On a Search Problem in Multidimensional Grids

Miklós Ruszinkó*
Computer and Automation Reseach Institute of the
Hungarian Academy of Sciences
H-1518, Budapest POB 63, Hungary; e-mail: ruszinko@lutra.sztaki.hu

Gábor Tardos†
Mathematical Institute of the Hungarian Academy of Sciences
H-1364, Budepest, Pf. 127, Hungary; e-mail: tardos@cs.elte.hu

ABSTRACT

A search problem of G. O. H. Katona was solved in [3] where an unknown point $\mathbf{x}$ *in a 2-dimensional grid has to be located using queries of type "is* $\mathbf{x} = (x_1, x_2) \leq \mathbf{a} = (a_1, a_2)$*?". Here* $\mathbf{a}$ *is an arbitrary lattice point and* $\mathbf{x} \leq \mathbf{a}$ *means that* $a_i \leq b_i$ *(*$i = 1, 2$*). In the recent paper we consider the generalization of this problem for arbitrary dimension d.*

1. Introduction

The following model of the combinatorial search was initiated by A. Rényi. A finite set X and a family $\mathcal{A}$ of its subsets are given. We need to find an unknown element x in X by asking questions of type "is x in A?", for any member A of $\mathcal{A}$. Our goal is to minimize the number of questions to identify x in the worst or in the average case.

There are two different approaches of the described model. In the first we have to ask all queries in advance and from the answers we have to identify x. This is called *parallel search*. In the second case the queries are asked sequentially, all questions may depend on the previous answers. This is called *adaptive search*. In this paper we give a worst case analysis in the adaptive model for a special search problem.

Running title: Search in Grids. AMS classification number: 90B40. Key words and phrases: Search, grid.
* The author is partially supported by the Hungarian National Foundation for Scientific Research, Grant No. F014919, and a grant of the "Magyar Tudományért" Foundation.
† The author is partially supported by the NSF grant CCR-92-00788, Hungarian National Fund for Scientific Research (OTKA) grant T4271, and a grant of the "Magyar Tudományért" Foundation.

Consider the well known fact, that in the adaptive case it is sufficient and necessary to ask $\lceil \log n \rceil$ questions of type "is $x \le b$?" for some $b \in X$ to find an unknown element x in an n-element ordered set X. But what can we say if X is partially ordered rather than ordered? G. O. H. Katona [1] asked this problem for partial orders that are direct products of chains i. e. when X is the set of bounded integer vectors of a fixed dimension and we must find an unknown vector $\mathbf{x} \in X$ by asking queries of the form "is $\mathbf{x} \le \mathbf{b}$?". (See next section for formal definition.)

In [3] M. Ruszinkó considered this problem in dimension two and solved it in most cases. In this paper we consider the same problem in arbitrary dimension d. So now we want to find an unknown integer vector $\mathbf{x} = (x_1, \dots, x_d)$ bounded by $1 \le x_i \le a_i$ for $i = 1, \dots, d$. The queries allowed are of the form "is $\mathbf{x} \le \mathbf{b}$?" for an integer vector $\mathbf{b} = (b_1, \dots, b_d)$ where answer "yes" means $x_i \le b_i$ for all $i = 1, \dots, d$. It is easy to see that it is necessary to ask $\lceil \sum_{i=1}^{d} \log a_i \rceil$ questions and it is sufficient to ask $\sum_{i=1}^{d} \lceil \log a_i \rceil$ queries (see Proposition 2.1). Thus the gap between these two trivial bounds is at most $d-1$. But which bound is closer to the correct value?

Chapter 2 contains the notations, definitions and trivial cases. In Chapter 3 we improve on the trivial upper bound and prove that in "most" lattices the lower bound is tight. On the other hand we show, that for a certain sequence of lattices $\Omega(d)$ more queries are needed than the trivial lower bound. In Chapter 4 we prove a theorem for the special 2-dimensional case generalizing Theorem 3.3 of [3] giving lots of examples where the lower bound is not tight. In Chapter 5 some open problems are posed.

