The Guessing Secrets problem: a probabilistic approach

Alberto Del Lungo* Guy Louchard † Claudio Marini* Franco Montagna*

November 12, 2003

Abstract

We introduce a probabilistic variant of the Guessing Secrets problem proposed by Chung et al. in [2]. In our variation, a player tries to discover the identity of a set S of n unknown secrets drawn by a second player, from a space Ω of N secrets. The first player tries to learn as much as possible about the elements of S by asking binary questions. For each question asked, the second player randomly chooses one of the n secrets of S that he uses in supplying the answer, which in any case must be truthful. We define a simple probabilistic guessing algorithm that allows us to guess all secrets of S with probability one. We show that the expected number of questions needed to guess all secrets is $2n^2 \log_2 N$ and the expected time complexity of the algorithm is $\mathcal{O}(n^2 \log N)$. We also propose a generalization of this probabilistic guessing secrets problem, and provide some similar results for this generalization.

1 Introduction

Chung et al. [2] proposed an information-theoretic problem which is a variant of the familiar "20 questions" game. They called the problem *Guessing Secrets*, and it can be formulated in the following way: a player **A** (called the "Seeker") tries to discover the identity of a set S of $n \geq 2$ unknown "secrets" drawn by a second player **B** (called the "Adversary"), from a large finite subset Ω of N secrets. While Ω is known to both players, the set S of secrets is known only to the adversary **B**. The Seeker **A** tries to learn as much as possible about the elements of S by asking **B** binary (Yes/No) questions. Each question takes the form: "Is your number (i.e., the secret you are referring to) in X?" for some subset X of Ω. For each question asked, **B** gets to adversarially choose which of the n secrets to use in supplying the answer, which in any case must be truthful. This restriction imposes that, if all the secrets give the same answer on a given question (i.e. if either all the secrets are in X, or none of them is in X), then **B** must answer accordingly, else **B** can give any answer to a given question. For example, suppose that $\Omega = \{1, 2, 3, 4, 5\}$ and $S = \{1, 2\}$. If the question of **A** is: "is your element less than three ?", **B** must answer "Yes", since the answer is "Yes" for both. On the contrary, if the question of **A** is: "is your element an even number ?", **B** could answer on behalf of 1 and say "No" or on behalf of 2 and say "Yes".

We point out that, if **B** chooses the same secret to answer every question **A** asks, **A** would never know anything about any other elements of S. Moreover, **A** cannot even hope to determine with certainty one secret of **B**. Indeed, **B** may use the following strategy: let $u \notin S$; then **B** answers YES to any question of the form "Is your element in X?" iff $X \cap (S \cup \{u\})$ has at least two elements. Clearly, this strategy is allowed by the rules of the game. Moreover, if **B** uses this strategy, it is impossible for **A** to distinguish u from the elements of S.

However, although **A** cannot learn the identity of any particular element of S, with appropriate questions it is possible to learn a lot about S. For instance, when S contains $n = 2$ secrets, **A** either

*Scienze Matematiche e Informatiche "Roberto Magari" Università di Siena, Via del Capitano 15, 53100, Siena, Italy. [marinic, montagna]@unisi.it

†Université Libre de Bruxelles, Département d'Informatique, CP 212, Boulevard du Triomphe, B-1050 Bruxelles, Belgium, louchard@ulb.ac.be. This paper was nearly completed when Alberto suddenly died on June 1, 2003. We present this work as a tribute to his memory.

learns one of the secrets explicitly, or **A** learns a set of three elements which contains both of the secrets. More generally, as shown in [2], when S contains $n \geq 2$ secrets, **A** can always choose a set of questions so that no matter how **B** selects the answers, the set of possible n-elements sets of secrets consistent with all the answers forms an intersecting family of n-sets of Ω (i.e., an intersecting k-uniform hypergraph), and this is the maximum amount of information about S that **A** can achieve. So, the Guessing Secrets problem concerns the determination of the maximum amount of information about S that can be recovered and the definition of an optimal strategy for selecting questions. For the case with two secrets (i.e., $n = 2$), there are several strategies with the optimal $\mathcal{O}(\log N)$ questions together with poly$(\log N)$ time algorithms to recover the secrets (see [1, 3, 10]). On the other hand, it is shown in [3] that the Guessing Secrets problem becomes intractable (NP-hard) when the number of elements of S is bigger than 2. This means that (unless P=NP) exact solutions of the problem require, in general, an exponential amount of time. In polynomial time only approximate solutions can be expected [1].

The computer scientists have discovered that Guessing Secrets problems define a class of interesting combinatorial problems which has arisen in connection with various topics in computer science and transmission of digital data as separating systems into smaller units, diagnosing technical problems, protecting data from unauthorized reproduction, authenticating ownership claims [12]. Chung et al. [2] point out some applications to the World Wide Web. For instance, let us consider a client computer with two or more internet protocol addresses (i.e., the secrets) either of which may be used at any moment. The focal point is to know how much we can hope to learn about both secrets (IP addresses) from yes-no answers and how quickly we can do it. The environment itself seems to suggest that the address used by the client at a given moment could be chosen at random according some particular probability distribution.

The above observation suggests a probabilistic variant of the Guessing Secrets problem. This variant differs from the classical Guessing Secrets problem in that for each question asked by player **A**, player **B** instead of choosing a secret adversarially, has to choose his secret *at random* (with uniform distribution), and then he has to answer truthfully, according to the secret chosen. So, **B** performs a random choice instead of a possible malevolent one. Thus e.g. if $S = \{1, 2, 5, 7\}$ and the question is "Is your number odd?", according to our paradigm **B** answers YES with probability $\frac{3}{4}$ and NO with probability $\frac{1}{4}$, whereas according to the rules of the classical Guessing Secrets game **B** is absolutely free to answer YES or NOT. The only assumption we consider is that the number of secrets of X is $o(N)$, that seems to be reasonable in many applications as the "list decoding" [1].

The Guessing Secret problem has some analogies with the Group Testing problem, even though there are relevant differences. Group Testing was introduced first by Dorfman in order to detect a few cases of syphillis among a big group of draftees. The idea was to test them in groups instead of one by one. Since then, Group Testing has been applied in many branches of sciences, e.g. in the DNA library screening, namely, the problem of determining which clone from the library contains which probe from a given collection, cf. e.g. [11].

A mathematical formulation of the problem is the following: let N be the number of items to be classified, and n be the number of defectives. Then we may test any group of items and the test tells us if there is at least one defective among the group. The problem, given N and n, is to compute the minimum number of tests necessary to detect all defectives. There are two kinds of procedures: an adaptive one, where the choice of groups to be tested may depend on the results of the previous tests, and the non-adaptive one, where the groups to be tested must be decided at the beginning. We refer to [4] for a general treatment of combinatorial group testing, and to [11] for a short survery.

The probabilistic Guessing Secrets problem seems to be close to the adaptive Group Testing, but in the Group Testing the test always tells you if one 'secret' (or defective element) is in the group you are testing, whereas in the probabilistic Guessing Secrets problem, when the set you are testing contains at least one secret, the test tells "YES" only with probability $\frac{\nu}{n}$, where ν is the number of secrets in the set, and n is the total number of secrets. Still, it is possible that the connections between the two games deserve a closer investigation.

The paper is organized as follows: in Section 2 we propose a simple probabilistic guessing algorithm

(Algorithm 1) based on binary search strategy that allows us to guess all secrets of S with probability one, and we prove an easy upper bound of the form $2n^2 \log_2 N$ (where N is the cardinality of Ω and n is the cardinality of S) for the expected value of the number of questions needed to guess all the secrets. In the whole paper we assume that $n \ll N$. In Section 3 we investigate a variant of Algorithm 1 (called Algorithm 2), we show that the number of questions needed to guess all the secrets (also called "total cost") is asymptotically a Gaussian RV of the form $\mathcal{N}(n^2 \log_2 N, n^3 \log_2 N)$. Then in Section 4 we investigate the asymptotic cost of Algorithm 1 (again a Gaussian distribution, of the form $\mathcal{N}(2n^2 \log_2 N, 4n^3 \log_2 N)$). In Section 5 we consider a generalization of this probabilistic guessing secrets in which for each question of the seeker, the adversary randomly chooses a set C of ρ secrets of S, with $\rho < n$, and he must answer "Yes" if and only if the answer is "Yes" for at least one element of C. In Section 6 we tackle this generalization by using the same strategy (Algorithm 3), and we obtain that if $\rho \ll n$ the number of questions needed to guess all secrets is $\mathcal{N}(2n^2/\rho \log_2 N, 4n^3/\rho^2 \log_2 N))$ and the expected time complexity of the algorithm is $\mathcal{O}(n^2/\rho \log N)$. We also investigate the case $\rho = \mathcal{O}(n)$. Finally in Section 7 we have implemented the algorithms, and the simulation corroborates the theoretical analysis.

2 A probabilistic variant of Guessing Secrets

In this section we introduce a probabilistic variant of the Guessing Secrets problem, and we propose a simple probabilistic guessing algorithm based on binary search. Unlike the deterministic algorithms, our algorithm allows us to guess all secrets with probability one.

The rules of the probabilistic variant. Let Ω be the search space, let N be its cardinality, let $S \subseteq \Omega$ be the set of secrets, and let n be the cardinality of S. Then:

- As before, **A** may ask questions of the form: *Is the secret in X?*, where X can be any subset of Ω.
- **B** chooses one secret $x \in S$ *at random* (with respect to the uniform distribution). If $x \in X$, then **B** answers "Yes", otherwise he answers "No".
- **A** wins the game if he correctly guesses all the secrets.

The algorithm is as follows.

The algorithm 1. At every step, the search space is a subset X of Ω. We choose XL as subset of X, half the size of X, that we denote by $\lfloor X/2 \rfloor$, and $XR = X \setminus XL$. The description of the algorithm is given in Algorithm 1.

Lemma 2.1 *With probability* 1 *the algorithm 1 stops with a correct answer.*

Proof. By induction, it is readily seen that until the algorithm stops, we have $X \cap S \neq \emptyset$. Thus either $XL \cap S \neq \emptyset$ or $XR \cap S \neq \emptyset$. It follows that at least one of the questions *Is the secret in XL?* and *Is the secret in XR?* has a YES-answer with probability at least $\frac{1}{n}$. Thus the probability of a YES answer after $2k$ questions is at least $1 - (1 - \frac{1}{n})^k$. Hence with probability 1 we will get a YES answer sooner or later. After $\log N$ positive answers the temporary search space X is a singleton and a new secret has been found. Hence with probability 1 we will eventually find all the secrets. ■

The expected time $E(Y)$ for one positive answer is bounded by $2\sum_{i=1}^{\infty} i\frac{1}{n}\left(1 - \frac{1}{n}\right)^{i-1} = 2n$. Hence the expected number of questions necessary to guess one secret is bounded by $2n \log_2 N$. Therefore we immediately get a rough upper bound for the expected number of questions necessary to win the game.

Theorem 2.2 *The expected value of the number of questions needed to guess all secrets is bounded by* $2n^2 \log_2 N$.

Algorithm 1

```
Input: the search secret space Ω
Output: the secret set S
X := Ω, Y := ∅;
repeat
   X := Ω − Y;
   repeat
      XL := ⌊X/2⌋; XR := X \ XL;
      repeat
         B picks x in S uniformly at random;
         if x ∈ (XL) then
            Answer:=Yes
         else
            B picks x in S uniformly at random;
            if x ∈ (XR) then
               Answer:=Yes
            end if
         end if
      until Answer:=Yes
      if x ∈ XL then
         X := XL
      else
         X := XR;
      end if
   until |X| = 1
   Y := Y ∪ X;
until |Y| = n
```

3 The analysis of a first version of the algorithm

In this section we perform a careful probabilistic analysis of a slightly simplified version of the algorithm: Algorithm 2.

