Performance Comparison of Some Addition Chain Methods Based on Integer Family

M. F. A. Kadir, M. A. Mohamed, R. Mohamad, M. Mamat and A. Muhammed

Abstract A generalized version of an addition chain problem, in which one must find a chain that simultaneously satisfies a sequence on integer in ascending order, is NP-complete. There is no known algorithm which can calculate an optimal addition chain for a given number with any guarantees of reasonable timing or small memory usage. Several methods were introduced to calculate relatively short chain and they are most used to support scalar multiplication operation tailored to limited computational resources in elliptic curve cryptography. In reality, one method is no better than the other except on certain occasions and only for specific integers. In this studies, we evaluate some existing addition chain methods against each other for their competitive performance by categorizing integers into various groups as the input. This result can be used as a benchmark for which method is suitable in which condition anticipated.

Keywords Addition chain · Elliptic curve cryptography · NP-complete Heuristic method · Composition method

1 Introduction

Nowadays, information security is at the greatest importance in which communication over open networks and storage of data in digital form plays a key role in daily life. The science of cryptography provides efficient tools to secure information.

M. F. A. Kadir (✉) · M. A. Mohamed · M. Mamat
Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Kuala Terengganu, Malaysia
e-mail: fadzil@unisza.edu.my

R. Mohamad
Department of System and Networking, Universiti Tenaga Nasional, Kajang, Malaysia

A. Muhammed
Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Seri Kembangan, Malaysia

K. J. Kim and N. Baek (eds.), *Information Science and Applications 2018*,
Lecture Notes in Electrical Engineering 514,
https://doi.org/10.1007/978-981-13-1056-0_22

Cryptography is defined as the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity identification, and data origin authentication. Due to their tampered resistance, cryptosystems are often implemented on constraint memory devices such as smart cards. For such cases elliptic curve cryptosystems (ECC) [1, 2] is considered to be the most appropriate. The ECC exploited the discrete logarithm problem on a general elliptic curve that has no subexponential time solution. The major advantage of ECC is that a small key of size 160-bits can provide comparable security level with other cryptographic standards such as RSA of 1024-bits but with much faster and and more efficient execution.

The basic operation of ECC involves scalar and multi-scalar multiplication. Addition chain method has been widely used to improved efficiency of large number operation such that found in ECC. An addition chain (AC) for a number n is a sequence, such that each new member is the sum of two earlier (not necessarily distinct) ones. The length of an AC for an integer n is calculated as the number of terms other than the first one. The number of operations is directly proportional to the number of terms. This way, efficiency can be achieved if we can have shorter chain that is, the number of operations can be reduced by shortening the sequence.

Many AC method have been introduced. This is due to finding the optimal chain for a set of numbers was proven to be NP-complete [3]. These methods aimed at generating chain closest to optimal value. AC methods can be grouped into two major family, heuristics and metaheuristics. One method is known to be no better than the other except on certain occasions for certain groups of integers. However, there is lack of performance evaluation in term of length generated to compare the efficiency of different methods. Therefore, some performance analysis focusing on finding the shorter length need to be done. In this study, we conduct performance evaluation by comparing the length of selected AC for heuristic methods to find the best methods (with shortest length) by focusing on comparing seven methods, evaluating integer from 2 to 180,000.

The remainder of this paper is structured as follows; Sect. 2 summaries the review of literature. Section 3 states the materials and methods used in this project. These will be followed by the results and discussion in Sect. 4 and conclusion in Sect. 5.

2 Related Works

There are many AC methods that have been introduced in order to find sub-optimal solution. The literatures divide them into two categories that are meta-heuristics and heuristics [4]. A meta-heuristics is an iterative master process that guides and modifies the operations of subordinates heuristics to efficiently produce high-quality solutions. Meta-heuristics support decision making with robust tools that provide high quality solution to important application in business, engineering, economics and science. Common target of meta-heuristics are to solve optimization problem

usually known by their complexities and it is inspired by analogies related to other factor such as natural, chemical, biological, electrical and thermal.

