FURTHER INVESTIGATIONS ON HARRIS ALGORITHM

Anton Iliev, Nikolay Kyurkchiev,
Asen Rahnev, Todorka Terzieva

Abstract. *In this note we will make further computational improvements of Harris algorithm [2, 12]. We improve speed using the technique of least absolute remainder [1]. Numerical experiment give us confidence that we receive new enhanced algorithm.*

Key words: Euclidean algorithm, Harris algorithm, hybrid algorithm, least absolute remainder.

2020 Mathematics Subject Classification: 11A05, 68W01

1. Introduction

Harris algorithm is well known hybrid iteration process which compute Greatest common divisor of two natural numbers a and b. In many classical and recent books and papers the Euclidean algorithm is well described, see [2]–[10] and [28]–[39]. Using symmetry properties of Euclidean iteration process we receive some computational benefits [11]–[27].

For testing purposes we will use the following computer: processor – Intel(R) Core(TM) i7-6700HQ CPU 2.60GHz, 2592 Mhz, 4 Core(s), 8 Logical Processor(s), RAM 16 GB, Microsoft Windows 10 Enterprise x64, Microsoft Visual C# 2017 x64.

2. Main Results

We present new iteration process, which improve Harris algorithm:

Algorithm 1.

```
int g = 0;
if ((a & 1) == 0 && (b & 1) == 0)
do { a >>= 1; b >>= 1; g++; }
while ((a & 1) == 0 && (b & 1) == 0);
u = a; v = b;
while ((u & 1) == 0) u >>= 1;
```

```
while ((v & 1) == 0) v >>= 1;
if (u > v) do { u %= v;
if (u < 1) { gcd = v << g; break; }
if ((u & 1) == 0)
{ do u >>= 1; while ((u & 1) == 0);
if (u == 1) { gcd = u << g; break; } }
else { ar = v - u;
if (u > ar)
{ u = ar;
do u >>= 1; while ((u & 1) == 0);
if (u == 1) { gcd = u << g; break; } } }
v %= u;
if (v < 1) { gcd = u << g; break; }
if ((v & 1) == 0)
{ do v >>= 1; while ((v & 1) == 0);
if (v == 1) { gcd = v << g; break; } }
else { ar = u - v;
if (v > ar)
{ v = ar;
do v >>= 1; while ((v & 1) == 0);
if (v == 1) { gcd = v << g; break; } } }
} while (true);
else do { v %= u;
if (v < 1) { gcd = u << g; break; }
if ((v & 1) == 0)
{ do v >>= 1; while ((v & 1) == 0);
if (v == 1) { gcd = v << g; break; } }
else { ar = u - v;
if (v > ar)
{ v = ar;
do v >>= 1; while ((v & 1) == 0);
if (v == 1) { gcd = v << g; break; } } }
u %= v;
if (u < 1) { gcd = v << g; break; }
if ((u & 1) == 0)
```

```
{ do u >>= 1; while ((u & 1) == 0);
if (u == 1) { gcd = u << g; break; } }
else { ar = v - u;
if (u > ar)
{ u = ar;
do u >>= 1; while ((u & 1) == 0);
if (u == 1) { gcd = u << g; break; } } }
} while (true);
```

as well as its recursive version

Algorithm 2.

```
static long Euclid(long u, long v, int g)
{
long ar;
if (u > v) { u %= v;
if (u < 1) { return v << g; }
if ((u & 1) == 0)
return Euclid(u >> 1, v, g);
else { if (u == 1) { return u << g; }
ar = v - u;
if (u > ar)
{ u = ar;
if ((u & 1) == 0)
return Euclid(u >> 1, v, g); } } }
else { v %= u;
if (v < 1) { return u << g; }
if ((v & 1) == 0)
return Euclid(u, v >> 1, g);
else { if (v == 1) { return v << g; }
ar = u - v;
if (v > ar)
{ v = ar;
if ((v & 1) == 0)
return Euclid(u, v >> 1, g); } } }
return Euclid(u, v, g);
}
```

The recursive function should be called by:

```
int g = 0;
if ((a & 1) == 0 && (b & 1) == 0)
do { a >>= 1; b >>= 1; g++; }
while ((a & 1) == 0 && (b & 1) == 0);
u = a; v = b;
while ((u & 1) == 0) u >>= 1;
while ((v & 1) == 0) v >>= 1;
gcd = Euclid(u, v, g);
```

Numerical Example.

For testing purposes of Algorithms 1 and 2 we will use the following main function:

```
long a, b, gcd, d1 = 0, u, v;
for (int i = 1; i < 100000001; i++) { a = i; b = 200000002 - i;
//here are placed the source code of algorithm 1 and
//calling of recursive algorithm 2
d1 += gcd;
}
Console.WriteLine(d1);
```

CPU time results are:

CPU time of Algorithm 1 is: **26.799 seconds.**

CPU time of Algorithm 2 is: **42.989 seconds.**

For the same numerical example Harris algorithms [2, 12] gave the following results – iterative 31.620 seconds and recursive 68.119 seconds.

