

# HYPERELLIPTIC CURVE CRYPTOSYSTEMS: CLOSING THE PERFORMANCE GAP TO ELLIPTIC CURVES

Jan Pelzl, Thomas Wollinger, Jorge Guajardo, and Christof Paar

Department of Electrical Engineering and Information Sciences  
Communication Security Group (COSY)  
Ruhr-Universität Bochum  
44780 Bochum, Germany  
email: {pelzl, wollinger, guajardo, cpaar}@crypto.rub.de

In 1976 Diffie and Hellman [DH76] revolutionized the field of cryptography by introducing the concept of public-key cryptography. Their key exchange protocol is based on the difficulty of solving the discrete logarithm (DL) problem over a finite field. Years later, [Kob87, Mil86] introduced a variant of the Diffie-Hellman key exchange, based on the difficulty of the DL problem in the group of points of an elliptic curve (EC) over a finite field. Since their introduction, elliptic curve cryptosystems (ECC) have been extensively studied not only by the research community but also in industry. In particular, there are several standards involving EC, such as the IEEE P1363 [P1399] standardization effort and the bank industry standards [ANS99]. It is important to point out that ECC benefit from shorter operand sizes when compared to RSA or DL based systems. This fact makes ECC particularly well suited for small processors and memory constrained environments.

In 1989 Koblitz suggested for the first time the generalization of EC to curves of higher genus, namely hyperelliptic curves (HEC) [Kob88]. In contrast to the ECC case, it has only been until recently that Koblitz's idea to use HEC for cryptographic applications, has been analyzed and implemented both in software [SS00, Eng99, SSI98, SS98, Kri97] and in more hardware-oriented platforms such as FPGAs [Wol01, BCLW02]. In 1999, [Sma99] concluded that there seems to be little practical benefit in using HEC, because of the difficulty of finding hyperelliptic curves and their relatively poor performance when compared to EC. However, recently the HEC group operation has been improved [MDM<sup>+</sup>02, Tak02, Har00, Lan02].

It is well known from the work in [Gau00] that the best algorithm to compute the discrete logarithm in the group of divisors of the Jacobian of a HEC is Pollard's rho method or one of its parallel variants [Pol78, vOW99]. For curves of genus higher than 4, [Gau00] showed that there exists an algorithm with complexity  $q^2$  where  $F_q$  is the field over which the HEC is defined. Thus, in this

work, we only consider HEC of genus less than four, as curves of higher genus are potentially insecure from the cryptographic point of view.

It is widely accepted that for cryptographic applications based on EC or HEC one needs a group order of size at least  $\approx 2^{160}$ . Thus, for HECC over  $F_q$  we will need  $g \cdot \log_2 q \approx 2^{160}$ , where  $g$  is the genus of the curve. In particular, for a curve of genus two, we will need a field  $F_q$  with  $|F_q| \approx 2^{80}$ , i.e., 80-bit long operands. Similarly, for curves of genus three, our discussion above implies 54-bit long operands. These field sizes make HEC specially promising for use in embedded environments where memory and speed are constrained, and where the above operand sizes seem well suited to their *small* processor architectures.

## Our Contributions

In the past years, the research community has worked hard to optimize the group operations of genus-2 curves [Har00, Nag00, Lan01]. For genus-3 hyperelliptic curves, [KGM<sup>+</sup>02] is the first attempt to obtain explicit<sup>1</sup> formulas for the group operation of HEC defined over odd characteristic fields. This work presents for the first time a generalized explicit formula for genus-3 curves including fields of characteristic 2. We optimized the formulae presented in [KGM<sup>+</sup>02] and we decreased the number of operations required to add and double two divisors. In particular, for certain curves our group doubling formula requires less than 50% of the field multiplications used by [KGM<sup>+</sup>02]. This improvement implies that one can implement HECC using our formulae twice as fast as when using [KGM<sup>+</sup>02] techniques.

Previous to our contribution, comparing HECC and ECC was a difficult task, because the operations involved in both systems were very different. Furthermore, the group operation of an ECC has a deterministic number of field operations regardless of the coordinate represen-

<sup>1</sup>By *explicit formula* we mean that we used only the underlying field arithmetic, i.e., no polynomial arithmetic is required.

tation. On the other hand, the complexity of HECC implementations based on the computation of polynomial greatest common divisors (gcd) is not deterministic<sup>2</sup>. With the introduction of explicit formulae for the computation of the group operation in a HECC as done here and in [Har00, Nag00, Lan01, KGM<sup>+</sup>02], the comparison is more exact as shown by a 10% difference between our theoretical and practical results. The aim of the comparison is to be able to estimate the performance of ECC versus HECC depending on the word sizes and the properties of the implemented field libraries. The most interesting results are that (1) in some special cases HECC can be faster than ECC of the same level of security and that (2) genus-3 curves are faster than genus-2 curves.

We support our theoretical findings with an HECC implementation on an embedded ARM processor. Our implementation uses the best explicit formulae for genus-2 and genus-3 curves. The timings are compared to the best known ECC implementations and we conclude that for our implementation genus-2 curves are about a factor of 2 slower than ECC, while genus-3 curves are approximately a factor of 1.5 slower.