Algorithm 2

Input: the search secret space Ω
Output: the secret set S
$X := \Omega$, $Y := \emptyset$;
repeat
  $X := \Omega - Y$;
  repeat
    $XL := \lfloor \frac{X}{2} \rfloor$; $XR := X \setminus XL$;
    repeat
      B picks x in S uniformly at random;
      if $x \in (XL$ or $XR)$ **then**
        Answer:=Yes
      end if
    until Answer:=Yes
    if $x \in XL$ **then**
      $X := XL$
    else
      $X := XR$;
    end if
  until $|X| = 1$
  $Y := Y \cup X$;
until $|Y| = n$

We summarize the notations we will use in the sequel:

$$\begin{aligned} n &= |S|, \\ \tau &= |Y|, \\ N &= |\Omega|, \\ G &= N - \tau, \\ j &= \text{ number of secrets in } XL, \\ \nu &= n - \tau. \end{aligned}$$

We recall the assumption we make: $1 \ll n = o(N)$.

3.1 Some known results

Let an urn contain M balls, ν of them black, and $M - \nu$ white. Draw, without replacement, α balls. The probability that j balls among α are black is given by

$$P_1(M, \nu, \alpha, j) = \frac{\binom{\nu}{j}\binom{M-\nu}{\alpha-j}}{\binom{M}{\alpha}}, \quad j = \sup\{0, \alpha - (M - \nu)\} \ldots \inf\{\nu, \alpha\}.$$

If $\alpha = \mathcal{O}(M) = \gamma M$, say, $M \gg 1$ and $\nu = o(M)$, then we easily derive that j has a Binomial distribution, with parameters ν, γ: $j \sim \mathrm{BIN}(\nu, \gamma)$. Indeed, by Johnson and Kotz [6], p.81, we know that the $r - th$ factorial moment of j is given by $\alpha^{(r)}\nu^{(r)}/M^{(r)}$ (where $x^{(r)}$ denotes the falling factorial of x). For large M this leads to $\gamma^r \nu^{(r)}$ which is exactly the $r - th$ factorial moment of the $\mathrm{BIN}(\nu, \gamma)$ distribution.

If the set X contains $\nu = n - \tau$ secrets, the probability that a random secret x drawn uniformly among n is included in X is given by $p(\nu) = \nu/n$ and the number i of questions before getting a positive answer is geometric with parameter p: $i \sim \mathrm{GEOM}(p)$, with mean $1/p$ and variance q/p^2.

Also, it is well known that, if $p \ll 1, \mathrm{GEOM} \sim \mathrm{EXP}(p)$, an exponential random variable (RV) with parameter p.

3.2 Complexity of algorithm 2

A moment of reflexion shows that the cost of finding one secret is made of two parts: in Phase 1 we first divide X until it contains exactly one secret, this costs c_1 questions. After that, in Phase 2, we divide again X until its size becomes 1. This costs c_2 questions. Indeed, when X contains exactly one secret, the probability of a question to yield a positive answer is constant, while in Phase 1, this probability depends on the number of secrets in X.

Fix τ. Set $\nu := n - \tau, G := N - \tau$. For simplicity assume that $G = 2^k$ for some integer k. In this section, we will first present a preliminary analysis of c_1, then compute the mean of c_1. Next we will derive an approximation of c_1 distribution, followed by an analysis of c_2 . Finally, we will give the total cost distribution, characterized by a Gaussian. It appears that the contribution of c_1 is negligible compared with that of c_2.

3.2.1 Preliminary analysis of c_1 (Phase 1)

In the remaining of the paper, an upper index denotes the step numbering. We have a sequence of RV $j^{(0)}, j^{(1)} \ldots$, with laws

$$P_1\left(G, \nu^{(0)}, \frac{G}{2}; j^{(0)}\right), P_1\left(\frac{G}{2}, \nu^{(1)}, \frac{G}{4}, j^{(1)}\right), \ldots \tag{1}$$

and

$$\begin{aligned}
\nu^{(i)} &= \text{number of secrets in } X \text{ at step } i, \\
\nu^{(0)} &= \nu, \\
\nu^{(1)} &= j^{(0)} \text{ if } x \in XL^{(0)}, \\
&= \nu^{(0)} - j^{(0)} \text{ if } x \in XR^{(0)}, \\
\nu^{(2)} &= j^{(1)} \text{ if } x \in XL^{(1)}, \\
&= \nu^{(1)} - j^{(1)} \text{ if } x \in XR^{(1)}, \\
&\ldots
\end{aligned}$$

As $n = o(N), j^{(i)}$ is asymptotically $\mathrm{BIN}(\nu^{(i)}, 1/2)$. Also,

$$\Pr(\nu^{(1)} = j^{(0)}) = \Pr(x \in XL^{(0)}) = \frac{j^{(0)}}{\nu^{(0)}}, \Pr(\nu^{(1)} = \nu^{(0)} - j^{(0)}) = \Pr(x \in XR^{(0)}) = \frac{\nu^{(0)} - j^{(0)}}{\nu^{(0)}},$$

and more generally,

$$\Pr(\nu^{(i+1)} = j^{(i)}) = \Pr(x \in XL^{(i)}) = \frac{j^{(i)}}{\nu^{(i)}}, \Pr(\nu^{(i+1)} = \nu^{(i)} - j^{(i)}) = \Pr(x \in XR^{(i)}) = \frac{\nu^{(i)} - j^{(i)}}{\nu^{(i)}}.$$

To each $\nu^{(i)}$ is associated a success probability

$$p^{(i)} = p(\nu^{(i)}) = \frac{\nu^{(i)}}{n}, \tag{2}$$

and a cost with probability distribution $\mathrm{GEOM}(p^{(i)})$. The total number L_1 of subdivisions until $\nu^{(i)} = 1$ is a RV, the distribution of which must be investigated: asymptotically it is similar to the Trie distribution. (See for instance Louchard [8]). In the latter case, we divide a set of ν keys, uniformly distributed $[0,1]$, according to their successive bits. (This leads to a $\mathrm{BIN}(\nu,1/2)$ distribution). The subdivision is performed until a particular subset (for instance, we always choose the left subset) is empty. The number L of subdivisions is a RV such that, for large ν,

$$\Pr[L \leq k] \sim e^{-\nu/2^k} = e^{-1/2^\eta} = e^{-e^{-\eta\ln(2)}}, \tag{3}$$

where

$$\eta := k - \log_2(\nu) = k - \lfloor \log_2(\nu) \rfloor - \{\log_2(\nu)\}, \{u\} := \text{ the fractional part of } u,$$

which shows that the distribution is a periodic function of $\log_2(\nu)$, highly concentrated around its mean. It is related to the extreme-value distribution $\exp(-\exp(-x))$. The asymptotic moments of L are also periodic functions of $\log_2(\nu)$. They can be written as Harmonic sums which are usually analyzed with Mellin transforms: see Flajolet et al [5]. For instance, (denoting the variance by $\mathbb{V}$),

$$\begin{aligned} \mathbb{E}(L) &\sim \log_2(\nu) + \frac{\gamma}{\ln(2)} + \frac{1}{2} + \omega_1(\nu), \\ \mathbb{V}(L) &\sim \frac{\pi^2}{6\ln(2)^2} + \frac{1}{12} + \omega_2(\nu), \end{aligned} \tag{4}$$

where $\omega_i(\nu)$ are also periodic functions of $\log_2(\nu)$ with very small amplitude ($\leq 10^{-5}$). Actually, this amounts to choosing the left subset with probability $1/2$.

But here, we choose the left (resp. right) subset with probability

$$P_2(\nu,j) := \Pr(\nu^{(1)} = j) = \frac{j}{\nu} \text{ (resp } \frac{\nu-j}{\nu}),$$

and

$$P_2(\nu,\nu-j) = 1 - P_2(\nu,j).$$

This relation will be true in all further analysis. The probability $P_2(\nu,j)$ introduces a slight bias in P_3. This can be analyzed as follows: set

$$P_3(\nu,k) := \Pr[L_1 \leq k].$$

Then

$$\begin{aligned} P_3(\nu,1) &= \binom{\nu}{1}\frac{2}{2^\nu}P_2(\nu,1) = \frac{1}{2^{\nu-1}}, \\ P_3(\nu,2) &= \sum_0^\nu \binom{\nu}{j}\frac{1}{2^\nu}\left[P_2(\nu,j)P_3(j,1) + P_2(\nu,\nu-j)P_3(\nu-j,1)\right] \\ &= \sum_0^\nu \binom{\nu}{j}\frac{1}{\nu 2^\nu}\left[j\frac{1}{2^{j-1}} + (\nu-j)\frac{1}{2^{\nu-j-1}}\right] = \sum_0^\nu \binom{\nu}{j}\frac{2j}{\nu 2^\nu 2^{j-1}} = \left(\frac{3}{4}\right)^{\nu-1}. \end{aligned}$$

By induction, we easily derive

$$P_3(\nu,k) = \left(1 - \frac{1}{2^k}\right)^{\nu-1} \sim e^{-(\nu-1)/2^k}.$$

Hence the distribution of L_1 is asymptotically the same as that of L, (see (3)), with the same moments.

Another useful probability is given by $P_4(\nu,k,j)$, the probability distribution of $\nu^{(k)}$ after k subdivisions. We obtain the distribution of $\nu^{(1)}$ as

$$P_4(\nu,1,j) = 2\binom{\nu}{j}\frac{1}{2^\nu}P_2(\nu,j) = 2\binom{\nu}{j}\frac{1}{2^\nu}\frac{j}{\nu},$$

and, of course, $\sum_0^\nu P_4(\nu,1,j)=1$. Also,

$$\mathbb{E}(\nu^{(1)})=\sum_0^\nu 2\binom{\nu}{j}\frac{1}{2^\nu}\frac{j^2}{\nu}=\frac{\nu+1}{2},$$

which entails

$$\mathbb{E}(\nu^{(k)})=\frac{\nu}{2^k}+1-\frac{1}{2^k}.$$

More generally,

$$P_4(\nu,k,j)=\sum_{l=0}^{\nu}2\binom{\nu}{l}\frac{1}{2^\nu}P_2(\nu,l)P_4(l,k-1,j)=\sum_{l=0}^{\nu}2\binom{\nu}{l}\frac{1}{2^\nu}\frac{l}{\nu}P_4(l,k-1,j).$$

The reader will recognize another Trie-like recurrence. Note that, at the end of Phase 1, G is of order $N/2^{\log_2(\nu)}=N/\nu\gg 1$.

