Journal of Engineering and Applied Sciences 11 (3): 640-643, 2016
ISSN: 1816-949X

An Initial Solution for Addition Chain Optimization Problem

[1]Adamu Muhammad Noma, [2]Mohamad Afendee Mohamed,
[1]Abdullah Muhammed and [1]Zuriati Ahmad Zulkarnain
[1]Faculty of Computer Science and Information Technology,
Universiti Putra Malaysia, Serdang, 43000 Selangor, Malaysia
[2]Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin,
Besut, 22200 Terengganu, Malaysia

Abstract: Addition chain for an integer corresponds to the number of multiplications in modular exponentiation using the integer as the exponent. Modular exponentiation is the basis of public key crypto systems. Finding an optimal addition chain is an NP-complete problem. Therefore, metaheuristic have been favored in searching for the near optimal solution. In this study we proposed probabilistic method for generating an initial solution to the problem, suitable for optimization using metaheuristic. It is aimed at providing foundation for an efficient constructive metaheuristic algorithm that can find high quality solution to the problem. To ensure diversity, the proposed method generates both star and non-star steps. It generates valid addition chain with diverse steps that is not far from optimal. Test on the generated chains also shows that a suitable metaheuristic can efficiently turn them to nearest optimal ones.

Key words: Addition chain, metaheuristic, modular exponentiation, public key crypto systems, probabilistic method

INTRODUCTION

Modular exponentiation has an additive properties and thus finding shortest addition chain for an exponent e-corresponds to that of the number of multiplications required in the exponentiation. An addition chain for is a sequence of natural numbers $\alpha_0 = 1$, $\alpha_1 = 2$, α_3, ..., $\alpha_r = e$ with the property that $\forall i \geq 1$, $\alpha_i = \alpha_j + \alpha_k$, $i > j \geq k$. The length of the chain e is the shortest for which there exists an addition chain of is denoted by.

The problem of finding shortest addition chain exists for over hundred years ago (Dellac, 1894). The fundamental question leading to addition chain was raised by Scholz (1937). Theoretical studies of the problem can be found in Balega (1976), Brauer (1939), Knuth (1998), Mignotte and Tall (2011), Thurber (1973) and Yao (1976). Exhaustive approaches were carryout by Bahig (2006), Bahig (2011), Clift (2011), Thurber (1999). Its generic case was established to be an NP-complete problem (Balega, 1976). As such heuristics, Bos and Coster (1989), Koc (1995), Lee *et al.* (2006), Park *et al.* (1999) and metaheuristic (Dominguez-Isidro *et al.*, 2015; Cortes *et al.*, 2008; Jose-Garcia *et al.*, 2011; Javier *et al.*, 2009) have become alternative approaches. To expect a high quality solution using metaheuristic a good and diverse initial solution is necessary that can easily be improved towards the optimal result.

The popular algorithm for addition chain is binary method (Knuth, 1998). A generalized algorithm based on the method is termed m-ary (Koc, 1995; Park *et al.*, 1999). Yet more improved versions than the m-ary are Constant Length Non-zero Window (CLNW) and Variable-Length Non-zero Window (VLNW) method (Koc, 1995), both refers to as Sliding Window Methods (SWM). Binary and its family (m-ary and SWM) are efficient in terms implementation but could not guarantee returning any (near) optimal chain. More effective ones include factor and power tree methods (Knuth, 1998). However, both are only feasible for relatively small exponents.

Theoretically establishing that addition chain is an NP-complete problem (Downey *et al.*, 1981) leads to exploring various metaheuristic. Those applied include Artificial Immune Systems (AIS) (Cortes *et al.*, 2008), Genetic Algorithm (GA) (Osorio-Hernandez *et al.*, 2009), Particle Swamp Optimization (PSO) (Javier *et al.*, 2009), Evolutionary Programming (EP) (Dominguez-Isidro *et al.*, 2015) and ant Colony Optimization (ACO) (Nedjah and

Corresponding Author: Adamu Muhammad Noma, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang, 43000 Selangor, Malaysia

Mourelle, 2006). They are evolutionary or swamp intelligence-based and therefore population-oriented that utilize only star steps. In this study, we proposed probabilistic method of generating initial solution for the addition chain problem utilizing both the star and non-star steps. This is to set a foundation for a constructive metaheuristic application that can generate an even high quality solution in an efficient way: utilizing minimal computing and memory resource.

MATERIALS AND METHODS

We denote a valid arbitrary addition chain generated and its length as A (e) and l_A (e), respectively. Given e the objective of the generator is to generate an A (e) with as sort as possible l_A (e). To achieve this, we categorize the possible steps in constructing the elements of the chain into four (4) as defined thus:

- Doubling: if $\alpha_i = 2\alpha_{i-1}$
- Lucas: if $\alpha_i = \alpha_{i-1}+\alpha_{i-2}$
- Other-star: if $\alpha_i = \alpha_{i-1}+\alpha_k$, $k\epsilon\{i-3, ..., 0\}$
- Non-star: if $\alpha_i = \alpha_j+\alpha_k$, $j\geq k\geq i-2$

An element α_i, $i>0$ of an addition chain A is said to be a star element if $\alpha_i = \alpha_{i-1}+\alpha_k$: $k<i$; an addition chain A (e) is said to be a star chain (Brauer, 1939) for e if $\forall\alpha_{i>0}\in$A (e) are star elements; the shortest length of star chain for an integer is denoted as l*(e). Step 1-3 are therefore responsible for constructing star element.