## 1. REFERENCES

- [ANS99] ANSI X9.62-1999. The Elliptic Curve Digital Signature Algorithm. Technical report, ANSI, 1999.
- [BCLW02] N. Boston, T. Clancy, Y. Liow, and J. Webster. Genus Two Hyperelliptic Curve Coprocessor. In *CHES, LNCS*, New York, 2002. Springer Verlag.
- [DH76] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22:644–654, 1976.
- [Eng99] Andreas Enge. The extended Euclidean algorithm on polynomials, and the computational efficiency of hyperelliptic cryptosystems, November 1999. Preprint.
- [Gau00] P. Gaudry. An algorithm for solving the discrete log problem on hyperelliptic curves. In Bart Preneel, editor, *Advances in Cryptology — EUROCRYPT 2000*, volume LNCS 1807, pages 19–34, Berlin, Germany, 2000. Springer-Verlag.
- [Har00] Robert Harley. Fast Arithmetic on Genus Two Curves. Available at <http://cristal.inria.fr/harley/hyper/adding.txt> and [doubling.c](http://cristal.inria.fr/harley/hyper/doubling.c).
- [KGM<sup>+</sup>02] Junichi Kuroki, Masaki Gonda, Kazuto Matsuo, Jinhui Chao, and Shigeo Tsujii. Fast Genus Three Hyperelliptic Curve Cryptosystems. SCIS, Jan.29-Feb.1 2002.
- [Kob87] N. Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48:203–209, 1987.
- [Kob88] N. Koblitz. A Family of Jacobians Suitable for Discrete Log Cryptosystems. In Shafi Goldwasser, editor, *Advances in Cryptology - Crypto '88*, volume 403 of *Lecture Notes in Computer Science*, pages 94 – 99, Berlin, 1988. Springer-Verlag.
- [Kri97] Uwe Krieger. signature.c, February 1997. Diplomarbeit, Universität Essen, Fachbereich 6 (Mathematik und Informatik).
- [Lan01] Tanja Lange. Efficient Arithmetic on Hyperelliptic Curves, 2001. PhD Thesis. Universität Gesamthochschule Essen.
- [Lan02] Tanja Lange. Efficient Arithmetic on Genus 2 Hyperelliptic Curves over Finite Fields via Explicit Formulae. Cryptology ePrint Archive, Report 2002/121, 2002. <http://eprint.iacr.org/>.
- [MDM<sup>+</sup>02] Y. Miyamoto, H. Doi, K. Matsuo, J. Chao, and S. Tsuji. A Fast Addition Algorithm of Genus Two Hyperelliptic Curve. In *SCIS, IEICE Japan*, pages 497 – 502, 2002. in Japanese.
- [Mil86] V. Miller. Uses of elliptic curves in cryptography. In H. C. Williams, editor, *Advances in Cryptology — CRYPTO '85*, volume LNCS 218, pages 417–426, Berlin, Germany, 1986. Springer-Verlag.
- [Nag00] K. Nagao. Improving group law algorithms for Jacobians of hyperelliptic curves. In W. Bosma, editor, *ANTS IV*, volume 1838 of *Lecture Notes in Computer Science*, pages 439 – 448, Berlin, 2000. Springer Verlag.
- [P1399] *IEEE P1363 Standard Specifications for Public Key Cryptography*, November 1999. Last Preliminary Draft.
- [Pol78] J. M. Pollard. Monte carlo methods for index computation mod  $p$ . *Mathematics of Computation*, 32(143):918–924, July 1978.
- [Sma99] Nigel P. Smart. On the Performance of Hyperelliptic Cryptosystems. In *Advances*

<sup>2</sup>[Eng99] considers the *average* complexity of the gcd computation of polynomial defined over a finite field

in *Cryptology - EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 165 – 175, Berlin, 1999. Springer-Verlag.

- [SS98] Y. Sakai and K. Sakurai. Design of Hyperelliptic Cryptosystems in small Characteristic and a Software Implementation over  $\mathbb{F}_{2^n}$ . In *Advances in Cryptology - ASIACRYPT '98*, volume 1514 of *Lecture Notes in Computer Science*, pages 80 – 94, Berlin, 1998. Springer Verlag.
- [SS00] Y. Sakai and K. Sakurai. On the Practical Performance of Hyperelliptic Curve Cryptosystems in Software Implementation. volume E83-A NO.4, April 2000. IEICE Trans.
- [SSI98] Y. Sakai, K. Sakurai, and H. Ishizuka. Secure Hyperelliptic Cryptosystems and their Performance. In *Public Key Cryptography*, volume 1431 of *Lecture Notes in Computer Science*, pages 164 – 181, Berlin, 1998. Springer-Verlag.
- [Tak02] M. Takahashi. Improving Harley Algorithms for Jacobians of Genus 2 Hyperelliptic Curves. In *SCIS, IEICE Japan*, 2002. in Japanese.
- [vOW99] P. C. van Oorschot and M. J. Wiener. Paralle collision search with cryptanalytic applications. *Journal of cryptology*, 12(1):1–28, Winter 1999.
- [Wol01] Thomas Wollinger. Computer Architectures for Cryptosystems Based on Hyperelliptic Curves, 2001. Master Thesis, Worcester Polytechnic Institute.