2. Notations, Definitions, Trivial Cases

For $c \in \mathbf{R}$ we denote by $\lceil c \rceil$ the smallest integer $\ge c$, and by $\lfloor c \rfloor$ the largest integer $\le c$, $\{c\} = c - \lfloor c \rfloor$. We denote by log the logarithm of base 2.

Let $d \ge 1$ be a fixed integer and let us consider d-dimensional vectors $\mathbf{a} = (a_1, \dots, a_d)$. All vectors considered in this paper are integer vectors. We say that $\mathbf{a} \le \mathbf{b}$ if $a_i \le b_i$ for every $i = 1, \dots, d$. Let us denote the coordinate-wise product of two vectors $\mathbf{a}$ and $\mathbf{b}$ by $\mathbf{ab}$. Let $\mathbf{1} = (1, \dots, 1)$. We consider the following search problem for a vector $\mathbf{a} \ge \mathbf{1}$.

> **Problem.** An unknown integer vector $\mathbf{1} \le \mathbf{x} \le \mathbf{a}$ has to be found by asking questions of the form "is $\mathbf{x} \le \mathbf{b}$?" for an arbitrary integer vector $\mathbf{b}$. The questions can be asked adaptively, i. e. later questions can depend on the answers for earlier questions. What is the minimum number of questions needed to find the unknown vector $\mathbf{x}$ in the worst case?

Let us denote this minimum by $f(\mathbf{a})$, i. e. it is the minimum for all search strategies of the maximum for all vectors $\mathbf{1} \le \mathbf{x} \le \mathbf{a}$ of the number of queries used to find $\mathbf{x}$.

Proposition 2.1. *For any* $\mathbf{a} = (a_1, \ldots, a_d)$ *the following inequalities hold*

$$\left\lceil \sum_{i=1}^{d} \log a_i \right\rceil \le f(\mathbf{a}) \le \sum_{i=1}^{d} \lceil \log a_i \rceil.$$

Proof. The lower bound follows from the observation that the number of queries has to be at least the logarithm of size of the set we are searching in and in this case it is $\prod_{i=1}^{d} a_i$. A binary search for each coordinate of $\mathbf{x}$ yields the upper bound. ∎

Let us denote the lower bound $\lceil \sum_{i=1}^{d} \log a_i \rceil$ by $t(\mathbf{a})$. We call a vector $\mathbf{a}$ *loose* if this bound is not tight, i. e. if $f(\mathbf{a}) > t(\mathbf{a})$.

In the following section we prove stronger bounds on the function $f(\mathbf{a})$.

3. Tighter upper and lower bounds

The following theorem says that the gap between f and the trivial lower bound can be large. Let us denote the d-dimensional vector with all coordinates 3 by $\mathbf{3}_d$.

Theorem 3.1. $f(\mathbf{3}_d) = 2d$.

The main idea of the proof is that in the middle of the search, instead of dealing with the set of vectors $\mathbf{x}$ consistent with the answers so far we deal with a "nice" subset of it.

Definition. We call a set S of d dimensional integer vectors *nice* if it is a direct product of factors of the following type:

(0) A single point $\{2\}$ or $\{3\}$ in a coordinate,

(1) Two points $\{1, 2\}$ or $\{2, 3\}$ in a coordinate,

(2) The set $\{1, 2, 3\}$ in a coordinate,

(3) The five element set $\{(1, 3), (2, 3), (3, 1), (3, 2), (3, 3)\}$ in two coordinates.

We define the value of a factor of type (i) to be i for $i = 0, 1, 2, 3$ and the *value* $v(S)$ of a nice set S to be the sum of the values of its factors. The proof will be easy after the following lemma.

