# Distribution of $\#\boldsymbol{E}^t$ for prime-order elliptic curves: Numerical results

David Leon Gil

June 8, 2015

## 1 Preface

This is a preliminary note of some numerical experiments; the results may be rather wrong.

## 2 Introduction

Twist-secure curves for elliptic curve cryptography are much in vogue these days; **djb** and others have proposed that twist-security is an essential safety condition for choosing curves.

**curve25519** cites **KaliskiJCryptology** as introducing the so-called "unsafe-twist" attack,f but I have been unable to find any evidence either there or in his (quite excellent) thesis, **KaliskiThesis** that he was aware of this attack.

## 3 Elliptic twin curves

We follow the definitions of (**ShparlinskiSutantyo** ), with some minor modifications.

Let $\mathbb{E}_j$ be the elliptic curve of invariant $j$, and $\mathbb{E}_j(\mathbb{F}_q)$ be its reduction over a finite field of characteristic $p > 5, n \geq 1$ with $p$ prime. Let and $t(\mathbb{E}_j(\mathbb{F}_q))$ be the trace of Frobenius of that elliptic curve.

Let $\widetilde{\mathbb{E}}_j(\mathbb{F}_q)$ be the non-trivial quadratic twist of $\mathbb{E}_j(\mathbb{F}_q)$ over the same field.

An *elliptic twin* is a pair consisting of a prime $p$, and two primes not equal to $p$ or 0, $(p, l, r)$ such that

$$\#\mathbb{E}_j(\mathbb{F}_p)\#\widetilde{\mathbb{E}}_j(\mathbb{F}_p) = l + r = 2p + 2 - l + r \tag{1}$$

It is clear that there exist elliptic twins over arbitrary prime fields, but the formulae of (**ShparlinskiSutantyo** ) are not amenable to characterizing local fluctuations in the density of such elliptic twins holding the finite field fixed.

# 4 Primes

We consider the non-Mersenne SECP primes, standardized for the use of the federal government in **recur** which are, where $N \coloneqq 2^{32}$:

$$\begin{aligned} \mathrm{P}_{224} &= N^7 - N^3 + N^0 \\ \mathrm{P}_{256} &= N^8 - N^7 + N^6 + N^3 - N^0 \\ \mathrm{P}_{384} &= N^{12} - N^4 - N^3 + N^1 - N^0 \end{aligned} \tag{2}$$

They are subset of the class of Generalized Mersennes defined by (**Solinas** ).

In future work, we plan to extend the study to consider the more general question of the distribution of group structure and curve exponent for the reduction of curves over fields for which the number of points is non-prime, and apply similar techniques with respect to the two curves proposed for IETF use, the nearly-Mersenne $\mathrm{M}_{255} = 2^{255} - 19$ and the Hamburg-Solinas trinomial $\mathrm{H}_{448} = 2^{448} - 2^{224} - 1$. [1]

(We probably won't extend this work to the Mersenne $\mathrm{M}_{521}$, as that particular calculation is pestiferous.)

# 5 Numerical methods

## 5.1 Finding prime-order curves

A slightly modified version of PARI/GP was used to calculate the traces of prime-order curves, based on code of **Hamburg** (The particular code used for this version of this paper may be found at (**junkpari** ).) Point-counting was aborted early if $\#\mathbb{E}_j$ was found to have a small prime factor.

## 5.2 Range

We calculate $T_f(j)$ for each $\mathbb{E}_j$ for $0 < j < 2^{20}, j \neq 1728$, then test $\#\mathbb{E}_j$ and $\#\mathbb{E}_j^t(\mathbb{Z}_q))$ for (pseudo-)primality.

[1] The Hamburg primes are "Karatsuba-friendly" **Hamburg** who was the first to realize that these are extremely efficient for elliptic curve cryptography.

(For this to be a reasonable procedure, it requires the assumption that $j$-invariant is not correlated with the probability of the curve being an elliptic twin, even on a local scale of $2^2$0.)

### 5.3 Results

| | $N_\pi$ | $N_{\pi'}$ | $N_{\pi'}/N_\pi$ |
|---|---|---|---|
| $P_{224}$ | 2790 | 31 | 1.1e-2 |
| $P_{256}$ | 1956 | 15 | 0.8e-2 |
| $P_{384}$ | 1131 | 20 | 1.8e-2 |

## 6 Future work

Because the above method results in fairly low precision in estimating $N_{\pi'}/N_\pi$ because of the small number of doubly-prime curves, we plan to use two slightly different methods, essentially similar to those of (**ShparlinskiSutantyo** ).

