lemma, too). It is also clear that a minimal generator is contained in the minimal Hilbert basis. We are going to estimate the weights of the minimal generators.

First we need the following observations. Let L be a full ranked $l \times (l+1)$ submatrix of M, then the equation $Ly = 0$ has a non-trivial integral solution

$$y = (\det L_1, -\det L_2, \dots, (-1)^l \det L_{l+1}),$$

where L_i is the l by l submatrix obtained from L by deleting the i^{th} column. Let

$$g(L) = \frac{\sum_{i=1}^{l+1} |\det L_i|}{\text{g.c.d.}(|\det L_i|)_{i=1}^{l+1}}. \tag{3.1}$$

Consider a minimal generator x with the corresponding submatrix M_x of $l+1$ columns. By Lemma 3.3, M_x has rank l. So M_x contains an $l \times (l+1)$ full ranked submatrix L, and $\bar{x}$ is a non-trivial positive solution of the equation $Ly = 0$. Moreover, x is minimal, hence $\bar{x} = (|\det L_1|/d, |\det L_2|/d, ..., |\det L_{l+1}|/d)$, where $d = \text{g.c.d.}(|\det L_i|)_{i=1}^{l+1}$ (remember $\bar{x}$ is positive). Thus $w(x) = w(\bar{x}) = g(L)$.

Denote

$$\gamma(n) = \max g(L),$$

where L runs over the set of all $l \times (l+1)$ full ranked submatrices of M, $l \le n$ such that $Ly = 0$ has a positive solution. Note that M consists of all possible $\{0, +1, -1\}$ column vectors, readers can easily see that $\gamma(n)$ takes the same value if we allow L to run over a larger set, namely on all $l \times (l+1)$ $(l \le n)$ full rank submatrices of M.

It is clear from the previous argument that the maximum weight of a minimal generator is $\gamma(n)$. Now we are ready to state the next theorem.

Theorem 3.4. *Let $f(n)$ be the maximum weight of a vector in the minimal Hilbert basis H, then*

$$\gamma(n) \le f(n) \le \frac{3^n - 1}{2} \gamma(n)$$

Proof: The first inequality is immediate since every minimal generator is an element of the Hilbert basis. Observe that if z is a column vector of M, then so is $-z$, hence for every element y of the Hilbert basis, y has at most $(3^n - 1)/2$ positive coordinates. Let K' be the intersection of K and the hyperplanes $y_i = 0$, where y_i are the zero coordinates of y. It is clear that y is an element of the Hilbert basis of K' and K' has dimension at most $(3^n - 1)/2$. On the other hand, if y is written as a positive combination of some vectors of K, then those vectors should also be contained in K'.

Due to Lemma 3.1, y can be expressed as a positive combination of minimal generators. Moreover, by Lemma 3.2 and the above note, we need at most $(3^n - 1)/2$ terms in the combination. So y can be written in the form

$$y = \sum_{i=1}^{(3^n-1)/2} \alpha_i x_i$$

where x_i are minimal generators, and α_i are non-negative coefficients. Since H is the minimal Hilbert basis, we know that $y - x_i \notin K$, thus $\alpha_i < 1$ for every i. This yields

$$w(y) \le \sum_{i=1}^{(3^n-1)/2} \alpha_i w(x_i) < \sum_{i=1}^{(3^n-1)/2} w(x_i) < (3^n - 1)\gamma(n)/2$$

proving the theorem. □

Corollary 3.5. $f(n) \le \frac{(3^n-1)(n+1)^{(n+1)/2}}{2}$

Proof: Take a l by $l+1$ matrix L, with L_i being its $l \times l$ submatrices. The sum $\sum_{i=1}^{l} |\det L_i|$ can be seen as the determinant of a $(l+1) \times (l+1)$ matrix L', which is an extension of L by an appropriate $(-1, 1)$ row.

Note that $|\det L'|$ is the volume of the parallelepiped spanned by its column vectors, which is not larger than the product of their norms. Since L' is a $\{0, 1, -1\}$ matrix, the norm of each column vector is at most $(l+1)^{1/2}$, which is not larger than $(n+1)^{1/2}$. Consequenly, $\gamma(n) \le (n+1)^{(n+1)/2}$. This concludes the claim, using the second inequality in Theorem 3.4. □

Due to Section 2, the corollary means that one cannot determine the uniformness of weights of more than $\frac{(3^n-1)(n+1)^{(n+1)/2}}{2}$ coins, using n weighings. It is also easy to see that the value $\gamma(n)$ can be achieved by an $n \times (n+1)$ matrix. We believe that $\exp n \log n$ is the right order of magnitude of $\gamma(n)$ and of $f(n)$ (Conj. 4.2). There are $n \times (n+1)$ matrices L, where the nominator of $g(L)$ already has this order (for example, we can obtain one by adding a column to an $n \times n$ Hadamard matrix). However, it seems not so trivial to make the denominator small at the same time.