The cost c_1 is now given by

$$c_1=\sum_{i=0}^{L_1}\mathrm{GEOM}(p^{(i)}).$$

3.2.2 The mean of c_1

Starting with ν secrets, let us denote

$$E(\nu):=\mathbb{E}(c_1(\nu)).$$

We have

$$\begin{aligned}E(\nu) &= \sum_0^\nu\binom{\nu}{j}\frac{1}{2^\nu}\left[\frac{j}{\nu}E(j)+\frac{\nu-j}{\nu}E(\nu-j)\right]+\frac{n}{\nu}\\ &= \frac{n}{\nu}+\sum_0^\nu\binom{\nu}{j}\frac{2}{2^\nu}\left[\frac{j}{\nu}E(j)\right],\nu\geq 2,E(0)=0,E(1)=n.\end{aligned}$$

Set

$$H_\nu:=\frac{\nu E(\nu)}{n}.$$

This leads to

$$H_\nu=1+\sum_0^\nu\binom{\nu}{j}\frac{2}{2^\nu}H_j,\nu\geq 2,H_0=0,H_1=1.$$

This kind of recurrence has been fully analyzed in the litterature: see, for instance Szpankowski [13], Sec.7.6 for an excellent treatment. Taking the exponential generating function

$$H(z):=\sum_0^\infty\frac{H_\nu z^\nu}{\nu!},$$

we derive after standard algebra

$$H(z)=e^z-1-z+2H\left(\frac{z}{2}\right)e^{z/2}.$$

Set now

$$B(z)=H(z)e^{-z}\text{ with }B(z)=\sum_0^\infty\frac{b_\nu z^\nu}{\nu!},b_0=0,b_1=1.$$

We obtain

$$B(z)=A(z)+2B\left(\frac{z}{2}\right),A(z)=1-e^{-z}(1+z),a_0=0,a_1=0,a_j=\frac{(-1)^j(j-1)}{j!}.$$

This leads to

$$b_j = \frac{(-1)^j (j-1)}{1 - 1/2^{j-1}}, j \geq 2.$$

Hence

$$H_\nu = \sum_{k=2}^{\nu} \frac{\binom{\nu}{k}(-1)^k k}{1 - 1/2^{k-1}} - \sum_{k=2}^{\nu} \frac{\binom{\nu}{k}(-1)^k}{1 - 1/2^{k-1}} + \binom{\nu}{1} 1.$$

But now we can apply Rice's formula. According to Szpankowski [13], Sec. 6.5, we know that

$$\sum_{k=2}^{n+r} \frac{\binom{n+r}{k}(-1)^k \binom{k}{r}}{1 - 1/2^{k-1}} \sim \frac{1}{L} n[\ln n + \gamma - \delta_{r,0} + L^2/2] + nP_r(n) + e(n), r = 0, 1, n \to \infty$$

where

$$\begin{aligned} L &:= \ln 2, \\ P_r(n) &:= \text{ periodic function of } \log_2(n), \text{ with period 1, mean 0 and small amplitude,} \\ e(n) &= \mathcal{O}(1). \end{aligned}$$

This leads to

$$H_\nu \sim \nu \left(1 + \frac{1}{L}\right)$$

and

$$E(\nu) \sim n \left(1 + \frac{1}{L}\right). \tag{5}$$

We will see in section 3.3 that the contribution of c_1 is negligible compared with that of c_2.

3.2.3 An approximation of c_1 distribution

The distribution of c_1 seems intractable (even in asymptotic form). However, as the contribution of c_1 is negligible compared with that of c_2, a rough analysis can be done as follows. Approximately, each $\nu^{(i)}$ is of order $\nu^{(i-1)}/2 \sim \nu/2^i$ and, in first approximation, this leads, at subdivision i, to a GEOM($\frac{\nu}{n2^i}$), with mean $\frac{n2^i}{\nu}$. Hence

$$\mathbb{E}(c_1) \sim \frac{n}{\nu} \sum_{i=0}^{L_1} 2^i \sim 2n, \tag{6}$$

independently of ν. This differs slightly from (5).

The dominant terms in c_1 are clearly given by $i = L_1 - \mathcal{O}(1)$ (i.e. small values of $\nu^{(i)}$) and are roughly given by a sum of GEOM RV, with geometrically decreasing mean (starting from n). Asymptotically, as n is large, we have a sum of exponential RV, with parameters $1/n, 2/n, 4/n \ldots$. Set

$$\bar{c_1} := \frac{c_1}{n},$$

then

$$\bar{c_1} \sim Y := \sum_{0}^{\infty} Y_k,$$

where Y_k is an exponential RV of parameter 2^k. The Laplace transform of the Y density is asymptotically given by

$$\varphi(\xi) = \prod_{i=0}^{\infty} \frac{2^i}{\xi + 2^i} = \prod_{i=0}^{\infty} \frac{1}{1 + \xi/2^i}.$$

The moments generating function of Y is easily derived: we will use one of the celebrated Euler identities[1] :

$$\prod_{k=0}^{\infty} \frac{1}{1-q^k z} = 1 + \sum_{i=1}^{\infty} \frac{z^i}{\prod_{k=1}^{i}(1-q^k)}.$$

Set $z = -\xi, q = 1/2$. This leads to

$$\varphi(\xi) = \sum_{i=0}^{\infty} \frac{(-\xi)^i}{Q(i)},$$

where

$$Q(j) := \prod_{k=1}^{j} \left(1 - \frac{1}{2^k}\right), \quad Q(0) := 1.$$

For instance[2],

$$\begin{aligned}
\mathbb{E}(Y) &= [-\xi]\varphi(\xi) = 2, \\
\mathbb{E}(Y^2) &= 2[\xi^2]\varphi(\xi) = \frac{16}{3}, \text{ hence} \\
\mathbb{V}(Y) &= \frac{4}{3},
\end{aligned}$$

hence

$$\mathbb{V}(c_1) \sim \frac{4}{3} n^2.$$

The distribution of Y is computed as follows. Let us first derive the density of $\sum_0^i Y_k$, where Y_k is an exponential RV of parameter 2^k. By induction, this density f_i is given by

$$f_i(x) = \sum_0^i \alpha[i,k] e^{-2^k x}. \tag{7}$$

with a distribution function

$$F_i(x) = \sum_0^i \alpha[i,k](1 - e^{-2^k x})/2^k. \tag{8}$$

Using

$$f_{i+1}(x) = \int_0^x f_i(x-u) 2^{i+1} e^{-2^{i+1}u}, f_0(x) = e^{-x},$$

the first values of α are given by

$$\alpha = \begin{pmatrix} 1 & & & \\ 2 & -2 & & \\ 8/3 & -4 & 4/3 & \\ 64/21 & -16/3 & 8/3 & -8/21 \end{pmatrix}.$$

Again, by induction, we derive the following recurrences:

$$\begin{aligned}
\alpha[i+1,k] &= \alpha[i,k] \frac{2^{i+1}}{2^{i+1} - 2^k}, k = 0 \ldots i, && (9) \\
\alpha[i+1,i+1] &= -\sum_{k=0}^{i} \alpha[i,k] \frac{2^{i+1}}{2^{i+1} - 2^k}, && (10) \\
\alpha[0,0] &= 1.
\end{aligned}$$

[1]For a simple proof, see Knuth [7], Ex.5.1.1/16

[2]We use the notation $[z^k]f(z)$ to denote the coefficient of z^k in the power expansion of $f(z)$.

We must have $\int_0^\infty f_i(x)dx = 1$, which entails

$$\sum_{k=0}^{i} \frac{\alpha[i,k]}{2^k} = 1. \tag{11}$$

This is easily checked as follows: assume (11) true for i. With (9) and (10), we compute, for $i+1$,

$$\sum_{k=0}^{i} \alpha[i,k]\frac{2^{i+1}}{(2^{i+1}-2^k)2^k} - \sum_{k=0}^{i} \alpha[i,k]\frac{2^{i+1}}{(2^{i+1}-2^k)2^{i+1}} = \sum_{k=0}^{i} \frac{\alpha[i,k]}{2^k} = 1,$$

by (11).

A quick examination of α values suggest two other relations:

$$\alpha[i,i] = 2^i \frac{(-1)^i}{R(i)}, \tag{12}$$

with

$$R(i) := \prod_{u=1}^{i} (2^u - 1), R(0) = 1,$$

and

$$\sum_{k=0}^{i} \alpha[i,k] = 0. \tag{13}$$

To prove (13), we write

$$\sum_{k=0}^{i+1} \alpha[i+1,k] = \sum_{k=0}^{i} \alpha[i,k]\frac{2^{i+1}}{2^{i+1}-2^k} - \sum_{k=0}^{i} \alpha[i,k]\frac{2^{i+1}}{2^{i+1}-2^k} = 0.$$

Let us now prove (12). Assume that it is true up to $i-1$. Then, by (9),

$$\alpha[i,k] = T(i-k)2^k \frac{(-1)^k}{R(k)}, k < i, \tag{14}$$

with

$$T(i) := \prod_{u=1}^{i} \frac{1}{(1-2^{-u})}, T(0) = 1, T(\infty) = 3.462746619\ldots$$

and (13) leads to

$$\sum_{k=1}^{i-1} T(i-k)2^k \frac{(-1)^k}{R(k)} + \alpha[i,0] + \alpha[i,i] = 0,$$

or

$$\sum_{k=0}^{i-2} T(i-k-1)2^{k+1} \frac{(-1)^{k+1}}{R(k+1)} + \alpha[i,0] + \alpha[i,i] = 0. \tag{15}$$

Similarly, (11) leads to

$$\sum_{k=0}^{i-2} T(i-k-1) \frac{(-1)^{k+1}}{R(k+1)} + \alpha[i,0] + \alpha[i,i]/2^i = 1. \tag{16}$$

Equations (15) and (16) give

$$\sum_{k=0}^{i-2} T(i-k-1) \frac{(-1)^{k+1}}{R(k)} + \alpha[i,i][1-1/2^i] = -1. \tag{17}$$

But, for $i-1$, (11) gives

$$\sum_{k=0}^{i-2} T(i-k-1)\frac{(-1)^k}{R(k)} + \frac{(-1)^{i-1}}{R(i-1)} = 1,$$

and (12) is readily checked. Note that (12) has now exactly the same form as (14).

As $i \to \infty$, we obtain the following result

Proposition 3.1 *The density of Y is given by*

$$f(x) = T(\infty)\sum_{k=0}^{\infty} 2^k \frac{(-1)^k}{R(k)} e^{-2^k x}, \tag{18}$$

and the distribution function is given by

$$F(x) = T(\infty)\sum_{k=0}^{\infty} \frac{(-1)^k}{R(k)}[1-e^{-2^k x}]. \tag{19}$$

The mean is given by

$$T(\infty)\sum_{k=0}^{\infty} \frac{(-1)^k}{R(k)2^k},$$

which is exactly 2 by (A.4) of Louchard [9].

3.2.4 Analysis of c_2 (Phase 2)

Now we have exactly one secret in X. The probability of a question to yield a positive answer is constant and given by $p = 1/n$. We divide X until $|X| = 1$, i.e., we perform $L_2 \sim \log_2(N-\tau) - \log_2(\nu)$ subdivisions, each leading to a GEOM$(1/n)$ cost. But asymptotically, this leads to a sum of L_2 exponential RV EXP$(1/n)$ i.e. a gamma distribution, with distribution function

$$1 - e^{-x/n}\sum_{i=0}^{L_2-1} \frac{(x/n)^i}{i!}, \tag{20}$$

mean $L_2 n$ and variance $L_2 n^2$

3.3 Total cost

We can now compute the total expected cost $\mathbb{E}(c)$: (we use here (6))

$$\begin{aligned}
\mathbb{E}(c) &= \sum_{\nu=0}^{n} \mathbb{E}(c_1+c_2) \sim 2n^2 + n\sum_{\tau=0}^{n-1}[\log_2(N-\tau) - \log_2(n-\tau)] \\
&\sim 2n^2 + \frac{n}{\ln 2}\left[(n-1)\ln N + (n-1) - \frac{(n-1)^2}{2N} - \frac{(n-1)^3}{2N^2} - n\ln n\right] \\
&\sim \frac{n^2}{\ln 2}\ln N.
\end{aligned}$$

This shows indeed that the contribution of c_1 is negligible compared with that of c_2. Also, as N gets large, the sum of gamma RV leads to a Gaussian distribution: we obtain the following proposition

Proposition 3.2 *The total cost of Algorithm 2 is asymptotically given by*

$$c \sim \mathcal{N}(n^2\log_2 N, n^3\log_2 N).$$

4 Algorithm 1

In this section, we give a complete analysis of Algorithm 1 . We recall that, here, the Adversary chooses a new x for each question. We analyze successively c_1, c_2 and the total cost.