Lemma 3.2. *Let S be a nice set of lattice points. For any query of the form "is $\mathbf{x} \le \mathbf{b}$?" there is a nice subset T of S such that for every element $\mathbf{x} \in T$ the question is answered in the same way and $v(T) \ge v(S) - 1$.*

Proof. We call a factor of the nice set S trivial if $\mathbf{b}$'s projection to it is $\ge$ than all elements of the factor. The proof is a case analysis. We define T separately for the following seven cases. While some of these cases overlap (then either definition of T satisfies the conditions of the lemma) it is easy to see that together they cover all possibilities.

(a) In case all the factors are trivial we define $T = S$.

(b) If a factor of type (0) is nontrivial we also define $T = S$.

(c) If a factor of type (1) is nontrivial we define T by replacing this factor in S by the one element set $\{2\}$ if it was $\{1, 2\}$ or by $\{3\}$ if it was $\{2, 3\}$.

(d) In case a factor of type (3) is nontrivial we define T by replacing this factor of S by a factor $\{3\}$ (of type (0)) in the coordinate $\mathbf{b}$'s projection is less than 3 and by a factor of type (2) in the other coordinate.

(e) If at least two different factors of type (2) are nontrivial then we define T by replacing two such factors of S by a factor of type (3).

(f) If $\mathbf{b}$'s projection to a factor of type (2) is ≤ 1 then we define T by replacing this factor of S by $\{2, 3\}$.

(g) Finally, if all the factors are trivial except for one type (2) factor to which $\mathbf{b}$'s projection is 2 we define T by replacing this factor of S by $\{1, 2\}$, a factor of type (1).

In the first two cases $T = S$, therefore $v(T) = v(S)$, while in the last five cases we have $v(T) = v(S) - 1$. In the first and last cases the answer to the question "is $\mathbf{x} \leq \mathbf{b}$?" is positive for any $\mathbf{x} \in T$ while in the other five cases the answer is negative for all $\mathbf{x} \in T$. ∎

Proof of Theorem 3.1. Proposition 2.1 gives $f(\mathbf{3}_d) \leq 2d$. For the lower bound we give an adversary argument. We start with the set of all possible vectors $\mathbf{x}$. This is a nice set of value $2d$. After any question and answer we consider a nice subset of this that is consistent with every answer so far. By Lemma 3.2. we can answer the queries such that the value of the nice set considered decreases by at most 1. This means that after less than $2d$ queries the value is positive so we have more than one possible vectors $\mathbf{x}$ consistent with all the answers, so more queries are needed. Therefore $f(\mathbf{3}_d) \geq 2d$. ∎

Theorem 3.1. gives an example where the value of the function f is large, it is $t + \lfloor (2 - \log 3)d \rfloor$. Here $2 - \log 3 = 0.415\ldots$ Theorem 3.4 states that the difference cannot be much larger. We are going to use the following lemma from [3].

Lemma 3.3. *[3, Theorem 3.2.] If* $\{\log a_1\} + \{\log a_2\} \leq 0.8$ *then* $f(a_1, a_2) = \lceil \log a_1 + \log a_2 \rceil$. ∎

Theorem 3.4. *For any d-dimensional vector* $\mathbf{a} \geq \mathbf{1}$ *we have* $f(\mathbf{a}) < t(\mathbf{a}) + 0.6d$.

Proof. Let $\mathbf{a} = (a_1, \ldots, a_d)$. We prove the statement by induction on d using Lemma

3.3.

For $d = 1, 2$ the inequality obviously holds.

