Two Observations on Probabilistic Primality Testing

center tab(:); c c c c c s s s. Pierre Beauchemin:Gilles Brassard [1]:Claude Crépeau [2, 3]:Claude Goutier

Université de Montréal C.P. 6128, Succ "A" Montréal(Québec) Canada H3C 3J7

ABSTRACT

In this note, we make two loosely related observations on Rabin's probabilistic primality test. The first remark gives a rather strange and provocative reason as to *why is Rabin's test so good*. It turns out that a single iteration fails with a non-negligible probability on a composite number of the form $4j+3$ *only if* this number happens to be easy to split. The second observation is much more fundamental because is it *not* restricted to primality testing: it has profound consequences for the entire field of probabilistic algorithms. There we ask the question: *how good is Rabin's algorithm?* Whenever one wishes to produce a uniformly distributed random probabilistic prime with a given bound on the error probability, it turns out that the *size* of the desired prime must be taken into account.

1. Introduction

In this note, we make two loosely related observations on Rabin's probabilistic primality test. The first remark gives a rather strange and provocative reason as to *why is Rabin's test so good*. It turns out that a single iteration fails with a non-negligible probability on a composite number of the form $4j+3$ *only if* this number happens to be easy to split. The second observation is much more fundamental because is it *not* restricted to primality testing: it has profound consequences for the entire field of probabilistic algorithms. There we ask the question: *how good is Rabin's algorithm?* Whenever one wishes to produce a uniformly distributed random probabilistic prime with a given bound on the error probability, it turns out that the *size* of the desired prime must be taken into account.

2. A Brief Survey of Primality Testing

How difficult is it to distinguish prime numbers from composite numbers? This is perhaps the single most important problem in computational number theory. We do not attempt here an exhaustive review of its long history. Let us only mention some of the most outstanding modern steps. It has been known for several years that the problem of recognizing prime numbers belongs to **P** under the Extended Riemann's Hypothesis [Mi] and that it belongs to Co-**RP** [R1, SS] and **NP** [P] without any assumptions. It can also be solved in *almost* polynomial time by a deterministic algorithm that runs for a number of steps in $O(m^{O(\log\log m)})$, where m is the size of the number to be tested [APR]. More recently, it was found to lie in **RP** [GK, AH], and therefore in **ZPP** [G] as well. In other words, this problem can be solved in probabilistic polynomial time by a Las Vegas [B] algorithm: whenever an answer is obtained, that answer is correct.

---

1. Supported in part by NSERC grant A4107

2. Supported in part by an NSERC postgraduate scholarship

3. Current address: MIT

From a theoretical point of view, the problem of primality testing is therefore solved (although it remains of interest to figure out whether or not it belongs to **P** without assumptions). However, the polynomial that gives the running time of [GK] is of the twelfth degree and [AH] does not improve on this, which makes these algorithms of little practical use. For very large numbers (several hundreds of decimal digits), this leaves us with Rabin's probabilistic test [R1] as the best approach. Let prob[$Rabin(n) = verdict$] denote the probability that one iteration of this algorithm on input n returns *verdict*, where *verdict* can either be ''*prime*'' or ''*composite*''. The basic theorem about Rabin's test is that

$$\text{prob}[Rabin(n) = \text{``}prime\text{''} \mid n \text{ is indeed prime}] = 1$$

whereas

$$\text{prob}[Rabin(n) = \text{``}prime\text{''} \mid n \text{ is in fact composite}] \le 1/4 .$$

One is therefore certain that n is composite whenever any single run of $Rabin(n)$ returns ''*composite*''. On the other hand, one can never be sure that n is a prime no matter how many runs of $Rabin(n)$ have returned ''*prime*''. This test is usually run in a loop as follows:

```
function RepeatRabin(n, k)
  { n is an odd integer to be tested for primality;
    k is a safety parameter discussed below }
  var i : integer;  done : Boolean
  i ← 0
  repeat
    i ← i + 1
    done ← (Rabin(n) = ''composite'')
  until done  or  i ≥ k
  if done  then return ''composite'' { for sure }
           else return ''prime'' { probably (?) } .
```

There is a trade-off in the choice of the parameter k above: the bigger it is, the more confident we are in the advent of a ''*prime*'' answer but the more time it takes to build up this confidence. This paper addresses two aspects of the question: *just how confident in a number's primality can we be after running this test?* .