4.1 Analysis of c_1 (Phase 1)

The following changes are in order. Assume that we have ν secrets in X, with j secrets in XL ($\nu - j$ secrets in XR). The success probability (2) is now given by

$$p(\nu, j) = 1 - \left(1 - \frac{j}{n}\right)\left(1 - \frac{\nu - j}{n}\right) = \frac{\nu}{n} - \frac{j(\nu - j)}{n^2}.$$

Note that

$$p(\nu, j) \sim \frac{\nu}{n}, \text{ independently of } j, \text{ if } \nu = o(n). \tag{21}$$

To obtain a fair choice between L and R, we flip a coin to decide whether the next question is related to L or R. We choose the left subset (with j secrets) with probability

$$P_2(\nu, j) := \Pr(\nu^1 = j) = \left[\frac{1}{2}\frac{j}{n} + \frac{1}{2}\left(1 - \frac{\nu - j}{n}\right)\frac{j}{n}\right] \Big/ p(\nu, j) = \left[\frac{j}{n} - \frac{1}{2}\frac{j(\nu - j)}{n^2}\right] \Big/ p(\nu, j).$$

Note again that

$$P_2(\nu, j) \sim \frac{j}{\nu}, \text{ if } \nu = o(n). \tag{22}$$

$P_3(\nu, k) := \Pr[L_1 \leq k]$ is given by

$$\begin{aligned}
P_3(\nu, 1) &= \binom{\nu}{1}\frac{2}{2^\nu}P_2(\nu, 1) = \binom{\nu}{1}\frac{2}{2^\nu}\left[\frac{1}{n} - \frac{1}{2}\frac{\nu - 1}{n^2}\right] \Big/ \left[\frac{\nu}{n} - \frac{\nu - 1}{n^2}\right], \\
P_3(\nu, 2) &= \sum_{j=0}^{\nu}\binom{\nu}{j}\frac{1}{2^\nu}[P_2(\nu, j)P_3(j, 1) + P_2(\nu, \nu - j)P_3(\nu - j, 1)] \\
&= 2\sum_{j=0}^{\nu}\binom{\nu}{j}\frac{1}{2^\nu}P_2(\nu, j)P_3(j, 1),
\end{aligned}$$

and, more generally,

$$P_3(\nu, k) = 2\sum_{j=0}^{\nu}\binom{\nu}{j}\frac{1}{2^\nu}P_2(\nu, j)P_3(j, k - 1). \tag{23}$$

Up to now, we have been unable to compute exactly this probability, but we conjecture that, asymptotically, the distribution is again given by (3). This is corroborated by the numerical checks in Sec.7. A clue to the proof could be the following remark: in the range R_1 where the Binomial is significant, i.e. $R_1 := [j = \nu/2 + \beta\sqrt{\nu}, \beta = \mathcal{O}(1)]$, the relative error $\Delta := [P_2(\nu, j) - j/\nu]\nu/j$ is asymptotically given by

$$\begin{aligned}
\nu = n &: \frac{2}{3}\frac{\beta}{\sqrt{n}} + \mathcal{O}(1/n), \\
\nu = n/2 &: \frac{2}{7}\frac{\beta}{\sqrt{n}} + \mathcal{O}(1/n), \\
\nu = n/4 &: \frac{4}{15}\frac{\beta}{\sqrt{n}} + \mathcal{O}(1/n),
\end{aligned}$$

so the distribution of $\nu^{(1)}$ is asymptotically the same in both algorithms. Of course, in the full range $R_2 := [j = \nu/2 + \beta\nu, \beta = \mathcal{O}(1)]$, the relative error is given by

$$\Delta = \frac{2\beta(2\beta - 1)}{4\beta^2 + 3}.$$

But the probability of being outside R_1 is exponentially negligible.

The probability distribution of $\nu^{(k)}$ after k subdivisions is now given for $k=1$ by

$$P_4(\nu,1,j) = 2\binom{\nu}{j}\frac{1}{2^\nu}P_2(\nu,j),$$

and $\sum_0^\nu P_4(\nu,1,j)=1$. Also,

$$\mathbb{E}(\nu^{(1)}) = \sum_0^\nu 2\binom{\nu}{j}\frac{1}{2^\nu}P_2(\nu,j)j, \tag{24}$$

Up to now, we could not obtain a closed form for this expression. However, an asymptotic series for large n is easy to obtain:

- Fix ν. We know that small values of $\nu^{(i)}$ lead to dominant terms at the end of Phase 1. We derive

$$\mathbb{E}(\nu^{(1)}) = (\nu+1)/2 + \frac{\nu^2-3\nu+2}{8\nu n} + \frac{(\nu-1)(\nu^3-5\nu^2+14\nu-16)}{32\nu^2 n^2} + \mathcal{O}(1/n^3).$$

- If $\nu = \mathcal{O}(n)$ (beginning of Phase 1), we can proceed as follows. Set $\nu = n/2^k, k=\mathcal{O}(1), \alpha := j/\nu$. We obtain

$$P_2(\nu,j) = \frac{\alpha[1-(1-\alpha)/2^{k+1}]}{1-\alpha(1-\alpha)/2^k}.$$

Note that $\alpha < 1$ as far as $j < \nu$. $P_2(\nu,\nu) = 1$, but the contribution to $\mathbb{E}(\nu^{(1)})$ in (24) is exponentially small. So we can expand $P_2(\nu,j)$ wrt α, which gives

$$\begin{aligned} k=0 &: \ P_2(\nu,j) = \alpha/2+\alpha^2+\alpha^3/2-\alpha^4/2-\alpha^5-\alpha^6/2+\alpha^7/2+\alpha^8+\dots, \\ k=1 &: \ P_2(\nu,j) = 3\alpha/4+5\alpha^2/8-\alpha^3/16-11\alpha^4/32+\dots \end{aligned}$$

According to (24), we must compute the mean of

$$2P_2(\nu,j)\alpha\nu,$$

with respect to the Binomial $\mathrm{BIN}(\nu,1/2)$. The moments of α^l are easily derived from the Generating Function

$$\mathbb{E}[e^{w\alpha}] = [(1+\exp(w/\nu))/2]^\nu,$$

i.e.

$$\begin{aligned} \mathbb{E}(\alpha) &= \frac{1}{2}, \\ \mathbb{E}(\alpha^2) &= \left(1+\frac{1}{\nu}\right)\Big/4, \\ \mathbb{E}(\alpha^3) &= \left(1+\frac{3}{\nu}\right)\Big/8, \\ \mathbb{E}(\alpha^4) &= (\nu^3+6\nu^2+3\nu-2)/(16\nu^3), \\ &\dots \end{aligned}$$

This leads to

$$\begin{aligned} \mathbb{E}(\nu^{(1)}) &= \left(1-\frac{1}{2^{k+1}}\right)(1+\nu)/2 + \left(\frac{1}{4^k}-\frac{3}{2^k}\right)(3+\nu)/8 \\ &+ (\nu^3+6\nu^2+3\nu-2)\left(\frac{1}{4^{k-1}}-\frac{1}{2^{k-1}}-\frac{1}{8^k}\right)\Big/(16\nu^2)+\dots. \end{aligned}$$

Of course, as k gets large, we recover $(\nu+1)/2$.

- More interestingly, fix k and let ν get large. We obtain $\mathbb{E}(\alpha^l) \sim 1/2^l$ and the mean of $2P_2(\nu, j)\alpha\nu$ is asymptotically given by $\nu/2$ which conforms to the fact that the distribution of $\nu^{(1)}$ is asymptotically the same in both algorithms.

The recurrence equation for P_4 is now

$$P_4(\nu, k, j) = \sum_{l=0}^{\nu} 2\binom{\nu}{l}\frac{1}{2^\nu}P_2(\nu, l)P_4(l, k-1, j).$$

Thanks to (21) and (22) and to the fact that the dominant contribution to c_1 is due to small values of $\nu^{(i)}$, the analysis of c_1 is similar to the one given in Sec. 3.2.1, 3.2.2, 3.2.3 (with a cost of 2 questions instead of 1 at each trial). This shows that the contribution of c_1 is again negligible compared with that of c_2.

4.2 Total cost

As soon as $\nu = 1$, we obtain

$$p(1,1) = 1 - \left(1 - \frac{1}{n}\right)\left(1 - \frac{0}{n}\right) = \frac{1}{n},$$

and the cost of c_2 is analyzed exactly as in Sec. 3.2.4 with 2 questions at each trial. The total cost is now given by the following result:

Proposition 4.1 *The total cost of Algorithm 1 is asymptotically given by*

$$c \sim \mathcal{N}(2n^2 \log_2 N, 4n^3 \log_2 N).$$

5 A generalization of the probabilistic Guessing Secrets

In this section we introduce a generalization of the probabilistic guessing secrets defined in Section 2. **The rules of the generalization**. Let Ω be the search space, let N be its cardinality, let $S \subseteq \Omega$ be the set of secrets, and let n be the cardinality of S. Then:

- As before, **A** may ask questions of the form: *Is the secret in X ?*, where X can be any subset of Ω.
- **B** chooses a set C of ρ secrets of S *at random* (with respect to the uniform distribution). If there exists at least $x \in C$ such that $x \in X$, then **B** answers "Yes", otherwise he answers "No".
- **A** wins the game if he correctly guesses all the secrets.

We propose an algorithm which is very similar to Algorithm 1. Its description is given in Algorithm 3.

We start by analyzing the variant of Algorithm 2. First of all, the probability $q(\nu)$ of answering NO is given by

$$q(\nu) = \frac{\binom{\nu}{0}\binom{n-\nu}{\rho}}{\binom{n}{\rho}}, p(\nu) = 1 - q(\nu). \tag{25}$$

For instance,

$$\begin{aligned}
p(1) &= \frac{\rho}{n}, \\
p(2) &= 1 - \left(1 - \frac{\rho}{n}\right)\left(1 - \frac{\rho}{n-1}\right), \\
p(4) &= 1 - \left(1 - \frac{\rho}{n}\right)\left(1 - \frac{\rho}{n-1}\right)\left(1 - \frac{\rho}{n-2}\right)\left(1 - \frac{\rho}{n-3}\right).
\end{aligned}$$

Algorithm 3

```
Input: the search secret space Ω
Output: the secret set S
X := Ω, Y := ∅;
repeat
  X := Ω − Y;
  repeat
    XL := ⌊X/2⌋; XR := X \ XL;
    repeat
      B picks a set C of ρ secrets in S uniformly at random;
      if there is at least one x ∈ C such that x ∈ XL then
        Answer:=Yes
      else
        B picks a set C of ρ secrets in S uniformly at random;
        if there is at least one x ∈ C such that x ∈ XR then
          Answer:=Yes
        end if
      end if
    until Answer:=Yes
    if x ∈ XL then
      X := XL
    else
      X := XR;
    end if
  until |X| = 1
  Y := Y ∪ X;
until |Y| = n
```

The law giving $j^{(i)}$ is the same as in Sec.3.2.1, (1) but the choice of ν must be carefully examined. We denote by JL the subset of secrets in XL (with $j = |JL|$), and by p_{JL} the probability that C intersects JL but not JR, and similarly, p_{JR} denotes the probability that C intersects JR but not JL. Also $p_{JL\cap JR}$ denotes the probability that C intersects both JL and JR. We have the obvious relations

$$\begin{aligned} q(j) &= q(\nu) + p_{JR}, \\ q(\nu - j) &= q(\nu) + p_{JL}, \\ p(\nu) &= p_{JL} + p_{JR} + p_{JL\cap JR}, \end{aligned}$$

which allows the computation of $p_{JL}, p_{JR}, p_{JL\cap JR}$.

Hence we obtain

$$P_2(\nu, j^{(0)}) := \Pr[\nu^{(1)} = j^{(0)}] = \frac{2p_{JL} + p_{JL\cap JR}}{2p(\nu)},$$

if we flip a coin to decide which of XL or XR is chosen when the answer is YES for both subsets. This leads to

$$P_2(\nu, j) = \left\{ 1 - \frac{\binom{n-\nu}{\rho}}{\binom{n}{\rho}} + \frac{\binom{n-(\nu-j)}{\rho}}{\binom{n}{\rho}} - \frac{\binom{n-j}{\rho}}{\binom{n}{\rho}} \right\} \Big/ (2p(\nu)). \tag{26}$$

We conjecture that, asymptotically, the distribution P_3 (see (23)) is again given by (3).