In case $d > 2$ let us first suppose, that $\{\log a_1\} + \{\log a_2\} \le 0.8$. Then

$$f(\mathbf{a} = a_1, \ldots, a_d) \le f(a_1, a_2) + f(a_3, \ldots, a_d)$$

$$\le \lceil \log a_1 + \log a_2 \rceil + \left\lceil \sum_{i=3}^{d} \log a_i \right\rceil + 0.6(d-2) < \left\lceil \sum_{i=1}^{d} \log a_i \right\rceil + 0.6d$$

Let us now consider the case when for any pair of components i, j $(1 \le i < j \le d)$ we have $\{\log a_i\} + \{\log a_j\} > 0.8$. In this case the gap between the upper and lower bounds of Proposition 2.1. is obviously less than $0.6d$. ∎

The following theorem asserts that the gap between $f(\mathbf{a})$ and $t(\mathbf{a})$ is small for vectors $\mathbf{a}$ with all coordinates large.

Theorem 3.5. *For every dimension $d \ge 1$ and every $\epsilon > 0$ there is an $n > 0$ such that for any integer vector $\mathbf{a} = (a_1, \ldots, a_d)$ with all coordinates greater than n we have $f(\mathbf{a}) \le \lceil \sum_{i=1}^{d} \log a_i + \epsilon \rceil$.*

The proof uses nothing but the monotonity of f and the following two simple observations. The first of the observations is noted in [3]. We present a simpler proof here to be self-contained.

Lemma 3.6. *[3, Theorem 3.1.3.] For the two-dimensional vector $\mathbf{v} = (2^k - 1, 2^k + 1)$ where $k \ge 1$ integer we have $f(\mathbf{v}) = 2k$.*

Proof. As $2k$ is the lower bound in Proposition 2.1 it is enough to give a strategy that finds the unknown vector $\mathbf{x} = (x_1, x_2)$ asking at most $2k$ queries. We ask queries "is $\mathbf{x} \le \mathbf{b}$?" with $\mathbf{b} = (2^k - 2^{k-i}, 2^k)$ for $i = 1, 2, \ldots, k$ but only till we hear the first answer "yes". If this happens after the i^{th} question then $2^k - 2^{k-i-1} < x_1 \le 2^k - 2^{k-i}$ and so it can be found in $k - i$ queries by binary search, while $1 \le x_2 \le 2^k$ can be found in k queries by binary search. In case all answers are negative to our first k queries, than $x_2 = 2^k + 1$ and $1 \le x_1 \le 2^k - 1$ can be found in k queries by binary search. ∎

Recall that we denote by $\mathbf{ab}$ the coordinate-wise product of the vectors $\mathbf{a}$ and $\mathbf{b}$.

Lemma 3.7. *For arbitrary d-dimensional vectors $\mathbf{a}$ and $\mathbf{b}$ we have $f(\mathbf{ab}) \le f(\mathbf{a}) + f(\mathbf{b})$.*

Proof. Note that any vector $\mathbf{1} \le \mathbf{x} \le \mathbf{ab}$ can uniquely be written in the form $\mathbf{x} = (\mathbf{y} - \mathbf{1})\mathbf{a} + \mathbf{z}$ where $\mathbf{1} \le \mathbf{y} \le \mathbf{b}$ and $\mathbf{1} \le \mathbf{z} \le \mathbf{a}$. We can find first $\mathbf{y}$ in $f(\mathbf{b})$ queries asking queries of the form "is $\mathbf{x} \le \mathbf{ca}$?". Once we found $\mathbf{y}$ we can find $\mathbf{z}$ in $f(\mathbf{a})$

queries asking queries of the form "is $\mathbf{x} \le (\mathbf{y}-\mathbf{1})\mathbf{a}+\mathbf{c}$?". This search strategy establishes the claimed inequality. ∎

Proof of Theorem 3.5. Let us take a positive integer $k > \log(6d/\epsilon)$. Let $\mathbf{v}_{ik}$ be the d-dimensional vector with all 1 coordinates except for the first coordinate being 2^k-1 and the i^{th} coordinate being 2^k+1 ($i=2,\dots,d$). By Lemma 3.6 we have $f(\mathbf{v}_{ik})=2k$. Recall that $\{x\}=x-\lfloor x\rfloor$ and let $l_i=\lceil\{\log a_i\}/\{\log(2^k+1)\}\rceil$ and $\mathbf{w}=(2^{m_1},\dots,2^{m_d})$ where $m_i=\lceil\log a_i - l_i\log(2^k+1)\rceil$ for $i=2,\dots,d$ and $m_1=\left\lceil\log a_1-\sum_{i=2}^d l_i\log(2^k-1)\right\rceil$.