In solving AC problem, many meta-heuristics methods have been proposed. Genetic algorithms was first used by [5], allowing one-point crossover and uniform mutation, followed by [6] with the use of two-point crossover together with a local search mutation operator and a repair mechanism built within the initialization of the population. The most notable one, which employ a representation based on factorial number system together with neighborhood functions and distribution functions is credited to [7]. Various other methods such as ant colony optimization [8], artificial immune system [9], population-based optimization [10], and simulated annealing [11]. However, the most notable one is that of an evolutionary programming (EP) [12]. EP simulates evolution at species level. Therefore, no crossover operator is employed. In this proposed approach, an individual is represented at genotype level, that is an individual is a feasible AC. The fitness value of each individual is the length of the AC. Therefore, shorter strings are preferred. An advantage of the EP algorithm comprises the solution encoding with suitable fitness function and initial population, a mutation operator, and the survivor selection mechanism. These elements are easy to implement comparing to operators such as crossover and parent selection found in genetic algorithm. However, there are a few issues with meta-heuristics techniques such as consume a lot time to get optimal results, dependencies of specific cases for good result and complexities in implementation.

Heuristic is a much simpler techniques which seeks good solutions at a reasonable computational cost. Heuristic is designed and tuned for some specific problem. Heuristics are criteria, methods, or principles for deciding which among several alternative courses of action promises to be the most effective in order to achieve some goal. In broad, we can categorized these methods into a few different ways. By looking into the way of input representation we have binary or m-ary methods by using different radix, and unsigned and signed methods by allowing both positive and negative integer values. Either way, the idea is to reduce the number of operations that are addition and doubling, as much as possible. This can be done via manipulating the representation such that the digit representing the operation is reduced to the digit unpresented such that found in Binary Method (BM) [13]. Moreover, the location and adjacency of the digit can also be impactful for some methods such as Non-Adjacent Form (NAF) [14], and Complementary Recoding (CR) [15], Decomposition Method (DM) [16], Composition Method (CM) [17], Signed Decomposition Method (SDM) [18], and Signed Composition Method (SCM) [19].

3 Materials and Method

In general the studies of AC can be represented by a framework in Fig. 1. This section discusses how do we conduct the experiment and the tools required. The performance metric used is the length of the generated AC. The chosen methods are Binary Method, Non Adjacent Form, Complementary Recording, Decomposition

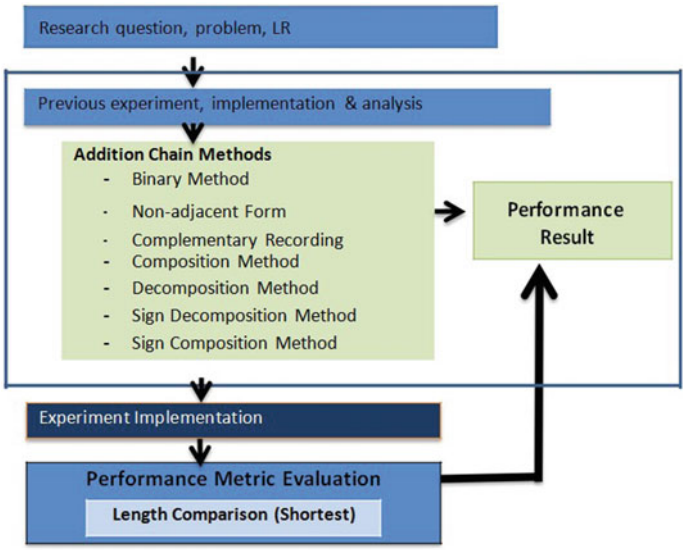

Fig. 1 Investigative framework

Method, Composition Method, Signed Decomposition Method, and Signed Composition Method.

We implemented these algorithms based on that of suggested by the original articles and perform the performance measurement under Dev C++ environment. The GNU Multiple Precision Arithmetic Library (GMP) is a free library for arbitrary-precision arithmetic, operating on signed integers, rational numbers, and floating point numbers is used to support large integer operation. GMP has a rich set of functions, and the functions have a regular interface. Besides that main target applications of GMP are cryptography applications and research same as in our project. To install the GMP library a few tool should also be installed which are MinGW 3.4.2—used as interfaced, GCC and GCC++, and MySYS 1.0.10. Meanwhile, for graphing and analysis the GNUPlot 4.6 and Microsoft Excel 2010 are used respectively.