3. Conclusion

We give how in Harris algorithm can be implemented the technique of least absolute remainder and this leads to computational speed improvements.

Acknowledgments

This work has been accomplished with the financial support by the Grant No. BG05M2OP001-1.001-0003, financed by the Science and Education for Smart Growth Operational Program (2014–2020) and co-financed by the European Union through the European structural and Investment funds.

References

[1] T. Moore, On the Least Absolute Remainder Euclidean Algorithm, *Fibonacci Quarterly*, **30**, 1992, 161–165.

[2] V. Harris, An algorithm for finding the greatest common divisor, *Fibonacci Quarterly*, 8, 1970, 102–103.

[3] Th. Cormen, Ch. Leiserson, R. Rivest, Cl. Stein, *Introduction to Algorithms*, 3rd ed., The MIT Press, Cambridge, 2009.

[4] J. Tembhurne, S. Sathe, New Modified Euclidean and Binary Greatest Common Divisor Algorithm, *IETE Journal of Research*, 62, No. 6, 2016, 852–858.

[5] K. Garov, A. Rahnev, *Textbook-notes on programming in BASIC for facultative training in mathematics for 9.–10. Grade of ESPU*, Sofia, 1986, (in Bulgarian).

[6] A. Golev, *Textbook on algorithms and programs in C#*, University Press "Paisii Hilendarski", Plovdiv, 2012, (in Bulgarian).

[7] T. Terzieva, *Introduction to web programming*, University Press "Paisii Hilendarski", Plovdiv, 2021, ISBN: 978-619-202-623-3, (in Bulgarian).

[8] T. Terzieva, *Development of algorithmic thinking in the Informatics Education*, University Press "Paisii Hilendarski", Plovdiv, 2021, ISBN: 978-619-202-622-6, (in Bulgarian).

[9] T. Terzieva, *Educational tools for teaching in digital environment*, University Press "Paisii Hilendarski", Plovdiv, 2021, (in Bulgarian).

[10] S. Enkov, *Programming in Arduino Environment*, University Press "Paisii Hilendarski", Plovdiv, 2017, (in Bulgarian).

[11] A. Iliev, N. Kyurkchiev, A Note on Knuth's Implementation of Euclid's Greatest Common Divisor Algorithm, *International Journal of Pure and Applied Mathematics*, **117**, 2017, 603–608.

[12] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Harris-Stein Modification of Euclidean Algorithm for Greatest Common Divisor. II, *International Journal of Pure and Applied Mathematics*, 120, No. 3, 2018, 379–388.

[13] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Tembhurne–Sathe Modification of Euclidean Algorithm for Greatest Com-

mon Divisor. IV, *Dynamic Systems and Applications*, 28, No. 1, 2019, 143–152.

[14] A. Iliev, N. Kyurkchiev, A. Golev, A Note on Knuth's Implementation of Extended Euclidean Greatest Common Divisor Algorithm, *International Journal of Pure and Applied Mathematics*, **118**, 2018, 31–37.

[15] A. Iliev, N. Kyurkchiev, A Note on Euclidean and Extended Euclidean Algorithms for Greatest Common Divisor for Polynomials, *International Journal of Pure and Applied Mathematics*, **118**, 2018, 713–721.

[16] A. Iliev, N. Kyurkchiev, A Note on Knuth's Algorithm for Computing Extended Greatest Common Divisor using SGN Function, *International Journal of Scientific Engineering and Applied Science*, **4** No. 3, 2018, 26–29.

[17] A. Iliev, N. Kyurkchiev, *New Trends in Practical Algorithms: Some Computational and Approximation Aspects*, LAP LAMBERT Academic Publishing, Beau Bassin, 2018.

[18] A. Iliev, N. Kyurkchiev, The faster extended Euclidean algorithm, *Collection of scientific works from conference*, Pamporovo, Bulgaria, 28–30 November 2018, 2019, 21–26.

[19] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Least Absolute Remainder Algorithm for Greatest Common Divisor. III, *Neural, Parallel, and Scientific Computations*, **27** No. 1, 2019, 1–9.

[20] A. Iliev, N. Kyurkchiev, A. Rahnev, *Nontrivial Practical Algorithms: Part 2*, LAP LAMBERT Academic Publishing, Beau Bassin, 2019.

[21] A. Iliev, N. Valchanov, T. Terzieva, Generalization and Optimization of Some Algorithms, *Collection of scientific works of National Conference "Education in Information Society"*, Plovdiv, ADIS, 12–13 May 2009, 2009, 52–58, (in Bulgarian).