3. Why is Rabin's Test so Good?

This section only applies when n is of the form $4j+3$. In this case, Rabin's test (which is then equivalent to Solovay-Strassen's [SS]) becomes quite simple. Let

$$\mathbb{Z}_n^* = \{ x \mid 1 \le x < n \textbf{ and } \gcd(x, n) = 1 \}$$

and $$\mathrm{R}_n = \{ a \in \mathbb{Z}_n^* \mid a^{(n-1)/2} \equiv \pm 1 \pmod{n} \}.$$

The basic theorem states that $\mathrm{R}_n = \mathbb{Z}_n^*$ whenever n is a prime, whereas $\#\mathrm{R}_n \le \#\mathbb{Z}_n^*/4$ otherwise, still assuming that $n \equiv 3 \pmod 4$. Notice that both 1 and $n-1$ always belong to R_n. This theorem is used naturally as follows:

```
function Rabin(n)
  { we assume that n is of the form 4j+3 }
  a ← random integer uniformly selected in 2..n−2
  if a ∈ Rn then return ''prime''
            else return ''composite'' .
```

Whenever n is composite, the error probability of this procedure is clearly given by $(\#\mathrm{R}_n - 2)/(n-3)$, so that elements of R_n others than 1 and $n-1$ are known as *false witnesses* for n. From the basic theorem, we know that this error probability is always smaller than 25%. However, it is well known to be often *much* smaller. Monier gives an exact (but rather scaring) formula for this probability [Mo]; see also [Kr]. As a corollary of Monier's formula, the error probability never exceeds $(\phi(n)/2^{r-1}-2)/(n-3)$, where r is the number of distinct prime factors of n and $\phi(n) = \# \mathbb{Z}_n^*$ denotes Euler's function [HW]. Despite this tightening of the bound on the error probability (at least when n has more than three distinct prime factors), it turns out that the latter is usually *still* much smaller. In other words, Rabin's test performs in practice much better than one might naively expect.

For instance, $42{,}799\ (=127\times337)$ admits only 880 false witnesses, compared to $\phi(42{,}799)/4 = 10{,}584$. Even better, Rabin's test *never* fails on integers of the form $3\times5\times7\times11\times\cdots$ such as 15,015: these admit no false witnesses at all. More impressively, it is enough to test deterministically for each $a \in \{2,5,7,13\}$ in order to decide primality without *any* failures up to 25×10^9 (using $\{2,3,5,7\}$ still leaves *one* error in this range [PSW]). Although "high risk" numbers exist, such as $n = 79{,}003\ (=199\times397)$ or $3{,}215{,}031{,}751\ (=151\times751\times28{,}351)$ with $\#R_n = \#\mathbb{Z}_n^*/4$ and $(\#R_n-2)/(n-3) \approx 24.8\%$, these are not the rule (one can nonetheless prove, using Monier's formula, that every composite number of the form $(12m+7)(24m+13)$ is such a high risk number, provided both $12m+7$ and $24m+13$ are prime). We address here the following bizarre question: *why* is Rabin's test so good?

In order to give some sort of answer, we define the following set:

$$H_n = \{\, b \in \mathbb{Z}_n^* \mid b \notin R_n \text{ and } (\exists a \in R_n)[a^2 \equiv b^2 \pmod{n}] \,\}.$$

Assume for the moment that n is not a prime power (i.e. not of the form p^m for some prime p and some integer $m \geq 2$). Theorem 1 states that each element of H_n is a *handle* that allows easy splitting of n (i.e. finding at least one non trivial factor of n) and that there are at least as many such handles as there are false witnesses. Our provocative interpretation states that it is only possible for *a single iteration* of Rabin's test to fail (i.e. declare n prime) with a non-negligible probability if it happens that n is easy to split (and hence obviously composite)! This result extends to every composite n of the form $4j+3$ in an obvious way since prime powers are easy to split. In other words, there exists a simple probabilistic splitting algorithm whose running time is small on every composite number congruent to 3 modulo 4 on which Rabin's test is not extremely effective. More precisely, for any polynomial p, the splitting algorithm succeeds at finding a non trivial factor in expected polynomial time on all those composite integers n such that prob[$Rabin(n)$ = ''*prime*''] $\geq 1/p(|n|)$ and $n \equiv 3 \pmod 4$, where $|n|$ denotes the size of n (in bits or in decimal digits).