*Procedure 1.* We generate $N$ random $j$-invariants, $(j_{0,0}, \dots, j_{N,0})$, and increment each by 1 until we find a doubly prime curve.

*Procedure 2.* We generate random $j$-invariants repeatedly until we collect N primes.

The procedure we will use to generate the random $j$-invariants is, for both procedures,

```
for i in range(N):
  ok = False
  while not ok:
    input = shake256("Elliptic Twins over GF(0x%x),
                     procedure %u: i=%u, try=%u".format(procedure, P, j, try))
    maybep = l2b(shake256.squeeze(bitlen / 8))
    if maybep < P:
      ok = True
  js[i] = maybep
```

and similarly for procedure 2.

## 7 Concluding, mostly irrelevant aside

The quantity $1/\mathrm{p}(p) = N_\pi(\mathrm{p})/N_{\pi'}(\mathrm{p})$ is an estimator for the number of trials required, when choosing a prime curve uniformly at random in $\mathbb{F}_q$ for that curve to be an elliptic twin.

The probability, however, that no elliptic curves in a set of $N$ are elliptic twins is, of course,

$$1 - \prod_{0 \le i < n} 1 - \mathrm{p}_q \tag{3}$$

With respect to the curves standardized by NIST and generated by the NSA, this calculation gives a probability of very approximately $< 5\%$ of any of the curves over $\mathsf{P}_{224}, \mathsf{P}_{256}, \mathsf{P}_{384}$ being elliptic twins.

A Bayesian might thus conclude that it is more likely than not that the NSA's curves were not generated by a process that samples from a uniform distribution on prime-order curves over the chosen prime fields.

An appropriate prior might include both

- The fact that the NSA has specified only two of those curves for military use as part of its Suite B, and – based on publicly disclosed information – uses P-384 or classified cryptography primarily for its own use.
- The previously disclosed evidence, w.r.t. DES, that the NSA has additional safety criteria for cryptography which it does not disclose in standardization processes.

All this suggests that the NSA's choice was subject to additional, not publicly disclosed, safety criteria.

## 8 Acknowledgments

The patch to PARI/GP used is derived from a patch by Michael Hamburg.

All of the non-trivial mathematics is entirely derived from prior work by Robert Ransom. The only novelty is in the application to this particular question.

## Appendix. Cofactors for SafeCurves

This table is adapted (read stolen directly) from Bernstein, 2013.

| **Curve** | $h(\mathbb{E}_j)$ | $h(\mathbb{E}_j^t(\mathbb{Z}_q)))$ |
|---|---|---|
| `M221` | $2^3$ | $2^2$ |
| `E222` | $2^2$ | $\mathbf{2^2}$ |
| `secp224r1` | 1 | $3^2 \cdot 11 \cdot 47 \cdot 3015283 \cdot 40375823 \cdot 267983539294927$ |
| `Curve1174` | $2^2$ | $\mathbf{2^2}$ |
| `Curve25519` | $2^3$ | $2^2$ |
| `BN(2,254)` | 1 | $3^3 \cdot 3583 \cdot 298908837206431 \cdot 117111846430157829036976164449$ |
| `brainpoolP256` | 1 | $5^2 \cdot 175939 \cdot 492167257 \cdot 8062915307 \cdot 2590895598527 \cdot 4233394996199$ |
| `FRP256v1` | 1 | $7 \cdot 439 \cdot 11760675247 \cdot 3617872258517821$ |
| `secp256r1` | 1 | $3 \cdot 5 \cdot 13 \cdot 179$ |
| `secp256k1` | 1 | $3^2 \cdot 13^2 \cdot 3319 \cdot 22639$ |
| `E382` | $2^2$ | $\mathbf{2^2}$ |
| `M383` | $2^3$ | $2^2$ |
| `Curve383187` | $2^3$ | $2^2$ |
| `brainpoolP384` | 1 | $7 \cdot 11^2 \cdot 241 \cdot 5557 \cdot 125972502705620325124785968921221517$ |
| `secp384r1` | 1 | $\mathbf{1}$ |
| `Curve41417` | $2^3$ | $\mathbf{2^3}$ |
| `Ed448` | $2^2$ | $\mathbf{2^2}$ |
| `M511` | $2^3$ | $2^2$ |
| `E521` | $2^2$ | $\mathbf{2^2}$ |

# References

Bernstein, Daniel J. (2013). *SafeCurves: choosing safe curves for elliptic-curve cryptography: Twist security.* URL: `http://safecurves.cr.yp.to/twist.html`.

Shparlinski, Igor E and Daniel Sutantyo (2014). “Distribution of elliptic twin primes in isogeny and isomorphism classes”. In: *Journal of Number Theory* 137, pp. 1–15.