Научна конференция „Иновационни ИКТ за дигитално научноизследователско пространство по математика, информатика и педагогика на обучението“, 7-8 ноември 2019 г., Пампорово, България
Scientific Conference “Innovative ICT for Digital Research Area in Mathematics, Informatics and Pedagogy of Education”, 7-8 November 2019, Pamporovo, Bulgaria

A NOTE ON EUCLIDEAN SEQUENCING ALGORITHM

H. Gyulyustan

Abstract. In recent works [12]–[31] the authors gave many optimized versions of classical widespread realizations of Euclidean algorithm. Here we present a new faster version of the Euclidean Sequencing Algorithm (ESA) [40] which algorithm has been proven to be optimal for searching binary cycles with minimal variance in [40].

Keywords: *Euclid's algorithm, Euclidean algorithm for binary cycles with minimal variance, reduced number of operations*

1. Introduction

The Euclidean algorithms and their modifications [1]–[40] are widely spread in many practical oriented tasks.

Statement of the problem [40]. Let $A:=[a_1,a_2,...,a_n]$ be an alphabet of $n:=|A|$ distinct symbols and $m\in\mathbb{N}^n$ is a vector of n positive integers representing prescribed multiplicities of said symbols in such a way that $\left[a_k\,|m_k\right]:=\left[\underbrace{a_k,a_k,...,a_k}_{m_k\text{ times}}\right],\quad a_k\in A.$ The couple $S:=(A,m)$ shall be termed a "cyclic sequencing problem". Let $N:=\sum_{k=1}^{n}m_k$.

Let $C:\mathbb{Z}_N\to A$ be a mapping from the group of rest classes modulo N to the alphabet with $C_j:=C\left([\,j]_N\right)$.

Дата на получаване: 01.11.2019 г.
Дата на рецензиране: 20.03.2020 г.
Дата на публикуване: 02.06.2020 г.

Let $J_k := C^{-1}(a_k) = \{[j]_N \in \mathbb{Z}_N | C_j = a_k\}$ be the counter image of the k th symbol in the alphabet. Let $\Omega(A,m) := \{C : \mathbb{Z}_N \to A | m_k = |J_k|, \forall k \in \{1,...,n\}\}$ be the set of "admissible" cycles.

In [40] the following theorem has been proved:

Theorem 1. Let (A,m) be a cyclic sequencing problem with a binary symbol alphabet. Then the cycle $C \in \Omega(A,m)$ returned by the Euclidean Sequencing Algorithm minimizes variance.

Let us consider the following Euclidean Sequencing Algorithm (ESA) [40]:

//initialization

$A_0 = a_{\arg\max\{m_1,m_2\}}$; $P_1 = |J_{\arg\max\{m_1,m_2\}}|$;

$B_0 = a_{\arg\min\{m_1,m_2\}}$; $D_1 = |J_{\arg\min\{m_1,m_2\}}|$;

$i = 0$; $R_0 = 1$;

//looping is iterated until a null rest is found

while $R_i > 0$ do

$i = i + 1$;

$N_i = P_i + D_i$;

$Q_i = \lfloor P_i / D_i \rfloor$;

$R_i = P_i - Q_i D_i$;

$A_i = A_{i-1}^{Q_i} B_{i-1}$;

$B_i = A_{i-1}$;

$P_{i+1} = D_i$;

$D_{i+1} = R_i$;

end

//finalization

$C = A_i^{D_i}$;

Algorithm 1. Euclidean Sequencing Algorithm (ESA)

2. Main results

We suggest the following new optimization of Euclidean Sequencing Algorithm (ESA):

//initialization

$A_0 = a_{\arg\max\{m_1,m_2\}}$; $P_1 = \left|J_{\arg\max\{m_1,m_2\}}\right|$;

$B_0 = a_{\arg\min\{m_1,m_2\}}$; $D_1 = \left|J_{\arg\min\{m_1,m_2\}}\right|$;

$i = 0$; $R_{-1} = P_1$; $R_0 = D_1$;