We know from calculus that

$$k+\log e/(2^k+1) < \log(2^k+1) < k+\log e/2^k \tag{1}$$

where e is the base of the natural logarithm. Therefore $\{\log(2^k+1)\}\ge\log e/(2^k+1)\ge 1/2^k$ and thus $l_i\le 2^k$. We choose $n=2^{dk2^k}$ to ensure that $\log a_i\ge dk2^k$ and therefore $m_i\ge 0$ and thus $\mathbf{w}$ is an integer vector.

We have $f(\mathbf{w})=\sum_{i=1}^d m_i$ since the upper and lower bounds in Proposition 2.1 are the same in this case. Let us define $\mathbf{b}=\mathbf{w}\prod_{i=2}^d \mathbf{v}_{ik}^{l_i}$ a coordinate-wise product of $\sum_{i=2}^d l_i+1$ vectors. By the definition of $\mathbf{w}$ we have $\mathbf{a}\le\mathbf{b}$ and therefore

$$f(\mathbf{a})\le f(\mathbf{b})\le 2k\sum_{i=2}^d l_i+\sum_{i=1}^d m_i$$

by the monotonity of f and Lemma 3.7. We claim that

$$2k\sum_{i=2}^d l_i+\sum_{i=1}^d m_i\le\left\lceil\sum_{i=1}^d\log a_i+\epsilon\right\rceil.$$

This is clearly enough to prove the theorem.

Consider the value $y_i=\log a_i-l_i\log(2^k+1)$ for $i=2,\dots,d$. By the definition of l_i and m_i and by (1) we have $\lceil y_i\rceil=m_i$ and $m_i-y_i\le\{\log(2^k+1)\}<\log e/2^k$. The i^{th} coordinate of $\mathbf{b}=(b_1,\dots,b_d)$ is $b_i=(2^k+1)^{l_i}2^{m_i}$ so $0\le\log b_i-\log a_i<\log e/2^k$ for $i=2,\dots d$. By the definition of m_1 for the first coordinate we have $0\le\log b_1-\log a_1<1$. The product of the coordinates of $\mathbf{b}$ is the product of the same for all factors. Taking logarithm we obtain

$$\sum_{i=1}^d\log b_i=\log(2^{2k}-1)\sum_{i=2}^d l_i+\sum_{i=1}^d m_i.$$

Here

$$\log(2^{2k}-1)>2k-\log e/(2^{2k}-1)$$

and

$$\sum_{i=2}^d l_i\le(d-1)2^k.$$

Putting all this together we get

$$2k\sum_{i=2}^{d} l_i + \sum_{i=1}^{d} m_i < \sum_{i=1}^{d} \log b_i + \log e(d-1)2^k/(2^{2k}-1)$$

$$< \sum_{i=1}^{d} \log a_i + \log e(d-1)/2^k + \log e(d-1)/(2^k-1) + 1$$

$$< \sum_{i=1}^{d} \log a_i + \epsilon + 1 \leq \left\lceil \sum_{i=2}^{d} \log a_i + \epsilon \right\rceil + 1.$$

The third inequality follows from the choice of k. Here the first and the last expressions are integers and therefore they differ by at least 1. This gives the inequality we need to finish the proof. ∎

The threshold value $n = n(d, \epsilon)$ we obtain from this proof is $(6d/\epsilon)^{6d^2/\epsilon}$. This can be improved to $(d/\epsilon)^{6d/\epsilon}$ with no effort.