In generating the valid chain, sequence of elements consisting of one of the four defined steps is necessarily constructed. Steps 1 and 2 can easily be employed but 3 and 4 require some probability function to select the needed element (s) from among the available ones. In the selection process we assume normal distribution in α_0, α_{i-3} where each element is equally likely to be selected. The algorithm therefore calls random function rnd [x, y] which returns k in [0, i-3] (or j, k in [0, i-2] in the case of step to form an α_i.

The addition chain generator: To generate a valid addition chain for an arbitrary integer e, the algorithm begins with sub-chain: 1-2. This is followed by successively building the chain through randomly selecting among the 4 steps to create elements of the chain, until the current element $\alpha_i = m\geq[e/2]$, e being the integer for which the chain is to be generated. Each of the steps is equal likely of being utilized. For as long as $\alpha_m<e$, the algorithm continue to add the largest possible element (s) until a valid chain is formed such that $\alpha_r = e$. This is done by successively adding max $\{\alpha_{j<m}\leq e-\alpha_i\}$ to the current α_i to form the next element $\alpha_{i=i+1}$, for $i>m$. The addition chain generator is presented as algorithm 1.

Algorithm 1: Generate (e)

Input {e}; Output: {addition chain A (e) = $\{\alpha_0,\alpha_1, ..., \alpha_r = e\}$}.
set A (e) = $\{\alpha_0 = 1, \alpha_1 = 2\}$; (initialize A)
While $e_r<e/2$ do (here e_r indicates the last elements of the chain)
set r = r+1; (increment by 1)
set step = rnd [0, 3]; (select step at random)
if step = 0 set $\alpha_r = 2\times\alpha_{r-1}$; (apply doubling)
else if step = 1 set $\alpha_r = \alpha_{r-1}+\alpha_{r-2}$; (apply lucas)
else if step = 2 set $\alpha_r = \alpha_{r-1}+\alpha_{rnd[r-3, 0]}$; (apply other-star)
else set $\alpha_r = \alpha_{rnd[r-3, 0]}+\alpha_{rnd[r-3, 0]}$; (apply non-star)
end while;
while $\alpha_r<e$ do
set r = r+1
set $\alpha_r = \alpha_{r-1}+\alpha_k$ (where $\alpha_k = \max\{\alpha_i\leq e-\alpha_{r-1}: 0\leq i<r\}$);
end while;
return A (e)

Note that algorithm 1 can generate any type of addition chain consisting of any combination of the four elements. This, of cause, depends on the probability values assign to the steps.

RESULTS AND DISCUSSION

To evaluate the addition chain generator we compared the result generated for set of small integers for which their optimal results are well established. Additionally, we generated with the algorithm, addition chain for medium sets of integers to demonstrate its responds with the integer size. Where the optimal result is established we compare the result with the optimal one.

Greedy approach is another alternative method to generate initial solution for metaheuristic applications. Star step is the only candidate suitable in this respect as both lucas and doubling steps generate a deterministic chain that cannot be further improved. In particular employing greedy doubling step (step 2) defaults to Binary Method. We therefore compare the generator with the greedy star. Given a range of integers e = [1, x] an accumulated addition chain for the integer set is defined as:

$$T(x)\sum_{e=1}^{x} lA(e) \quad (1)$$

We compared results for addition chains for in the range [1, 4096] against their corresponding optimal ones. It is presented in Table 1.

The result in Table 1 shows that generally our addition generator is consistent and closer to the optimal results than the greedy star. It consistently generates, on average, 1.5 times the optimal result while the greedy star generates from 2 up to 4 times the optimal.

To demonstrate the suitability of the generator in metaheuristic application, we evaluate average and minimal chain generated by both probability generator

Table 1: Comparing cumulative addition chain for sets of small integers

e∈	Optimal	Greedy star	Probability generator
[1.512]	4924.000	9.965	6.759
[1.1000]	10.808	21.052	15.178
[1.1024]	11.115	21.718	15.606
[1.2000]	24.063	49.973	34.618
[1.2048]	24.731	50.862	35.619
[1.4096]	54.408	120.500	79.716

Table 2: Average versus minimum addition chain length for n-bits integers after 100 runs

e∈ (bits)	Greedy starr	Probability generator	e∈ (bits)	Greedy star	Probability generator
16	55 (43)	27 (22)	48	405 (330)	97 (83)
32	192 (155)	63 (52)	64	687 (572)	132 (115)

and the greedy star for 100 runs. In this case, we utilize 16, 32, 48 and 64 bit randomly generated integers. The result is presented in Table 2.