//looping is iterated until a null rest is found

do

$i = i + 1$;

$Q_i = \lfloor R_{i-2} / R_{i-1} \rfloor$;

$R_i = R_{i-2} - Q_i R_{i-1}$;

$A_i = A_{i-1}^{Q_i} B_{i-1}$;

$B_i = A_{i-1}$;

while $R_i > 0$;

//finalization

$C = A_i^{R_{i-1}}$;

Algorithm 2. Optimized Euclidean Sequencing Algorithm (ESA)

Obviously, the Algorithm 2 is optimized version of Algorithm 1 because for every $i \geq 1$ it is satisfied $P_i = R_{i-2}$, $D_i = R_{i-1}$ and $N_i = R_{i-2} + R_{i-1}$. As a result the computation of P_i, D_i and N_i in Algorithm 1 is unnecessary in new Algorithm 2.

Numerical Example.

Using Algorithm 1 the following example is given in [40]:

i	$\mathcal{A}_i$	N_i	P_i	D_i	Q_i	R_i	A_i	B_i
0							$A_0 = a_1$	$B_0 = a_2$
1	$[A_0, B_0]$	32	18	14	1	4	$A_1 = A_0^1 B_0$	$B_1 = A_0$
2	$[A_1, B_1]$	18	14	4	3	2	$A_2 = A_1^3 B_1$	$B_2 = A_1$
3	$[A_2, B_2]$	6	4	2	2	0	$A_3 = A_2^2 B_2$	
							$C = A_3^2$	

For the same example Algorithm 2 gives the following results:

i	$\mathcal{A}_i$	Q_i	R_i	A_i	B_i
-1			18		
0			14	$A_0 = a_1$	$B_0 = a_2$
1	$[A_0, B_0]$	1	4	$A_1 = A_0^1 B_0$	$B_1 = A_0$
2	$[A_1, B_1]$	3	2	$A_2 = A_1^3 B_1$	$B_2 = A_1$
3	$[A_2, B_2]$	2	0	$A_3 = A_2^2 B_2$	
				$C = A_3^2$	

From both Algorithms 1 and 2 we obtain [40]:

$$\begin{aligned} C &= A_3^2 \\ &= (A_2^2 B_2)^2 \\ &= ((A_1^3 B_1)^2 A_1)^2 \\ &= (((A_0 B_0)^3 A_0)^2 A_0 B_0)^2 \\ &= a_1 a_2 a_1 a_2 a_1 a_2 a_1 a_1 a_2 a_1 a_2 a_1 a_2 a_1 a_1 a_2 \\ &\quad a_1 a_2 a_1 a_2 a_1 a_2 a_1 a_1 a_2 a_1 a_2 a_1 a_2 a_1 a_1 a_2. \end{aligned}$$

References

[1] A. Akritas, A new method for computing polynomial greatest common divisors and polynomial remainder sequences, *Numerische Mathematik*, 52 (1988), 119–127.

[2] A. Akritas, G. Malaschonok, P. Vigklas, On the Remainders Obtained in Finding the Greatest Common Divisor of Two Polynomials, *Serdica Journal of Computing*, 9 (2015), 123–138.

[3] L. Ammeraal, *Algorithms and Data Structures in C++*, John Wiley & Sons Inc., New York (1996).

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

[5] F. Chang, Factoring a Polynomial with Multiple-Roots, *World Academy of Science, Engineering and Technology*, 47 (2008), 492–495.

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

[7] A. Drozdek, *Data Structures and Algorithms in C++*, 4th ed., Cengage Learning (2013).

[8] 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)

[9] S. Goldman, K. Goldman, *A Practical Guide to Data Structures and Algorithms Using JAVA*, Chapman & Hall/CRC, Taylor & Francis Group, New York (2008).

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

[11] M. Goodrich, R. Tamassia, D. Mount, *Data Structures and Algorithms in C++*, 2nd ed., John Wiley & Sons Inc., New York (2011).

[12] 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.

[13] 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.