This whole experiment can be divided into 3 phases. First, we develop all the seven algorithms and measure their performance for integers from 1 to 180,000. An overall from this experiment, we can see the distribution of length for each integer. Second, we do the comparison using four group of test which are:

Test 1: We compare every two methods, totalling 21 combinations altogether, to find out how many wins (shorter), looses (longer) and draws (equal) in terms of the length of AC measured for every integer in each block.
Test 2: By splitting integers n into blocks such that $2^N + 1 \leq n \leq 2^{N+1}$ specified by N from 3 to 17, we compute an average length for each block.
Test 3: By separating odd $(2k + 1)$ from even $(2k)$ integers, we study the distribution of an AC generated by each method.

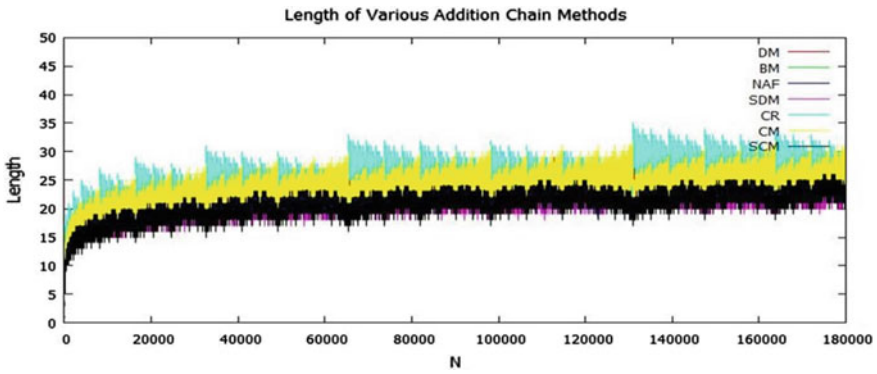

Fig. 2 Distribution length of 7 methods

Finally, we plot the graph for every test done to observe the distribution and comparison then analyze the result.

4 Results and Discussion

In this section, we discuss the results that we have been obtained from our three experiment setup.

For Test 1, Fig. 2 shows the distribution of the length of AC produced by the seven different methods. Observably, the lengths for all methods are increasing when the integer are growing but the rates of growth are fairly small. Table 1 shows the list of comparisons between methods for 21 combinations. These comparisons calculate the wins, loose and draw in term of the length of AC. The method with more wins (shorter chain) is considered as winner. The more the method wins the shorter the method generated the length. The SCM has wins all the comparison against the other methods whereas, CM and BM produces chain of having the same lengths.

For Test 2, the average length of AC for each block specified by the value of N are calculated by adding all the lengths for every integer and divide by N. Table 2 shows the result for $3 \leq N \leq 17$. We observed that for $N \leq 8$, SDM produces the shortest average among all methods, whereas for $N > 8$, SCM seems to outclass all other methods.

From the Test 1 and Test 2, the methods can be ranked as SCM (shortest), SDM, NAF, DM, BM and CM, and CR. In both tests, we found out SDM and SCM methods are competing between each other in term of performance. The length of SCM becomes shortest when the integer number growth.

In Test 3, by separating even from odd integers, we plot graphs representing the length of AC produced by each method. For odd integer, Fig. 3 shows that SDM and SCM are competing to each other to become the method with the shortest AC. For

Table 1 Length comparison

Methods	Wins	Loose	Draw	Results
DM versus BM	77647	41789	60564	DM
DM versus NAF	28829	113899	37272	NAF
DM versus CR	129105	31093	19802	DM
DM versus SDM	0	112992	67008	SDM
DM versus CM	77647	41789	60564	DM
DM versus SCM	18194	125631	36175	SCM
BM versus NAF	13010	122471	44519	NAF
BM versus CR	108370	53573	18057	BM
BM versus SDM	12786	140287	26927	SDM
BM versus CM	0	0	180000	
BM versus SCM	0	133111	46889	SCM
NAF versus CR	150918	0	29082	NAF
NAF versus SDM	41942	72350	65708	SDM
NAF versus CM	122471	13010	44519	NAF
NAF versus SCM	0	48929	131071	SCM
CR versus SDM	5317	163572	11111	SDM
CR versus CM	53573	108370	18057	CM
CR versus SCM	0	164937	15063	SCM
SDM versus CM	140287	12786	26927	SDM
SDM versus SCM	49898	50296	79806	SCM
CM versus SCM	0	133111	46889	SCM