4. The algorithms which perform better than 2^n

In this section we will construct vectors from the minimal Hilbert basis H with the sum of coordinates larger than 2^n. It will then, through the bijection established in Section 2, result in algorithms which perform better than 2^n.

The first non-trivial example is for $n = 3$, $m = 10$. Consider the matrix

$$M = \begin{pmatrix} -1 & 1 & 1 & -1 \\ 0 & -1 & 1 & -1 \\ 1 & 0 & -1 & -1 \end{pmatrix}$$

The rank of M is equal to 3, hence the kernel of M is a line. In fact this line is spanned by the vector $v = (4, 2, 3, 1)$. If we properly complete v with zeroes it will lie in the polyhedral cone K (for $n = 3$). In fact, by Lemma 3.3 it is a minimal generator.

Now if we wish to reconstruct the algorithm, all we have to do is to choose the A_i's and B_i's properly. Our choice is illustrated below.

$$S = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\},$$

$$\begin{aligned} A_1 &= \{1, 2, 3, 4, 10\} & B_1 &= \{5, 6, 7, 8, 9\}, \\ A_2 &= \{1, 2, 3, 4\} & B_2 &= \{7, 8, 9, 10\}, \\ A_3 &= \{7, 8, 9\} & B_3 &= \{5, 6, 10\}. \end{aligned}$$

Let us now observe the following fact. If we can solve the problem for m_1 coins in n_1 steps and for m_2 coins in n_2 steps, then we can solve it for $m_1 \cdot m_2$ coins in $n_1 + n_2$ steps. We do it in the following way. First we divide given $m_1 \cdot m_2$ coins into m_2 groups, m_1 coins in each group. Then, in n_1 steps, we decide whether all the coins in one of these groups are equal or not. Then, we pretend that each group is just a single coin. Since the weights of the coins are generic, then also the total weights in different groups are generic. This fact allows us in the next n_2 steps to make sure that the total weights of different groups are equal and so, because the weights of coins are generic, we can conclude that all the coins have the same weight.

We gave an example of an algorithm, which in 3 steps solves the problem for 10 coins, using the technique above we can solve the problem for 10^t coins in $3t$ steps. In terms of m and n we get $m = 10^{n/3} = (2.1544\ldots)^n$.

In order to improve this constant, all one has to do is to find proper matrices. This has been done with the help of computer. Below we give a table illustrating the best values we could achieve. We denote the maximum of the function g, that we could achive by $s(n)$. Values of $s(n)$ $n = 1, 2, 3, 4$ are equal to the actual values of $\gamma(n)$ and it is plausible to think that the same is true for $n = 5, 6, 7, 8$.

n	s(n)	c
1	2	2.0
2	4	2.0
3	10	2.1544
4	30	2.3403
5	114	2.5785
6	454	2.7723
7	2 234	3.0091
8	9 966	3.1609
9	48 490	3.3161
10	259 606	3.4788
11	1 471 258	3.6366
12	6 590 538	3.7003
13	42 021 372	3.8585
14	307 393 727	4.0389
15	2 132 870 658	4.1872

We end this section with the following conjecture.

Conjecture 4.1. *There exists a positive constant c and an algorithm, which in n steps solves the generic all-equal problem for at least* $\exp(cn \log n)$ *coins.*

The investigations above show that Conjecture 4.1 is equivalent to the following purely combinatorial open problem.

Conjecture 4.2. *There exists a series of matrices $(M_n)_{n=1}^{\infty}$, and a positive constant c such that*

1. *M_n is an $n \times (n+1)$ matrix of rank n;*
2. *the entries of M_n are all $1, -1$ or 0;*
3. *$g(M_n) \geq \exp(cn \log n)$, see 3.1 for the definition of the function g.*

References

[CHH] P.D. Chen, X.D. Hu, F.K. Hwang, *A new competitive algorithm for the counterfeit coin problem,* Information Processing Letters 51, 1994, 213-218.

[GP] F.R.Giles, W.R. Pulleyblank, *Total dual integrality and integer polyhedra,* Linear Algebra and Its Applications, 25, pp. 191–196, 1979.

[HH] X.D. Hu, F.K. Hwang, *A competitive algorithm for the counterfeit coin problem,* to appear.

[K] D.N. Kozlov, *Coins with arbitrary weights,* in preparation.

[LP] L. Pyber, *How to find many counterfeit coins ?,* Graph and Combinatorics 2, 1986, 173-177.

[Sc] A.Schrijver, *Theory of linear and integer programming,* Wiley-Interscience series in discrete mathematics, John Wiley & Sons Ltd., 1986.

Department of Mathematics, Royal Institute of Technology, S-100 44, Stockholm, Sweden and Department of Mathematics, Yale University, New Haven, CT 06520, USA

E-mail address: kozlov@math.kth.se