We will now consider successively the cases $\rho = o(n)$ and then the case $\rho = \mathcal{O}(n)$.

5.1 $\rho = o(n)$

We will see in the next section that, in this case, we can again use the rough approximation $\nu^{(i)} \sim \nu^{(i-1)}/2 \sim \nu/2^i$.

We must now carefully examine c_1:

- if $\nu^{(0)} \le n - \rho$ the analysis proceeds as in Sec. 3.2.1 and

$$\mathbb{E}(c_1) \sim \sum_{i=0}^{\log_2(\nu^{(0)})} \frac{1}{p(2^i)}. \tag{27}$$

- if $\nu^{(0)} > n - \rho$, then, asymptotically, $\nu^{(i)} > n - \rho$ for $i = 0 \ldots L_1^0 = \log_2(\nu^{(0)}) - \log_2(n - \rho)$ and $p(\nu^{(i)}) = 1$. So

$$\mathbb{E}(c_1) \sim \left[\log_2(\nu^{(0)}) - \log_2(n - \rho)\right] + \sum_{i=0}^{L_1^1} \frac{1}{p(2^i)}, \tag{28}$$

  with $L_1^1 \sim \log_2(n - \rho)$. Actually, as $\nu^{(0)} \le n$, then $\nu^{(1)} \le n/2$, and this case appears only once.

An asymptotic for c_1 is given in the next section.

As far as c_2 is concerned, the analysis proceeds exactly as in Sec.3.2.4, with now $p = p(1) = \rho/n$ instead of $1/n$.

5.1.1 Asymptotics of c_1

Let us first prove the following lemma

Lemma 5.1

$$\frac{\binom{n-\nu}{\rho}}{\binom{n}{\rho}} \sim 1 - \frac{\rho\nu}{n}, \ \textit{if } \rho = o(n), \nu = o(n).$$

Proof. Set $A := \frac{\binom{n-\nu}{\rho}}{\binom{n}{\rho}}$. We have

$$A = \left[1 - \frac{\rho}{n}\right]\left[1 - \frac{\rho}{n-1}\right]\ldots\left[1 - \frac{\rho}{n-(\nu-1)}\right].$$

This leads to

$$A \sim \left[1 - \frac{\rho}{n}\right]\left[1 - \frac{\rho}{n}\left(1 + \frac{1}{n}\right)\right]\ldots\left[1 - \frac{\rho}{n}\left(1 + \frac{\nu-1}{n}\right)\right].$$

Set $\alpha = 1 - \rho/n, \beta := \rho/n^2$. We obtain

$$A \sim \alpha\left[\alpha\left(1 - \frac{\beta}{\alpha}\right)\right]\left[\alpha\left(1 - 2\frac{\beta}{\alpha}\right)\right]\ldots\left[\alpha\left(1 - (\nu-1)\frac{\beta}{\alpha}\right)\right].$$

We have $\beta/\alpha = \mathcal{O}(\rho/n^2)$. We derive, asymptotically

$$\begin{aligned} A &\sim \alpha^{\nu} e^{-\beta/\alpha[1+2+\ldots+(\nu-1)]} \\ &\sim e^{-\rho\nu/n} e^{-\beta\nu(\nu-1)/(2\alpha)} && (29) \\ &\sim e^{-\rho\nu/n}, && (30) \end{aligned}$$

as $\rho(\nu/n)^2 = o(\rho\nu/n)$. This leads to

$$A \sim 1 - \frac{\rho\nu}{n}. \tag{31}$$

■

So

$$p(\nu) \sim \frac{\rho\nu}{n}, P_2(\nu, j) \sim \frac{j}{\nu}, \text{ if } \rho = o(n), \nu = o(n).$$

From (27), (28), we may assume $\nu = \mathcal{O}(1)$, (this corresponds to the most important contribution to c_1). So, if $\rho = o(n)$,

$$\sum_{i=0}^{\infty} \frac{1}{p(2^i)} \sim \frac{2n}{\rho}.$$

Hence,

$$\begin{aligned} &\text{If } \nu^0 \le n - \rho, \text{ then } \mathbb{E}(c_1) \sim \frac{2n}{\rho}, \\ &\text{If } \nu^0 > n - \rho, \text{ then } \mathbb{E}(c_1) \sim \log_2(\nu^{(0)}) - \log_2(n-\rho) + \frac{2n}{\rho}. && (32) \end{aligned}$$

Again, this sows that the contribution of c_1 is negligible compared with that of c_2.

With our usual approximations, the limiting distribution is again given by (19), when we set

$$\bar{c_1} := c_1\frac{\rho}{n}.$$

5.1.2 Total cost

The cost of c_2 is analyzed exactly as in Sec. 3.2.4 where we replace n by n/ρ. The total cost is now given by Proposition 3.2, with mean $n(n/\rho)\log_2 N$ and variance $n(n/\rho)^2 \log_2 N$.

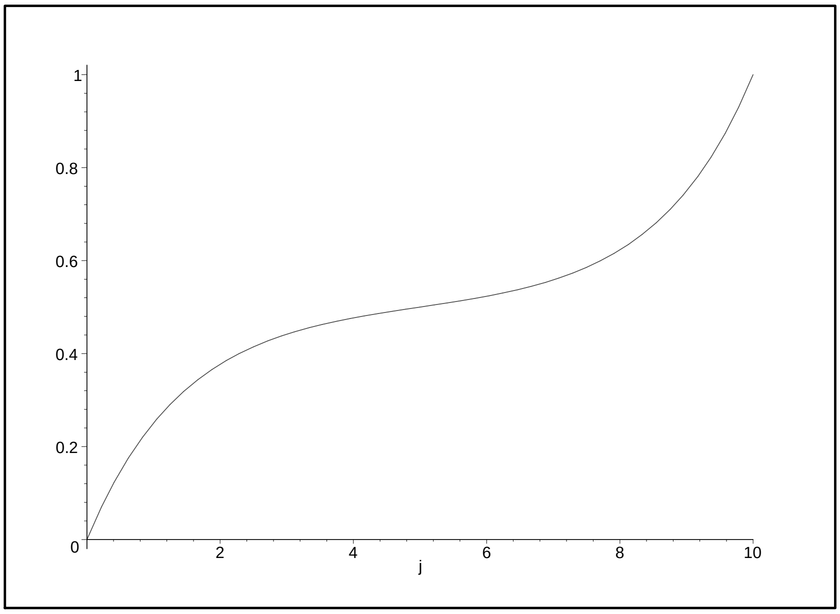

Figure 1: $P_2(\nu, j), \delta = 1/2, \nu = 10$

5.2 $\rho = \mathcal{O}(n)$

Now we obtain the following lemma

Lemma 5.2

$$\frac{\binom{n-\nu}{\rho}}{\binom{n}{\rho}} \sim \alpha^\nu, \ \textit{if } \rho = \mathcal{O}(n), \nu^2 = o(n).$$

Proof. With the same notations as in Lemma 5.1, we derive, with

$$\rho = \delta n, \alpha = 1 - \delta, \beta = \frac{\delta}{n}, \frac{\beta}{\alpha} = \frac{\delta}{1-\delta}\frac{1}{n},$$

$$A \sim \alpha^\nu e^{-\beta\nu(1-\nu)/(2\alpha)} \sim \alpha^\nu, \ \text{if } \nu^2 = o(n).$$

■

Again the restriction on ν is not important, as the dominant contribution to c_1 is related to $\nu = \mathcal{O}(1)$.
We also obtain

$$p(\nu) \sim 1 - \alpha^\nu, \tag{33}$$

$$P_2(\nu, j) \sim \left[1 - \alpha^\nu + \alpha^{\nu-j} - \alpha^j\right]/[2(1-\alpha^\nu]. \tag{34}$$

Note that $P_2(\nu, j)$ is no more asymptotically linear (as was the case in all previous analysis, see (26)). A plot of $P_2(\nu, j)$ for $\delta = 1/2, \nu = 10$ is given in Figure 1.
We derive

$$\sum_j P_4(\nu, 1, j) = \sum_j 2\binom{\nu}{j}\frac{1}{2^\nu}P_2(\nu, j) = 1,$$

$$\mathbb{E}(\nu^{(1)}) = \sum_0^\nu 2\binom{\nu}{j}\frac{1}{2^\nu}P_2(\nu, j)j = \frac{\nu}{2} + \frac{\nu(1-\delta/2)^{\nu-1}\delta}{2\left[1-(1-\delta)^\nu\right]}.$$

Actually, $\mathbb{E}(\nu^{(1)})$ differs rather little from $\nu/2$. Figure 2 shows, for $\delta = 1/2$, $\mathbb{E}(\nu^{(1)})$ with the exact form of $P_2(\nu, j)$, i.e.(26), the asymptotics (34) and $\nu/2$.

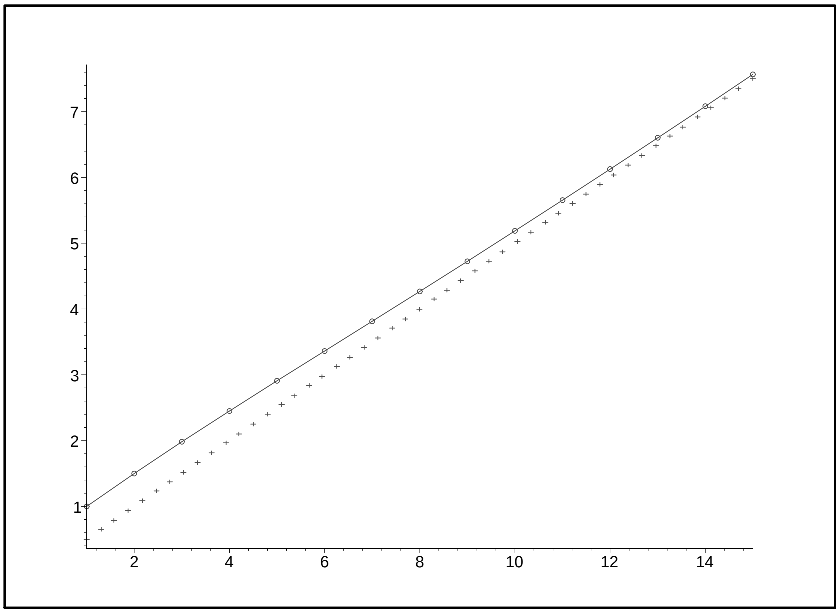

Figure 2: $\mathbb{E}(\nu^{(1)}), \delta = 1/2$ with the exact form of $P_2(\nu, j)$ (line), the asymptotics (circle) and $\nu/2$ (cross).

Next we compute

$$c_{1,1}(\delta) \sim \sum_{i=0}^{\infty} \frac{1}{p(2^i)} \sim \sum_{i=0}^{\infty} \frac{1}{1-\alpha^{2^i}} = \mathcal{O}(1).$$

Hence,

$$\begin{aligned} &\text{If } \nu^{(0)} \leq n-\rho, \text{ then } \mathbb{E}(c_1) \sim c_{1,1}(\delta), \\ &\text{If } \nu^{(0)} > n-\rho, \text{ then } \mathbb{E}(c_1) \sim \log_2(\nu^{(0)}) - \log_2(n-\rho) + c_{1,1}(\delta) = \mathcal{O}(\log_2(n)). \end{aligned}$$

Again, this will show that the contribution of c_1 is negligible compared with that of c_2. Of course, c_1 does not have the asymptotic distribution (19) anymore, but corresponds to $\sum_i \mathrm{GEOM}(p(2^i))$, with $p(\nu)$ given by (33). A plot of $c_{1,1}(\delta)$ is given in Figure 3.

c_2 is now given by a sum of L_2 $\mathrm{GEOM}(p = \rho/n = \delta)$ RV. The total cost is now given by Proposition 3.2, with mean $n(1/\delta)\log_2 N$ and variance $n(1-\delta)/\delta^2 \log_2 N$.