Table 2 Average lengths

$2^N + 1 \le n \le 2^{N+1}$	DM	BM	NAF	SDM	CR	CM	SCM
N = 3	4.5	4.625	4.875	4.5	5.75	4.625	4.5
N = 4	5.9375	6.0625	6.125	5.75	7.125	6.0625	5.8125
N = 5	7.34375	7.53125	7.46875	7.125	8.5625	7.53125	7.125
N = 6	8.71875	9.01563	8.78125	8.42188	10.0313	9.01563	8.45313
N = 7	10.1797	10.5078	10.1172	9.76563	11.5156	10.5078	9.78125
N = 8	11.6328	12.0039	11.4453	11.1016	13.0078	12.0039	11.1133
N = 9	13.0898	13.502	12.7793	12.4473	14.5039	13.502	12.4453
N = 10	14.54	15.001	14.1113	13.7793	16.002	15.001	13.7783
N = 11	16.0073	16.5005	15.4448	15.1138	17.501	16.5005	15.1113
N = 12	17.4746	18.0002	16.7778	16.4514	19.0005	18.0002	16.4446
N = 13	18.9456	19.5038	18.1149	17.7887	20.5072	19.5038	17.7815
N = 14	20.4062	20.9957	19.4403	19.1133	21.994	20.9957	19.107
N = 15	21.8826	22.5	20.7778	20.4524	23.5001	22.5	20.4445
N = 16	23.3583	24	22.1111	21.7878	25	24	21.7778
N = 17	24.8326	25.5	23.4445	23.4445	26.5	25.5	23.1111

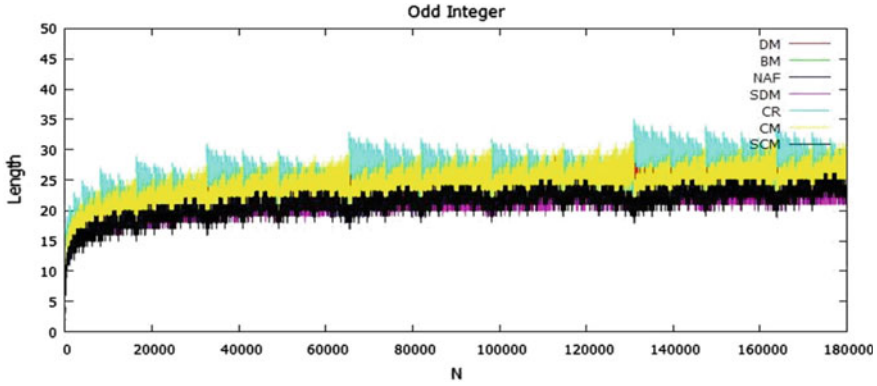

Fig. 3 Distribution of length for odd integer

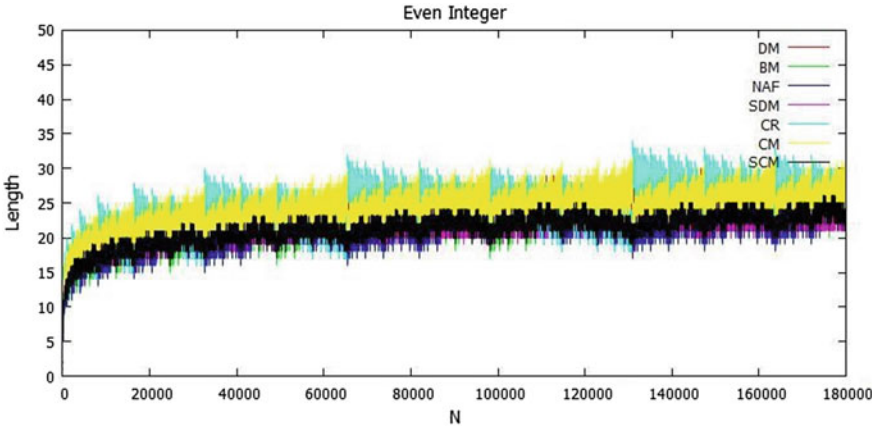

Fig. 4 Distribution of length for even integer

even integers, Fig. 4 shows that NAF method has contributed to the production of the shortest AC alongside with SDM and SCM.

5 Conclusion

Competitively, SDM and SCM generate the shortest length. Both methods use rules as their basis. However, SCM has outclassed SDM for large integers and therefore is more recommended for cryptographic implementation which utilizes large integers.