[22] A. Iliev, N. Kyurkchiev, A. Rahnev, New Extended Algorithm for Finding Greatest Common Divisor, *Neural, Parallel, and Scientific Computations*, 28 No. 1, 2020, 89–95.

[23] A. Iliev, N. Kyurkchiev, A. Rahnev, Efficient Binary Algorithm for Kronecker Symbol, *Communications in Applied Analysis*, 25 No. 1, 2021, 11–21.

[24] A. Iliev, N. Kyurkchiev, A. Rahnev, Efficient Algorithm for Kronecker Symbol, *International Electronic Journal of Pure and Applied Mathematics*, 15 No. 1, 2021, 23–30.

[25] A. Iliev, N. Kyurkchiev, A. Rahnev, T. Terzieva, Efficient Binary Ex-

tended Algorithm Using SGN Function, *International Journal of Differential Equations and Applications*, **20**, No. 2, 2021, 179–186.

[26] A. Iliev, N. Kyurkchiev, A. Rahnev, A Refinement of the Knuth's Extended Euclidean Algorithm for Computing Modular Multiplicative Inverse, *Communications in Applied Analysis*, 25, No. 1, 2021, 23–37.

[27] A. Iliev, N. Kyurkchiev, A. Rahnev, A Refinement of the Extended Euclidean Algorithm, *International Electronic Journal of Pure and Applied Mathematics*, 15 No. 1, 2021, 33–44.

[28] D. Knuth, *The Art of Computer Programming, Vol. 2, Seminumerical Algorithms*, 3rd ed., Addison-Wesley, Boston, 1998.

[29] A. Rahnev, K. Garov, O. Gavrailov, *Textbook for extracurricular work using BASIC*, MNP Press, Sofia, 1985, (in Bulgarian).

[30] A. Rahnev, K. Garov, O. Gavrailov, *BASIC in examples and tasks*, Government Press "Narodna prosveta", Sofia, 1990, (in Bulgarian).

[31] N. Kasakliev, *C# Programming Guide*, University Press "Paisii Hilendarski", Plovdiv, 2016, (in Bulgarian).

[32] A. Rahnev, N. Pavlov, N. Valchanov, T. Terzieva, *Object Oriented Programming*, Lightning Source UK Ltd., London, 2014.

[33] P. Thapar, U. Batra, Implementation of Elliptical Curve Cryptography Based Diffie-Hellman Key Exchange Mechanism in Contiki Operating System for Internet of Things, *International Journal of Electrical and Electronics Research (IJEER)*, **10** No. 2, 2022, 335–340.

[34] V. Matanski, An Efficient Binary Algorithm for Solving Equation $GCD * 2^{|J-K|} = X * A0 + Y * B0$, Proceedings of Anniversary International Scientific Conference "Computer Technologies and Applications", 15-17 September 2021, Pamporovo, Bulgaria, *Plovdiv University Press*, 79–86, ISBN: 978-619-202-702-5.

[35] H. Gyulyustan, A Note on Euclidean Sequencing Algorithm, Proceedings of the Scientific Conference "Innovative ICT for Digital Research Space in Mathematics, Informatics and Educational Pedagogy", Pamporovo, 7-8.11.2019, *Plovdiv University Press*, 2020, 57–63, ISBN: 978-619-202-572-4.

[36] P. Kyurkchiev, V. Matanski, The Faster Euclidean Algorithm for Computing Polynomial Multiplicative Inverse, Proceedings of the Scientific Conference "Innovative ICT in Research and Education: Mathematics, Informatics and Information Technologies", Pamporovo, 29–30 November 2018, 2019, 43–48, ISBN: 978-619-202-439-0.

[37] V. Matanski, P. Kyurkchiev, The Faster Lehmer's Greatest Common Divisor Algorithm, Proceedings of the Scientific Conference "Innovative ICT in Research and Education: Mathematics, Informatics and Information Technologies", Pamporovo, 29–30 November 2018, 2019, 37–42, ISBN: 978-619-202-439-0.

[38] Z. Ibran, E. Aljatlawi, A. Awin, On Continued Fractions and Their Applications, *Journal of Applied Mathematics and Physics*, **10**, 2022, 142–159.

[39] Y. Fan, G. Chen, M. Cui, Formalization of Finite Field GF(2^n) Based on COQ, *Computer Science*, **47** No. 12, 2020, 311–318.

Anton Iliev[1,*], Nikolay Kyurkchiev[2],
Asen Rahnev[3], Todorka Terzieva[4]
[1, 2, 3, 4] University of Plovdiv "Paisii Hilendarski",
Faculty of Mathematics and Informatics,
24, Tzar Asen Str., 4000 Plovdiv, Bulgaria,
[1, 2] Institute of Mathematics and Informatics,
Bulgarian Academy of Sciences,
Acad. G. Bonchev Str., Bl. 8, 1113 Sofia, Bulgaria
Corresponding author: `aii@uni-plovdiv.bg`