6 The variant of Algorithm 1 (i.e. Algorithm 3)

The success probability is now given by

$$p(\nu, j) = 1 - \frac{\binom{n-j}{\rho}}{\binom{n}{\rho}} \frac{\binom{n-(\nu-j)}{\rho}}{\binom{n}{\rho}}.$$

The probability of choosing the left subset is now given by

$$\begin{aligned} P_2(\nu, j) &:= \Pr(\nu^1 = j) \\ &= \left\{ \frac{1}{2}\left[1 - \frac{\binom{n-j}{\rho}}{\binom{n}{\rho}}\right] + \frac{1}{2}\left[\frac{\binom{n-(\nu-j)}{\rho}}{\binom{n}{\rho}}\left[1 - \frac{\binom{n-j}{\rho}}{\binom{n}{\rho}}\right]\right]\right\} \Big/ p(\nu, j) \\ &= \left\{ 1 - \frac{\binom{n-j}{\rho}}{\binom{n}{\rho}} + \frac{\binom{n-(\nu-j)}{\rho}}{\binom{n}{\rho}} - \frac{\binom{n-j}{\rho}}{\binom{n}{\rho}} \frac{\binom{n-(\nu-j)}{\rho}}{\binom{n}{\rho}} \right\} \Big/ (2p(\nu, j)). \end{aligned}$$

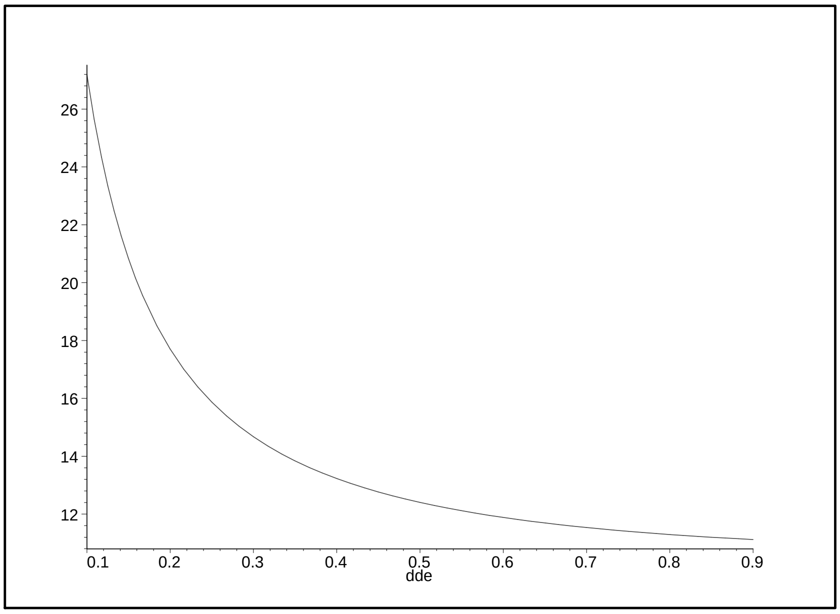

Figure 3: $c_{1,1}(\delta)$

6.1 $\rho = o(n)$

Again, we conjecture that the limiting distribution of $P_3(\nu, k)$ is the extreme-value distribution.

Assuming again, as a rough approximation, that $j \sim \nu/2$, we compute

$$\begin{aligned} p(1,1) &= \frac{\rho}{n}, \\ p(2,1) &= 1 - \left(1 - \frac{\rho}{n}\right)^2 \sim 1 - e^{-2\rho/n}, \\ p(4,2) &= 1 - \left[\left(1 - \frac{\rho}{n}\right)\left(1 - \frac{\rho}{n-1}\right)\right]^2 \sim 1 - \left(1 - \frac{\rho}{n}\right)^4 \sim 1 - e^{-4\rho/n}, \end{aligned}$$

and, more generally,

$$p(\nu, \nu/2) \sim 1 - e^{-\nu\rho/n},$$

which is exactly (30). The analysis proceeds now as in Sec. 5.1.1.

Note that, if $\rho = o(n), \nu = o(n)$, then $p(\nu, j) \sim \rho\nu/n, P_2(\nu, j) \sim j/\nu$. Again, this will show that the contribution of c_1 is negligible compared with that of c_2. We conjecture that the distribution of c_1 can be obtained as in Sec.5.1 (with a cost of 2 questions instead of 1 at each trial). The cost of c_2 is analyzed exactly as in Sec. 5.1.2 with 2 questions. We obtain the following result

Proposition 6.1 *The total cost of Algorithm 3 is asymptotically given, if $\rho = o(n)$, by*

$$c \sim \mathcal{N}\left(2n\frac{n}{\rho}\log_2 N, 4n\left(\frac{n}{\rho}\right)^2 \log_2 N\right).$$

6.2 $\rho = \mathcal{O}(n)$

Now we obtain the following asymptotics

Proposition 6.2 *The total cost of Algorithm 3 is asymptotically given, if $\delta = \rho/n$, by*

$$c \sim \mathcal{N}\left(2n\frac{1}{\delta}\log_2 N, 4n\frac{1-\delta}{\delta^2}\log_2 N\right).$$

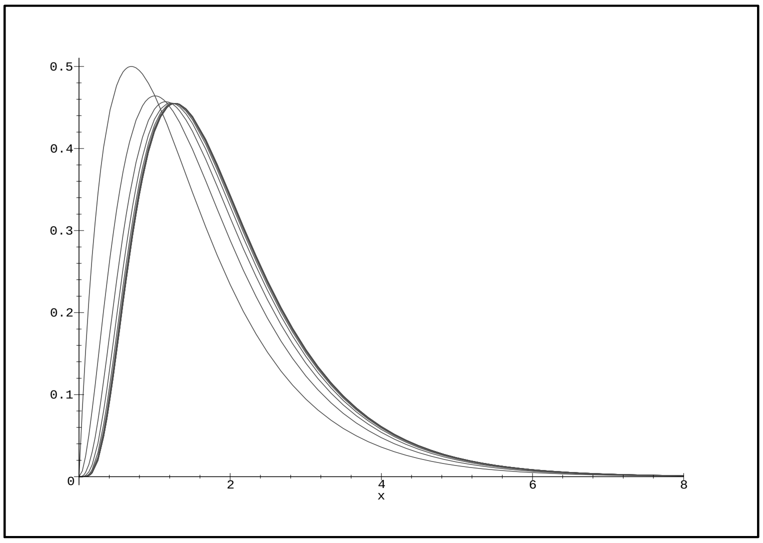

Figure 4: $f_i(x), i = 1..12$ and $f(x)$

7 Numerical checks and simulations

In this section, we first give a comparison of exact and asymptotic expressions, next we present a simulation of c_1 distribution, and finally we provide two simulations of c_2 distribution, with different parameters.

7.1 Comparison of exact and asymptotic expressions

Let us first check the precision of the limiting density (18) and of the limiting distribution function (19). Actually, 7 or 8 terms are enough to get 12 digits of precision in (18) and (19). Moreover, the convergence of f_i and F_i (given by (7) and (8)) is quite fast: f_{12} (or F_{12}) are almost indistinguishable from $f(x)$ (or $F(x)$). Figures 4 and 5 show the first graphs of f_i , $F_i, i = 1..12$ and the limiting graphs. Next, for the variant case, we have compared the exact expression $1/p(\nu)$, as given by (25), with the asymptotics (29), (30) and (31). This is given in Figures 6, 7, for $\rho = 10, n = 1000$.

A numerical check of P_3, i.e. (23), compared with (3), for Alg.2, Alg.1 and Alg.3($\rho = 10$) is given in Figures 8, 9 and 10, with $n = 100, \nu = 100$. The fit is nearly perfect.

7.2 Simulation of c_1 distribution

We have made two simulations, S_1, S_2 of Algorithm 1, of $m = 500$ trials each, with $\tau = 0, N = 300000, n = 3000, (S_1), n = 16(S_2)$ which has led to the following Table 1, where the ˜ sign is related to the observed quantities. We have $\log_2(N) = 18.19\ldots$.

$\mathbb{E}(L_1)$ is asymptotically given by (4), $\mathbb{E}(c_1) \sim (1 + 1/L)n, \mathbb{E}(c_2) \sim [\log_2(N) - \log_2(n)]n$. Obviously, N and n are not large enough to allow a good fit to the mean asymptotics. However, the observed distributions functions are more interesting. For S_1, we have compared the observed distribution function of L_1 with the asymptotic one given by (3), with $\eta := k - \log_2^{(0)}(\nu)$, where we use the estimator: $\log_2^{(0)}(\nu) := \widetilde{\mathbb{E}}(L_1) - \gamma/\ln(2) - 1/2$. This is given in Figure 11. The fit is rather good. Next, we have compared the observed distribution function of c_1 with the asymptotic one given by $F(\delta x/n)$, where F is given by (19) and $\delta = 2n/\widetilde{\mathbb{E}}(c_1)$, in order to fit the mean to the observed value. This is shown in Figure 12. Again the fit is reasonable. This is quite surprising, taking into account the rough approximation: $\nu^{(i)} \sim \nu/2^i$.

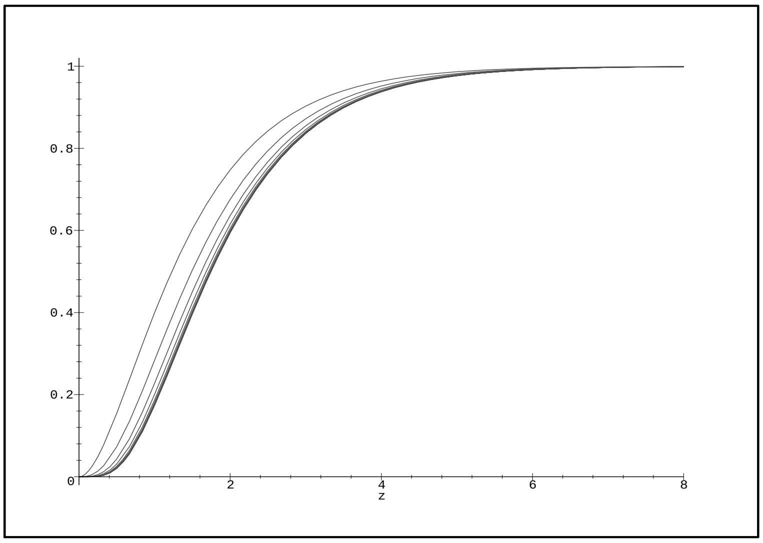

Figure 5: $F_i(x), i = 1..12$ and $F(x)$

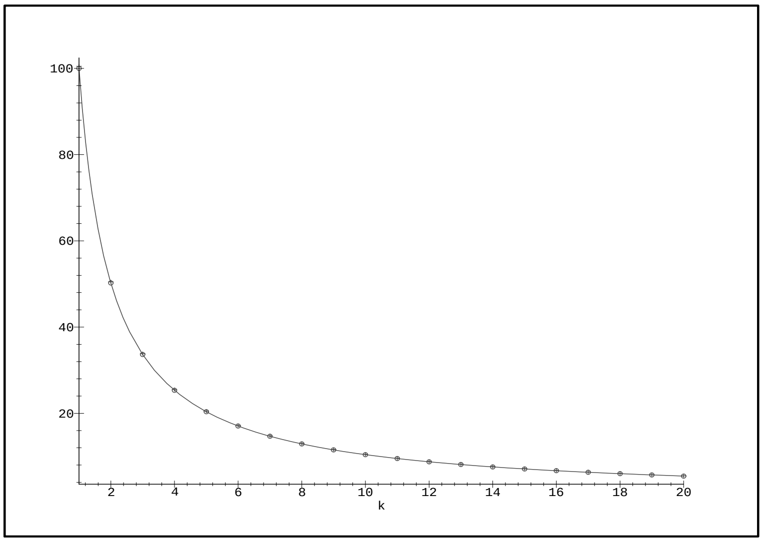

Figure 6: $1/p(\nu)$, as given by (25)(line), (29)(circle) and (30)(cross), $\rho = 10, n = 1000$. Crosses and circles are undistinguishable.