What happens when n is of the form $4j+1$? We leave this as an open question. Let us only point out that Rabin's test *could* actually work better on numbers congruent to 3 modulo 4 than on numbers congruent to 1 modulo 4. Indeed, among the 4842 odd composite integers smaller than 25×10^9 that count 2 as a false witness, only 1033 ($\approx$21%) are of the form $4j+3$ [PSW].

Theorem 1

(i) $(\forall b \in H_n)[\gcd(n,\ 1 + b^{(n-1)/2} \bmod n)$ is a non trivial divisor of $n]$;

(ii) $\#H_n \geq \#R_n$.

Proof

Except for the exponents, all calculations in this proof are done modulo n.

(i) Consider any $b \in \mathrm{H}_n$. Let $a \in \mathrm{R}_n$ be such that $a^2 = b^2$. Let $x = b^{(n-1)/2}$. We know that $x \neq \pm 1$ because $b \not\in \mathrm{R}_n$. On the other hand, $x^2 = b^{n-1} = (b^2)^{(n-1)/2} = (a^2)^{(n-1)/2} = (a^{(n-1)/2})^2 = (\pm 1)^2 = 1$. Therefore, x is a non trivial square root of 1, and this is enough to split n by the well-known formula $\gcd(n, 1+x)$ [R2].

(ii) For each $a \in \mathrm{R}_n$, define $\mathrm{B}_n(a) = \{ b \in \mathbb{Z}_n^* \mid a^2 \equiv b^2 \pmod{n} \}$. Because n is composite and is not a prime power, $\mathrm{B}_n(a)$ contains at least 4 elements [HW]. Consider $b \in \mathrm{B}_n(a)$ such that $b \neq \pm a$. We have

$$\begin{aligned} b^{(n+1)/2} &= (b^2)^{(n+1)/4} && (\text{because } n \equiv 3 \pmod{4}) \\ &= (a^2)^{(n+1)/4} = a^{(n+1)/2} \\ &= a \times a^{(n-1)/2} = \pm a && (\text{because } a \in \mathrm{R}_n). \end{aligned}$$

Therefore, $b^{(n-1)/2} = b^{(n+1)/2}/b = \pm a/b \neq \pm 1$ because $b \neq \pm a$, hence $b \in \mathrm{H}_n$. This shows that to each pair $a, -a$ of elements of R_n corresponds at least two distinct elements in H_n, completing the proof that H_n contains at least as many elements as R_n. Notice also that this reasoning is trivially extended to conclude, as mentioned earlier in this section, that $\#\mathrm{R}_n \leq \phi(n)/2^{r-1}$, where r is the number of distinct prime factors of n, because each quadratic residue admits in this case exactly 2^r distinct square roots [HW]. □

Notice that part (i) of this proof still holds when $n \equiv 1 \pmod 4$. Unfortunately, part (ii) fails miserably because R_n is then always empty, due to the fact that each square root of the square of a false witness is also a false witness. This fact may partly explain the observed phenomenon that Rabin's test seems to be less effective on these numbers.

4. How Good is Rabin's Test?

We must first ask the following question: *what* is Rabin's test good for?
At least two answers come to mind: to *decide* on the primality of a given integer and to *generate* one or several primes (perhaps of a given size). We shall consider these two settings in turns, starting with the second.

4.1. How to Generate Random Primes of a Given Size

The generation of large primes drawn with a uniform distribution from the set of all primes of a given size is of crucial importance in cryptography [RSA]. Although it is possible to generate such primes with certainty using the algorithms of [APR, AH], their running time is currently too high to be used in practice. It is also possible to efficiently generate large certified primes by a variation on Pratt's non-deterministic algorithm [P] (generate the **NP** certificate and the resulting prime hand in hand) or by more sophisticated techniques [CQ], but the resulting distribution would not be uniform. Again, the most attractive solution in practice is to use Rabin's test as follows:

function *GenPrime* (l, k)
  { l is the size of the prime to be produced;
   k is a safety parameter discussed below }
  repeat
   $n \leftarrow$ randomly selected l digit odd integer
  until *RepeatRabin* (n, k) = ‘‘*prime*’’
  return n .