Corollary 3.8. *For every dimension $d \geq 1$ there is an $n > 0$ such that for any vector $\mathbf{a}$ with all coordinates greater than n the trivial lower bound on $f(\mathbf{a})$ of Proposition 2.1 is off by at most one.*

Proof. Take $\epsilon = 1$ in Theorem 3.5. ∎

Corollary 3.9. *Let us fix the dimension d and let $\alpha(\mathbf{c})$ be the ratio of loose vectors $\mathbf{a}$ among all vectors $\mathbf{1} \leq \mathbf{a} \leq \mathbf{c}$. Then $\alpha(\mathbf{c})$ tends to 0 as all coordinates of $\mathbf{c}$ tend to infinity.*

Proof. Let us choose a small value $0 < \epsilon < 1$ and apply Theorem 3.5. It cannot be applied to vectors $\mathbf{a}$ with at least one coordinate smaller than the bound $n = n(d, \epsilon)$ in the theorem. Fortunately the ratio of these vectors with at least one small coordinate tend to 0. For the rest of the vectors $\mathbf{a} = (a_1, \ldots, a_d)$ the theorem implies that $\mathbf{a}$ is not loose unless $1 - \epsilon \leq \{\sum_{i=1}^{d} \log a_i\} < 1$. The ratio of the vectors satisfying this last inequality can be made arbitrarily small by choosing ϵ small enough. ∎

4. A note for the 2-dimensional case

Corollary 3.9 shows that the lower bound of Proposition 2.1 is almost always tight. There exist however plenty of loose vectors even in dimension 2. In [3] it was established that there are infinitely many such vectors. We extend this result in the following theorem which implies that for a positive fraction of all positive integers u there are infinitely many positive integers v such that the lower bound in Proposition 3.1 for the vector (u, v) is not tight. This result was independently achieved by E. Kolev and I. Landgev [2].

Theorem 4.1. *If u is a positive odd integer with $2 - \log 3 < \{\log u\} < \log 3 - 1$ then there are infinitely many positive integers v such that $f((u, v)) = \lceil \log u + \log v \rceil + 1$.*

Proof. Let u be an integer satisfying the conditions of Theorem 4.1. Take an integer $k \geq 1$ with $2^k \equiv 1 \pmod{w}$ simultaneously for all odd integers $1 \leq w \leq u$. We claim that $v = (2^k - 1)/u$ satisfies the statement of the theorem. This is enough for the proof since k can be chosen in infinitely many different ways. By Euler's theorem it can be any multiple of $u!$.

By Proposition 2.1 we have to prove only $f((u, v)) \geq \lceil \log u + \log v \rceil + 1 = k + 1$. Suppose that there exists a search strategy finding $\mathbf{x}$ in k queries. Since in $k - 1$ queries we cannot find an element of a set if it has more than 2^{k-1} elements, the first query "is $\mathbf{x} \leq (a, b)$?" has to be such that at most 2^{k-1} different vectors $\mathbf{x}$ leads to any given answer "yes" or "no". We may suppose here that $1 \leq a \leq u$ and $1 \leq b \leq v$. As the total number of possible vectors $\mathbf{x}$ is $uv = 2^k - 1$ this means that the number of vectors giving "yes" answer ab is 2^{k-1} or $2^{k-1} - 1$. In case $ab = 2^{k-1} - 1$ a is a divisor of $2^{k-1} - 1$ therefore it is odd. On the other hand by choosing v in the way above, we get, that a is a divisor of $2^k - 1$, too. Since these two numbers are co-primes, a has to be 1. Since ab is too small in this case we have to have the other possibility $ab = 2^{k-1}$. This case is possible but clearly in one way only, namely a has to be the largest power of 2 smaller than u and b has to be the largest power of 2 smaller than v. Let us note here that by assumption $2 - \log 3 < \{\log u\} < \log 3 - 1$ we have $u < (3/2)a$ and $v < (3/2)b$.