[14] A. Iliev, N. Kyurkchiev, A. Rahnev, A Note on Adaptation of the Knuth's Extended Euclidean Algorithm for Computing Multiplicative Inverse, *International Journal of Pure and Applied Mathematics*, 118 (2018), 281–290.

[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 Least Absolute Remainder Euclidean Algorithm for Greatest Common Divisor, *International Journal of Scientific Engineering and Applied Science*, 4, No. 3 (2018), 31–34.

[17] 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.

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

[19] A. Iliev, N. Kyurkchiev, 80th Anniversary of the birth of Prof. Donald Knuth, *Biomath Communications*, 5 (2018), 7 pp.

[20] A. Iliev, N. Kyurkchiev, New Realization of the Euclidean Algorithm, *Collection of scientific works of Eleventh National Conference with International Participation Education and Research in the Information Society*, Plovdiv, ADIS, June 1–2, (2018), 180–185. (in Bulgarian)

[21] A. Iliev, N. Kyurkchiev, New Organizing of the Euclid's Algorithm and one of its Applications to the Continued Fractions, *Collection of scientific works from conference "Mathematics. Informatics. Information Technologies. Application in Education"*, Pamporovo, Bulgaria, 10–12 October 2018, (2019), 199–207.

[22] A. Iliev, N. Kyurkchiev, The faster Euclidean algorithm, *Collection of scientific works from conference*, Pamporovo, Bulgaria, 28–30 November 2018, (2019), 15–20.

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

[24] P. Kyurkchiev, V. Matanski, The faster Euclidean algorithm for computing polynomial multiplicative inverse, *Collection of scientific works from conference*, Pamporovo, Bulgaria, 28–30 November 2018, (2019), 43–48.

[25] V. Matanski, P. Kyurkchiev, The faster Lehmer's greatest common divisor algorithm, *Collection of scientific works from conference*, Pamporovo, Bulgaria, 28–30 November 2018, (2019), 37–42.

[26] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement Euclidean Algorithm for Greatest Common Divisor. I, *Neural, Parallel, and Scientific Computations*, 26, No. 3 (2018), 355–362.

[27] 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.

[28] 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.

[29] A. Iliev, N. Kyurkchiev, A. Rahnev, A New Improvement of Tembhurne–Sathe Modification of Euclidean Algorithm for Greatest Common Divisor. IV, *Dynamic Systems and Applications*, 28, No. 1 (2019), 143–152.

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

[31] 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, http://scigems.math.bas.bg/jspui/handle/10525/1356

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

[33] Hr. Krushkov, A. Iliev, Practical programming guide in Pascal, Parts I and II, Koala press, Plovdiv (2002), (in Bulgarian).

[34] P. Nakov, P. Dobrikov, Programming=++Algorithms, 5th ed., Sofia (2015), (in Bulgarian).

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

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

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

[38] R. Sedgewick, K. Wayne, *Algorithms*, 4th ed., Addison-Wesley, Boston (2011).

[39] A. Stepanov, *Notes on Programming* (2007).

[40] L. Ghezzi, R. Baldacci, A Euclidean Algorithm for Binary Cycles with Minimal Variance, arXiv:1804.01207v1, 4 April 2018, (2018), https://arxiv.org/abs/1804.01207v1

Faculty of Mathematics and Informatics,
University of Plovdiv "Paisii Hilendarski",
24, Tzar Asen Str., 4000 Plovdiv, Bulgaria,
E-mail: hasan@uni-plovdiv.bg

БЕЛЕЖКА ВЪРХУ ЕВКЛИДОВИЯ АЛГОРИТЪМ ЗА СЕКВЕНИРАНЕ

Хасан Гюлюстан

Резюме: В скорошни работи [12]–[31] авторите са дали много оптимизирани версии на класически широко разпространени реализации на евклидовия алгоритъм. Тук ще представим нова по-бърза версия на евклидовия алгоритъм за секвениране (ESA) [40], който алгоритъм е доказан като оптимален за търсене на двоични цикли с минимална вариативност в [40].