The resulting output is a *probabilistic prime* in the sense that we can never be assured that it is indeed prime. We can nonetheless increase our confidence in the number’s primality by increasing the safety parameter k. (What a shame that Rabin’s algorithm can certify those cryptographically useless composite numbers whereas it can only give probabilistic information on the useful primes! — which is precisely why [GK, AH]’s algorithms are of such (as yet theoretical) interest.)

In order to use *GenPrime* for cryptographic purposes, it is important that its probability of returning a composite integer be estimated. The popular belief is that

$$\text{prob}[\mathit{GenPrime}(l, k) \text{ is composite}] \le 4^{-k}$$

because each of the k rounds of *RepeatRabin* has a probability smaller than ¼ of failing on any given composite number. If we repeatedly use *GenPrime* to produce m distinct "primes", we therefore expect on the average that less than $m \times 4^{-k}$ of them will turn out to be composite. For instance, Knuth writes: ‘‘if we certified a billion different primes with such a procedure [4], the expected number of mistakes would be less than $\frac{1}{1000000}$’’ [Kn, page 379].

This assertion may be true [5], but the reason is wrong. Indeed, it could *only* be true because prob[*Rabin* (n) = ‘‘*prime*’’] is so much smaller than ¼ on most composite numbers. Should the error probability be *exactly* ¼ on every composite odd integer, the number of expected errors would be **significantly larger** than $10^9 \times 4^{-25} \approx 10^{-6}$.

From now on, let us assume we use a hypothetical test (that we shall continue to call *Rabin*) such that

$$\text{prob}[\mathit{Rabin}(n) = \text{‘‘}\mathit{prime}\text{’’} \mid n \text{ is indeed prime}] = 1$$

as before, whereas

$$\text{prob}[\mathit{Rabin}(n) = \text{‘‘}\mathit{prime}\text{’’} \mid n \text{ is in fact composite}] = \tfrac{1}{4} \quad (\mathit{exactly})\ .$$

It turns out that the error probability of each instantiation of *GenPrime* (l, k) depends on the size of the desired prime. As one can easily compute from Lemma 1 and Theorem 2 below, the expected number of errors exceeds $l \times 10^{-6}$ provided $35 \le l \le 2 \times 10^{13}$. In particular, if one wanted one billion 1000-digit primes, the expected number of mistakes would exceed 0.001. To be more provocative, if one had a need for one billion one-billion-digit primes, running Rabin’s test a mere 25 times per "prime" would result in an expected 1022 composite numbers among them. Worse still, each call to *GenPrime* $(10^{15}, 25)$ has a roughly 50% chance of producing a composite number!

In a nutshell, the reason for this confusion is that $\text{prob}[X \mid Y] \ne \text{prob}[Y \mid X]$ in general. In particular, if X stand for ‘‘n is composite’’ and Y for ‘‘*RepeatRabin* (n, k) returned ‘‘*prime*’’’’, then it is true that $\text{prob}[Y \mid X] \le 4^{-k}$, but this does *not* allow us to conclude that $\text{prob}[X \mid Y] \le 4^{-k}$ as well. In

---

4. Knuth does not *explicitly* say how he would use Rabin’s test to certify those billion primes, except that he would run it ‘‘25-times-in-a-row’’ on each of them. It is our interpretation that he meant something along the lines of *RepeatRabin*(•, 25). Of course, Knuth’s assertion is vacuously true if taken literally: if the integers thus certified are indeed ‘‘one billion primes’’, no mistakes are possible at all!