Consider what happens after an answer "no" to the first question. There are still $2^{k-1} - 1$ possible vectors $\mathbf{x}$. Out of these vectors 2^{k-2} or $2^{k-2} - 1$ will answer the next question with a "yes" by the same reasons as above. Let the next question be "is $\mathbf{x} \leq (c, d)$?". The following is a case analysis, we find the contradiction by showing that none of the cases can occur.

(a) In case $d \leq b$ the number of possible vectors $\mathbf{x}$ giving answer "yes" is 0 or $(c - a)d \leq (u - a)b < ab/2 - 1 = 2^{k-2} - 1$, which is a contradiction.

(b) In case $c \leq a$ the number of possible vectors $\mathbf{x}$ giving answer "yes" is 0 or $c(d - b) \leq a(v - b) < ab/2 - 1 = 2^{k-2} - 1$, which is a contradiction.

(c) Finally, the only remaining case is $c > a$ and $d > b$. The number of possible vectors $\mathbf{x}$ giving answer "yes" is $cd - ab$ therefore cd must be either $(3/4)2^k$ or $(3/4)2^k - 1$. In the latter case c divides $(3/4)2^k - 1$ and therefore it is odd. Since we chose v in the way above we have that c divides $2^k - 1$, too. As these two numbers are co-primes c has to be 1. Therefore cd cannot be big enough. This contradiction shows that cd has to be $(3/4)2^k$. But now c as a divisor of $(3/4)2^k$ has to be a power of 2 or 3 times a power of 2 but there is no such number $a < c \leq u$.

These contradictions prove the theorem. ∎

5. Open problems

Finally let us consider the following problems.

(1) We conjecture that if (u, v) is loose then there are infinitely many integers v' such that (u, v') is loose.

This conjecture is motivated by a case-analysis for small values of u outlined here. For $u \leq 10$ there are no loose vectors (u, v). The first value of u where Theorem 4.1 applies is $u = 11$. (We mention here that this case is already settled in [3].) There are no loose vectors (u, v) with $u = 12, 14, 15, 16, 17, 18, 20, 21$. For $u = 13, 19, 22$ and 23 there are infinitely many loose vectors (u, v). For $u = 13$ and 19 it is possible to ask the first two questions in such a way that the number of vectors $\mathbf{1} \leq \mathbf{x} \leq (u, v)$ consistent with any set of answers is $\leq 2^{\lceil \log uv \rceil - 2}$ i. e. satisfies the "counting criterion", but we cannot ask the third question this way if v is chosen right. For $u = 22$ and 23 even the second question has to violate the counting criterion for some v. (23 is the second value of u where Theorem 4.1 applies.)

The following stronger conjecture (motivated by the proof of Theorem 4.1) would imply the previous conjecture.

Let u, v, and v' be positive integers. Let $k = \lceil \log uv \rceil$ and $k' = \lceil \log uv' \rceil$. Suppose that for any $w \leq u$ we have $2^k \equiv 2^{k'} \pmod{w}$. If (u, v) is loose and $v < v'$ then (u, v') is also loose.

(2) Find $x = \limsup_{d \to \infty} \max_{\mathbf{a}} \frac{f(\mathbf{a}) - t(\mathbf{a})}{d}$. Here $\mathbf{a}$ is a positive integer vector and d is its dimension.

By Theorem 3.1 and Theorem 3.4 we have $2 - \log 3 \leq x \leq 0.6$. The upper bound here can be made much closer to 1/2 with little effort. We conjecture however that the lower bound gives the correct value of x, i. e. the vectors $\mathbf{3}_d$ of Theorem 3.1 are the ones for which the trivial lower bound is the worst.

References

[1] G. O. H. Katona, personal communication.

[2] E. Kolev and I. Landgev, On a two-dimensional search problem, manuscript (1994).

[3] M. Ruszinkó, On a 2-dimensional search problem, *Journal of Statistical Planning and Inference*, **37** (1993), 371–383.