Acknowledgements This study is funded by Ministry of Higher Education Malaysia (FGRS/1/2017/ICT03/UNISZA/02/1(RR228)). Special thanks to University of Sultan Zainal Abidin for providing facilities to develop and evaluate the proposed framework.

References

1. Mohamed MA (2014) A survey on elliptic curve cryptography. Appl Math Sci 8(153–156):7665–7691
2. Koblitz N (1987) Elliptic curve cryptosystems. Math Comput 48:203–209
3. Downey F, Leong B, Seith R (1981) Computing sequences with addition chains. SIAM J Comput 10:638–646
4. Noma AM, Muhammed A, Mohamed MA, Zulkarnain ZA (2017) A review on heuristics for addition chain problem: towards efficient public key cryptosystems. J Comput Sci 13(8):275–289
5. Cruz-Cortés N, Rodríguez-Henríquez F, Juárez-Morales R, Coello Coello CA (2005) Finding optimal addition chains using a genetic algorithm approach. In: Hao Y et al (eds) Computational intelligence and security (CIS 2005). LNCS, vol 3801. Springer, Berlin, Heidelberg
6. Osorio-Hernandez L, Mezura-Montes E, Cruz-Cortes N, Rodriguez-Henriquez F (2009) A genetic algorithm with repair and local search mechanisms able to find minimal length addition chains for small exponents. In: IEEE congress on evolutionary computation (CEC 2009), pp 1422–1429
7. Rodriguez-Cristerna A, Torres-Jimenez J (2013) A genetic algorithm for the problem of minimal Brauer chains for large exponents. In: Melin P, Castillo O (eds) Soft computing applications in optimization, control, and recognition. Studies in fuzziness and soft computing, vol 294. Springer, Berlin, Heidelberg
8. Nedjah N, Mourelle LDM (2006) Towards minimal addition chains using ant colony optimization. J. Math Model Algorithms 5:525–543
9. Cruz-Cortés N, Rodríguez-Henríquez F, JuárezMorales R, Coello-Coello CA (2008) An artificial immune system heuristic for generating short addition chains. IEEE Trans Evol Comput 12:1–24
10. León-Javier A, Cruz-Cortés N, Moreno-Armendáriz MA, Orantes-Jiménez S (2009) Finding minimal addition chains with a particle swarm optimization algorithm. In: Aguirre AH, Borja RM, Garciá CAR (eds) Advances in artificial intelligence (MICAI 2009). LNCS, vol 5845. Springer, Berlin, Heidelberg
11. Jose-Garcia A, Romero-Monsivais H, Hernandez-Morales CG, Rodriguez-Cristerna A, Rivera-Islas I, Torres-Jimenez J (2011) A simulated annealing algorithm for the problem of minimal addition chains. In: Antunes L, Pinto HS (eds) Progress in artificial intelligence (EPIA 2011). LNCS, vol 7026. Springer, Berlin, Heidelberg (2011)
12. Dominguez-Isidro S, Mezura-Montes E, Cruz-Cortés N, Rodríguez-Henríquez F (2015) Evolutionary programming for the length minimization of addition chains. Eng Appl Artif Intell 37:125–134
13. Knuth DE (1981) The art of computer programming. Seminumeral algorithms, vol 2, 2nd edn. Addison-Wesley
14. Okeya K, Schmidt-Samoa K, Spahn C, Takagi T (2004) Signed binary representations revisited. In: Proceedings of CRYPTO'2004. LNCS 3152, pp 123–139
15. Balasubramaniam P, Karthikeyan E (2007) Elliptic curve scalar multiplication algorithm using complementary recoding. Appl Math Comput 190:51–56
16. Mohamed MA, Md Said MR, Mohd Atan KA, Ahmad Zulkarnain Z (2011) Shorter addition chain for smooth integers using decomposition method. Int J Comput Math 88(11):2222–2232
17. Mohamed MA, Mohd Atan KA (2012) Rule based representation of integer for a new addition chain method. Appl Math Sci 6(30):1497–1503
18. Mohamed MA, Said MRMd (2015) A hybrid addition chain method for faster scalar multiplication. Wseas Trans Commun 14:144–152
19. Mohamed MA, Ahmad A, Mohamed RR, Said MRM (2017) Shorter addition-subtraction chain with signed composition method. Int J Eng Technol 9(2):299–308