5. We *think* the assertion is true, but we have not yet actually carried out the calculation necessary to prove it.

order to get an estimate on prob$[X \mid Y]$, which is the cryptographically relevant probability, it is necessary to have an *a priori* probability that n is prime *before* even the first call to $Rabin(n)$ is performed. Fortunately, the prime number theorem [HW] comes to the rescue, which is where the size of the desired prime comes into play: the *a priori* probability that a randomly selected odd l-digit integer be prime is roughly $2/l\alpha$, where α stands for the natural logarithm of 10. More precisely,

Lemma 1

If n is uniformly, randomly selected among the odd l-digit integers, then

$$\mathrm{prob}[n \text{ is prime}] \approx \frac{2}{(l-1)\times\alpha}\left[1-\frac{10}{9l}\right] \approx 2/l\alpha \; .$$

Proof

Immediate from the prime number theorem [HW], which says that the number of primes not exceeding n is asymptotic to $n/\ln n$. Notice that this approximation is fairly accurate even for reasonably small values of n. For instance, there are 50,847,478 primes smaller than 10^9, whereas $n/\ln n$ would give about 48,254,942. □

Theorem 2

Let p be the probability that a uniformly, randomly selected odd l-digit number is prime. The probability that $GenPrime(l, k)$ returns a composite number is given by

$$\frac{1}{1+\frac{p}{1-p}\,4^k} \; .$$

This is about $(l\alpha/2)\times 4^{-k}$ provided l is substantially smaller than 4^k and about ½ when $l \approx 2\times 4^k/\alpha$.

Proof

Let X and Y be as before. Clearly, prob$[X] = 1-p$ and prob$[Y \mid X] = 4^{-k}$ (with our simplification to the effect that Rabin's test fails with probability *exactly* ¼ on composite numbers). We are interested in prob$[X \mid Y]$. We thus use the formula

$$\mathrm{prob}[X \mid Y] = \frac{\mathrm{prob}[X] \times \mathrm{prob}[Y \mid X]}{\mathrm{prob}[Y]} \; ,$$

which yields the theorem after routine algebraic manipulation because

$$\begin{aligned}\mathrm{prob}[Y] &= \mathrm{prob}[X]\times\mathrm{prob}[Y \mid X] + \mathrm{prob}[\textbf{not } X]\times\mathrm{prob}[Y \mid \textbf{not } X] \\ &= (1-p)\times 4^{-k} + p\times 1 \; . \quad \square\end{aligned}$$

Intuitively, the confidence we get in the number's primality from running Rabin's test several times must be weighted by the *a priori* overwhelming probability that it is composite if randomly chosen among the odd integers of a substantial size. For instance, if the size is $2\times 4^k/\alpha$, only about 1 in 4^k odd integers is prime. If a random odd integer of this size passes k rounds of Rabin's test, it is just as likely that this occurred because we were lucky enough to hit a prime or unlucky enough to observe such behaviour on a composite number!

4.2. How to Decide on the Primality of a Given Integer

Suppose some odd integer n is given to you. You are to decide whether you think it is prime or not. You therefore run Rabin's test for some number k of rounds, and it never finds n to be composite. What can you tell from this?

One obviously wrong answer is: ''this number is prime with probability $1-4^{-k}$''. This makes no sense because any *given* integer is either prime or not.

The classic answer is: ''I believe this number to be prime, and my error probability is at most 4^{-k} (in the sense that I expect to be wrong at most once every 4^k such statements if you quizz me long enough)''. This is wrong as well because *no* estimate on the error probability of ''I believe this number to be prime'' can be made without an *a priori* estimate on the probability that the number is prime. If you know that it was chosen randomly and uniformly among the odd integers of some given size l, section 4.1 tells you that you can still achieve an error probability below 4^{-k}, but at the cost of running Rabin's test for an additional (roughly) $\log_4 l$ rounds. However, if you do *not* know where the number comes from, you are at a *complete* loss.

Still, there *is* one thing you can say: ''I believe this number to be prime, and if I am wrong I have observed a natural phenomenon whose probability of occurrence was bounded by 4^{-k}''. This statement is certainly weak, but we cannot think of anything stronger one can infer in general from running Rabin's test any number of times.

As mentioned in the introduction, this observation is not restricted to primality testing. Whenever one runs *any* probabilistic algorithm that is not Las Vegas, care must be taken as to how to interpret the outcome. More implications of this issue are discussed in [BB].