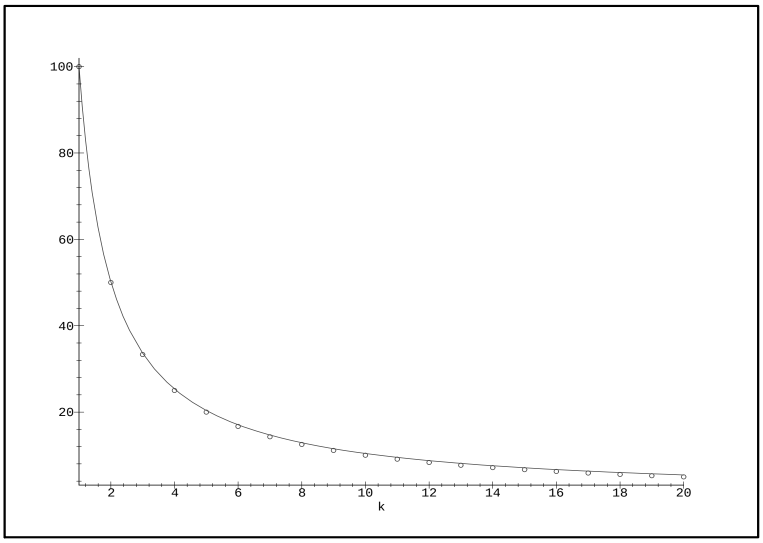

Figure 7: $1/p(\nu)$, as given by (25)(line), (31)(circle), $\rho = 10, n = 1000$.

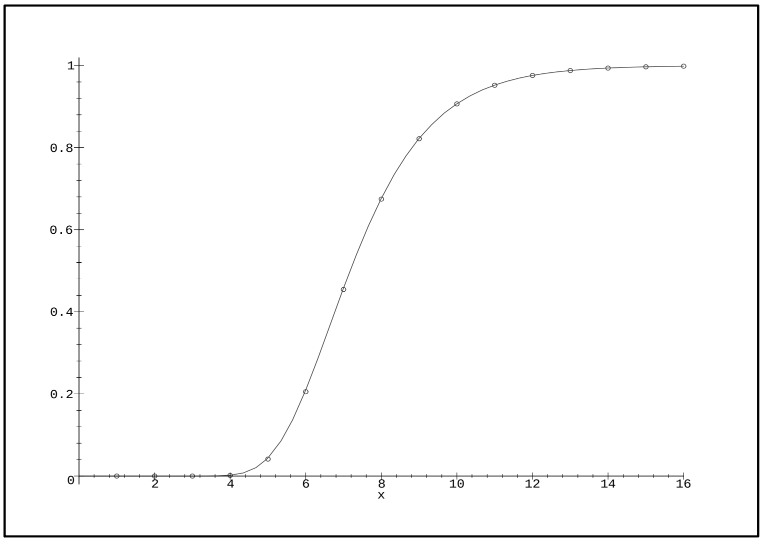

Figure 8: $P_3(100, k)$ (circle) compared with (3) (line), Alg.2.

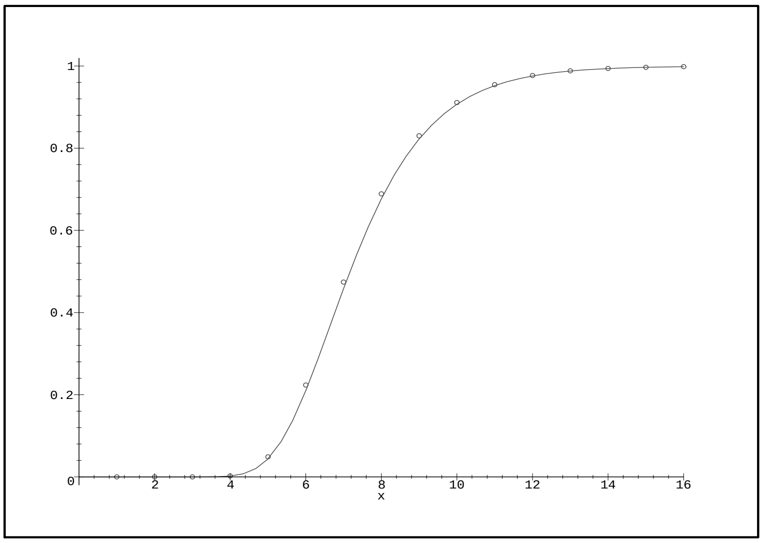

Figure 9: $P_3(100, k)$ (circle) compared with (3) (line), Alg.1.

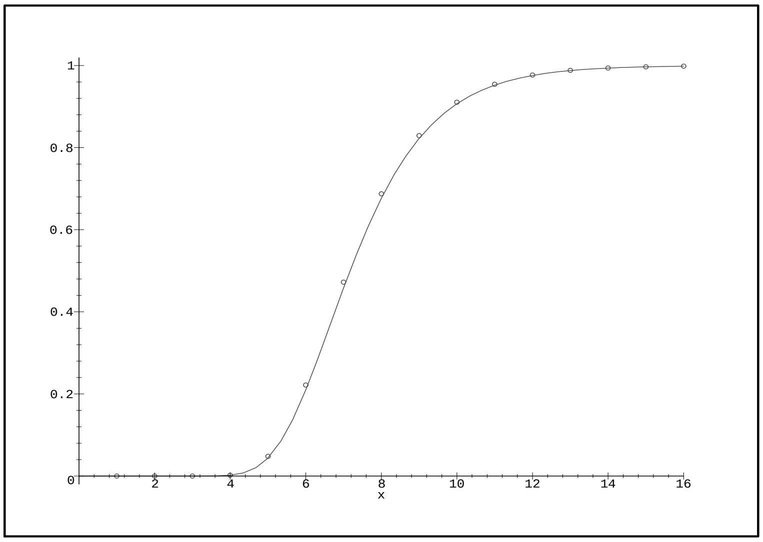

Figure 10: $P_3(100, k)$ (circle) compared with (3) (line), Alg.3, $\rho = 10$.

	n	$\log_2(n)$	$\log_2(N) - \log_2(n)$	$\widetilde{\mathbb{E}}(L_1)$ $\mathbb{E}(L_1)$	$\widetilde{\mathbb{V}}(L_1)$	$\widetilde{\mathbb{E}}(c_1)$ $\mathbb{E}(c_1)$	$\widetilde{\mathbb{V}}(c_1)$	$\widetilde{\mathbb{E}}(c_2)$ $\mathbb{E}(c_2)$	$\widetilde{\mathbb{V}}(c_2)$
S_1	3000	11.55	6.64	13.87 12.88	2.99	7440 7328	$.19\ 10^8$	13345 19931	$.72\ 10^8$
S_2	16	4	14.19	6.21 5.33	2.93	36.92 39.08	447.26	191 227	3551

Table 1: Simulations S_1, S_2

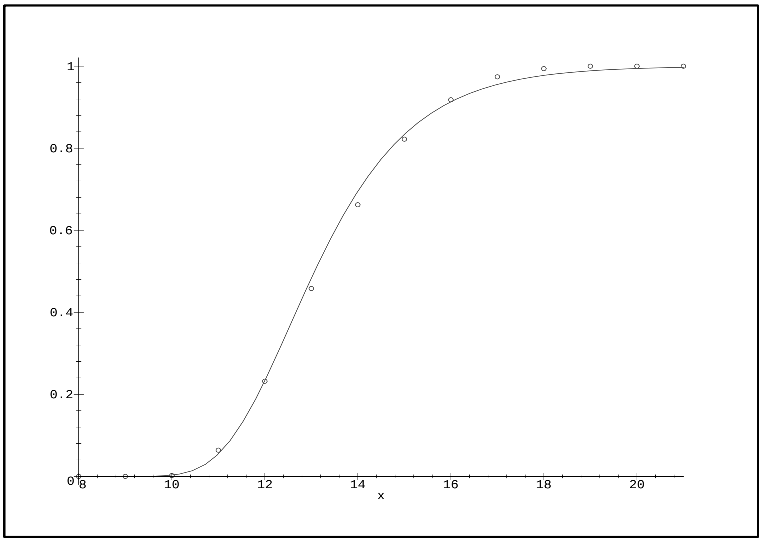

Figure 11: S_1: Distribution of L_1, asymptotic=line, observed=circle

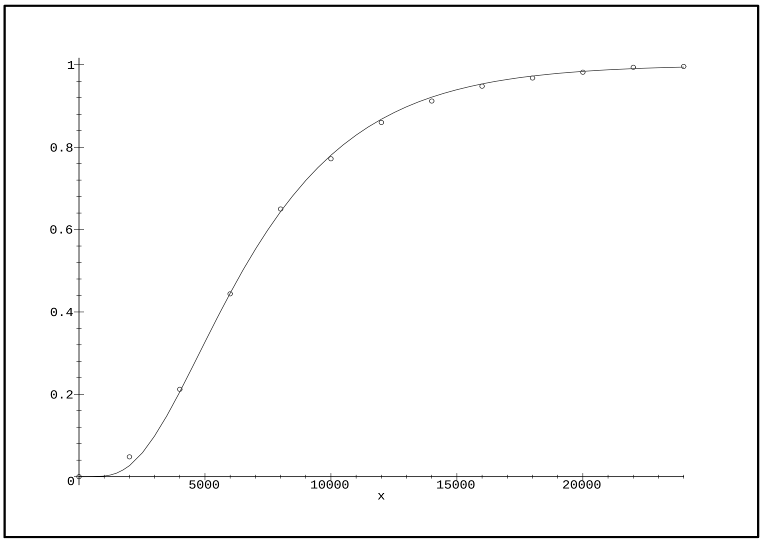

Figure 12: S_1: Distribution of c_1, asymptotic=line, observed=circle

7.3 First simulation of c_2 distribution

We have first compared the observed distribution function of c_2 with the asymptotic Gaussian one with mean and variance $\widetilde{\mathbb{E}}(c_2), \widetilde{\mathbb{V}}(c_2)$. This is given in Figure 13. The fit is bad. Similarly, we provide in Figure 14 the comparison between the observed distribution function of c_2 and the gamma distribution given by (20), with the estimator: $L_2 = 4$ (we are guided by $\widetilde{\mathbb{E}}(c_2)/n = 4.44$). Again, the fit is bad. The reason is the following: there is an uncertainty on the mean of L_1, which entails a random contribution on $L_2 \sim \log_2(N) - L_1$.

7.4 Second simulation of c_2 distribution

Now, we have decreased n to 16 in simulation S_2, keeping the same value for N, in order to stabilize L_2. The fits similar to Figures 11 and 12 are given in Figures 15 and 16. Now the fit of c_2 with the Gaussian and gamma distributions, with the estimator: $L_2 = 12$ (now, we are guided by $\widetilde{\mathbb{E}}(c_2)/n = 11.94$), given in Figures 17 and 18, is much better.

8 Conclusion

There are some questions about the probabilistic guessing secrets that remain open. For example, we do not know if our strategy is optimal. So,

- Is there any other search strategy that guesses all secrets of S with probability one and which provides an expected number of questions smaller than $2n^2 \log_2 N$?

- Is the expected time complexity of the algorithm for this strategy smaller than $\mathcal{O}(n^2 \log N)$?

Moreover, although the expected number of questions providing a set X that contains exactly one secret (i.e., c_1) has a negligible contribution compared with that of the expected number of questions providing a set X of size 1 (i.e., c_2), the determination of the distribution of c_1 seems intractable (even in asymptotic form). We made the following conjecture: the distribution of c_1 is asymptotically given by Proposition 3.1 and we gave experimental and probabilistic evidence to support it, but we do not have an analytic proof.

Acknowledgements

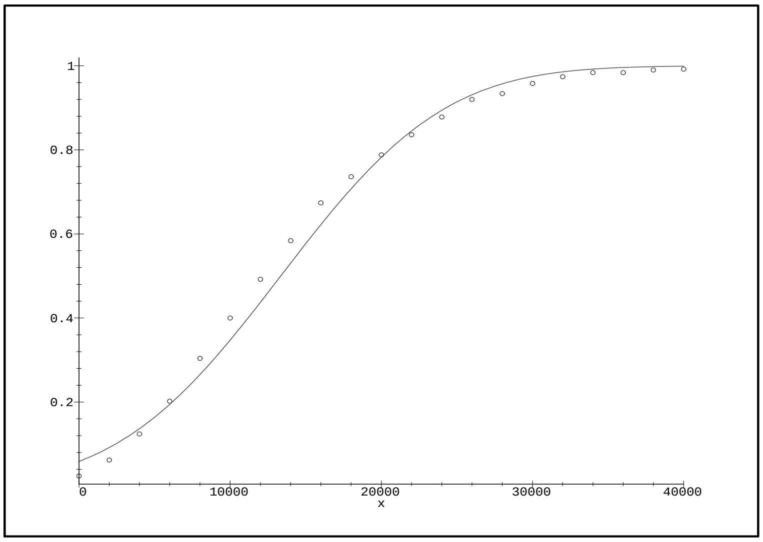

Figure 13: S_1: Distribution of c_2, asymptotic Gaussian=line, observed=circle

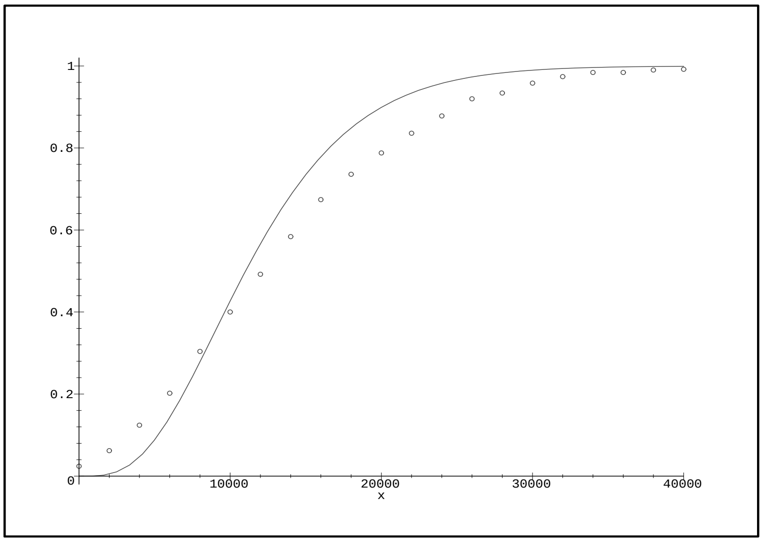

Figure 14: S_1: Distribution of c_2, asymptotic gamma=line, observed=circle

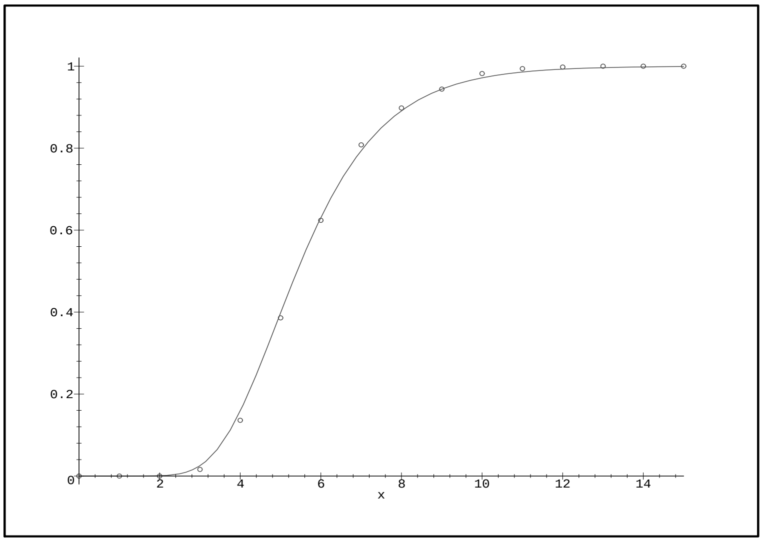

Figure 15: S_2: Distribution of L_1, asymptotic=line, observed=circle

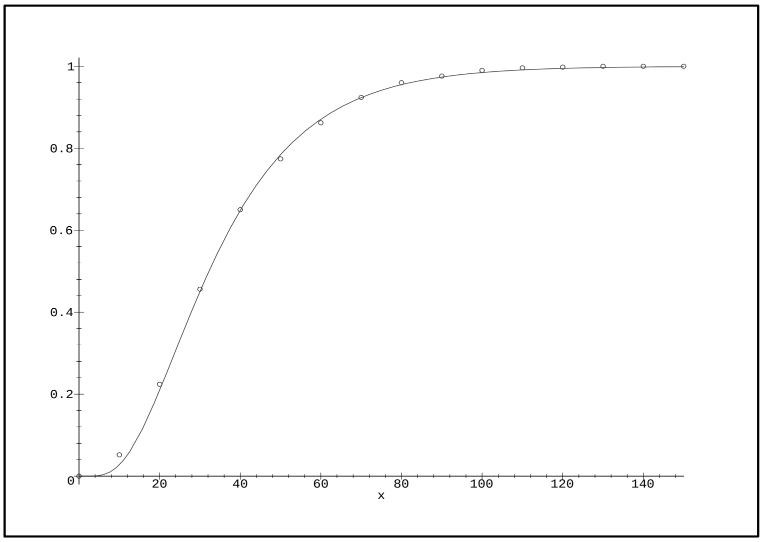

Figure 16: S_2: Distribution of c_1, asymptotic=line, observed=circle

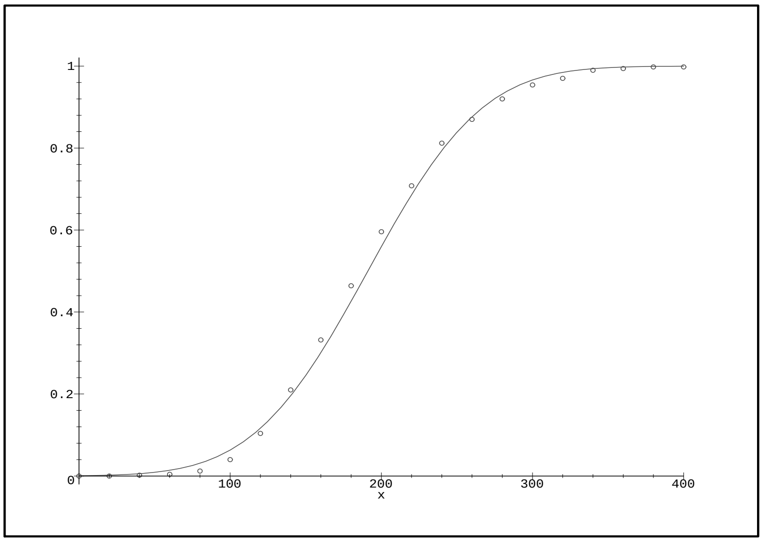

Figure 17: S_2: Distribution of c_2, asymptotic Gaussian=line, observed=circle

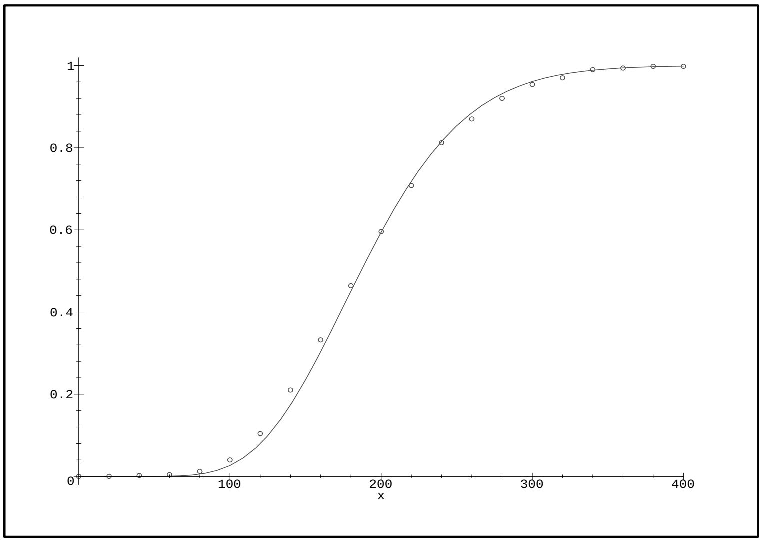

Figure 18: S_2: Distribution of c_2, asymptotic gamma=line, observed=circle

The second author wishes to thank the Scienze Matematiche e Informatiche "Roberto Magari" for its warm hospitality. In particular, the cooperation with Alberto was quite agreeable and fruitful. He will be deeply regretted by his numerous friends.

The pertinent comments and suggestions of the referees led to substantial improvements in our analysis and presentation.

References

[1] N. Alon, V. Guruswam, T. Kaufman, and M. Sudan. Guessing secrets efficiently via list decoding. In *Proceedings of the 13th Annual ACM-SIAM Symposium On Discrete Mathematics (SODA-02)*, pages 254–262, 2002.

[2] F. Chung, R. Graham, and F. T. Leighton. Guessing secrets. *Electronic Journal of Combinatorics*, 8, 2001. R13.

[3] F. Chung, R. Graham, and L. Lu. Guessing secrets with inner product questions. In *Proceedings of the 13th Annual ACM-SIAM Symposium On Discrete Mathematics (SODA-02)*, pages 247–253, 2002.

[4] D. Z. Du and F. K. Hwang. *Combinatorial group testing and its applications.* World Scientific, 1993.

[5] P. Flajolet, X. Gourdon, and P. Dumas. Mellin transforms and asymptotics: Harmonic sums. *Theoretical Computer Science*, 144:3–58, 1995.

[6] N.L. Johnson and S. Kotz. *Urns Models and Their Application.* J. Wiley, 1977.

[7] D. E. Knuth. *The Art of Computer Programming*, volume 3: Sorting and Searching. Addison-Wesley, 1973. Second edition, 1998.

[8] G. Louchard. Brownian motion and algorithm complexity. *BIT*, 26:17–34, 1986.

[9] G. Louchard. Exact and asymptotic distributions in digital and binary search trees. *RAIRO Theoretical Informatics and Applications*, 21(4):479–496, 1987.

[10] D. Micciancio and N. Segerlind. Guessing two secrets with small queries. Technical report, 2002.

[11] H. Q. Ngo and D. Zhu. A survey on combinatorial group testing algorithms with applications to DNA library screening. In *Discrete Mathematical Problems with medical applications*, pages 171–182. DIMACS, 1999.

[12] I. Peterson. Guessing secrets. *Science News*, 161(14):214–216, 2002.

[13] W. Szpankowski. *Average case Analysis of Algorithms on sequences.* Wiley, 2001.