Acknowledgements

The first observation sprung from an idea attributed to Silvio Micali by Charles H. Bennett (private communication) to the effect that any algorithm for extracting square roots modulo a prime can be turned into a probabilistic algorithm to decide primality. The main reason why our result applies directly only to numbers of the form $4j+3$ is that no efficient *deterministic* algorithm is known to extract square roots modulo a prime of the form $4j+1$.

No doubt the second observation has been made several times over by independant people, but we have not found any records of this. We wish to thank all those who showed interest for this work at the CRYPTO 86 meeting, in particular: Yvo Desmedt, Oded Goldreich, Gus Simmons and Moti Yung. Our thanks also go to Whitfield Diffie for granting us 12 minutes and 42 seconds to present these results.

References

[AH] Adleman, L. and M.-D. Huang, ''Recognizing primes in random polynomial time'', *presented at CRYPTO 86*, 1986.

[APR] Adleman, L., C. Pomerance and R. Rumeley, ''On distinguishing prime numbers from composite numbers'', *Annals of Mathematics*, Vol. 117, pp. 173-206, 1983.

[B] Babai, L., ''Monte Carlo algorithms in graph isomorphism testing", *Rapport de Recherches du Département de Mathématiques et de Statistiques*, Université de Montréal, D.M.S. #79-10, 1979.

[BB] Brassard, G. and P. Bratley, *Introduction to Algorithmics*, Prentice-Hall, Englewood Cliffs, New Jersey, to appear.

[CQ] Couvreur, C. and J. J. Quisquater, ‘‘An introduction to fast generation of large prime numbers’’, *Philips Journal of Research*, Vol. 37, nos. 5/6, pp. 231-264, 1982.

[G] Gill, J., ‘‘Computational complexity of probabilistic Turing machines’’, *SIAM Journal on Computing*, Vol. 6, no. 4, pp. 675-695, 1977.

[GK] Goldwasser, S. and J. Killian, ‘‘A provably correct and probably fast primality test’’, *Proceedings of the 18th Annual ACM Symposium on the Theory of Computing*, 1986.

[HW] Hardy, G. H. and E. M. Wright, *An Introduction to the Theory of Numbers*, Fifth edition, Oxford Science Publications, 1979.

[Kn] Knuth, D. E., *The Art of Computer Programming*, Volume 2 : *Seminumerical Algorithms*, Second edition, Addison-Wesley, Reading, Massachusetts, 1981.

[Kr] Kranakis, E., *Primality and Cryptography*, Wiley-Teubner Series in Computer Science, 1986.

[Mi] Miller, G. L., ‘‘Riemann’s hypothesis and tests for primality’’, *Journal of Computer and System Sciences*, Vol. 13, pp. 300-317, 1976.

[Mo] Monier, L., ‘‘Evaluation and comparison of two efficient probabilistic primality testing algorithms’’, *Theoretical Computer Science*, Vol. 11, pp. 97-108, 1980.

[PSW] Pomerance, C., J. L. Selfridge and S. Wagstaff, Jr., ‘‘The pseudoprimes to 25.10^9’’, *Mathematics of Computation*, Vol. 35, no. 151, pp. 1003-1026, July 1980.

[P] Pratt, V., ‘‘Every prime has a succinct certificate’’, *SIAM Journal on Computing*, pp. 214-220, 1975.

[R1] Rabin, M. O., ‘‘Probabilistic algorithms’’, in *Algorithms and Their Complexity: Recent Results and New Directions*, J. F. Traub (editor), Academic Press, New York, New York, pp. 21-39, 1976.

[R2] Rabin, M. O., ‘‘Digitalized signatures and public-key functions as intractable as factorization’’, *MIT/LCS/TR-212*, 1979.

[RSA] Rivest, R. L., A. Shamir and L. Adleman, ‘‘On digital signatures and public-key cryptosystems’’, *Communications of the ACM*, Vol. 21, no. 2, pp. 120-125, 1978.

[SS] Solovay, R. and V. Strassen, ‘‘A fast Monte Carlo test for primality’’, *SIAM Journal on Computing*, Vol. 6, pp. 84-85, 1977.