In the Table 2, the shortest chain generated is enclosed in the bracket along with the average. Note the variance between the average and the shortest chain by both the probabilistic method and the greedy star one. Both approaches ensure diversity in generating the elements and can therefore serve as initial solution for metaheuristic. Furthermore, the former generate much shorter chain on both average and on the basis of shortest chain generated: this shows that it is even much suitable for a constructive metaheuristic and the corresponding can effectively be transformed to an (near) optimal one.

CONCLUSION

Probabilistic addition chain generator effectively generate a valid addition chain for an integers that can easily be transformed to optimal one by suitable metaheuristic. Although, the chain generated is far from optimal it is consistent with respect to the optimal one. And it is flexible enough to serve as good initial solution in generating (nearest) optimal chain using suitable metaheuristic.

We intend to introduce improvement on the weighing function which we believe will further enhance the effectiveness of the generator algorithm. We will also introduce suitable metaheuristic that could efficiently enhance the result towards an optimal one.

REFERENCES

Bahig, H.M., 2006. Improved generation of minimal addition chains. Comput., 78: 161-172.

Bahig, H.M., 2011. Star reduction among minimal length addition chains. Comput., 91: 335-352.

Balega, E.G., 1976. The additive complexity of a natural number. Soviet Math. Doklad, 17: 5-9.

Bos, J. and M. Coster, 1989. Addition Chain Heuristics. In: The Theory and Application of Cryptology. Brassard, G. (Ed.). Springer, New York, USA., ISBN: 978-0-387-97317-3, pp: 400-407.

Brauer, A., 1939. On addition chains. Bull. Am. Math. Soc., 45: 736-739.

Clift, N.M., 2011. Calculating optimal addition chains. Comput., 91: 265-284.

Cortes, N.C., F.R. Henriquez and C.A.C. Coello, 2008. An artificial immune system heuristic for generating short addition chains. IEEE. Trans. Evol. Comput., 12: 1-24.

Dellac, H., 1894. Question 49. L'Intermediaire des Mathematiciens, 1: 20-20.

Dominguez-Isidro, S., E. Mezura-Montes and Osorio-Hernandez, 2015. Evolutionary programming for the length minimization of addition chains. Eng. Appl. Artif. Intelligence, 37: 125-134.

Downey, P., B. Leong and R. Sethi, 1981. Computing sequences with addition chains. SIAM J. Comput., 10: 638-646.

Javier, A.L., N.C. Cortes, M.A.M. Armendariz and S.O. Jimenez, 2009. Finding Minimal Addition Chains with a Particle Swarm Optimization Algorithm. In: Mexican International Conference on Artificial Intelligence. Arturo, H.A., M.B. Raul and A.R.G. Carlos (Eds.). Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-642-05257-6, pp: 680-691.

Jose-Garcia, A., H. Romero-Monsivais, C.G. Hernandez-Morales, A. Rodriguez-Cristerna and I. Rivera-Islas *et al.*, 2011. A Simulated Annealing Algorithm for the Problem of Minimal Addition Chains. In: Portuguese Conference on Artificial Intelligence. Antunes, L. and H.S. Pinto (Eds.). Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-642-24768-2, pp: 311-325.

Knuth, D.E., 1998. Evaluation of Powers. In: Art of Computer Programming, Volume 2: Seminumerical Algorithms, 3rd Edition, Knuth, D.E. (Ed.)., Addison-Wesley, USA., pp: 461-485.

Koc, C.K., 1995. Analysis of sliding window techniques for exponentiation. Comput. Math. Appl., 30: 17-24.

Lee, Y., H. Kim, S.M. Hong and H. Yoon, 2006. Expansion of sliding window method for finding shorter addition/subtraction-chains. Intl. J. Netw. Secur., 2: 34-40.

Mignotte, M. and A. Tall, 2011. A note on addition chains. Intl. J. Algebra, 5: 269-274.

Nedjah, N. and L.D.M. Mourelle, 2006. Towards minimal addition chains using ant colony optimisation. J. Math. Modell. Algorithms, 5: 525-543.

Osorio-Hernandez, L.G., E. Mezura-Montes, N. Cruz-Cortes and F. Rodriguez-Henriquez, 2009. A genetic algorithm with repair and local search mechanisms able to find minimal length addition chains for small exponents. Proceedings of the 2009 IEEE Congress on Evolutionary Computation, May 18-21, 2009, Trondheim, pp: 1422-1429.

Park, H., K. Park and Y. Cho, 1999. Analysis of the variable length nonzero window method for exponentiation. Comput. Math. Appl., 37: 21-29.

Scholz, A., 1937. Aufgaben und losungen, 253. Jahresbericht der Deutschen Mathematiker-Vereinigung, 47: 41-42.

Thurber, E.G., 1973. On addition chains $1 (mn)\leq 1 (n)-b $ and lower bounds for $ c (r) $. Duke Math. J., 40: 907-913.

Thurber, E.G., 1999. Efficient generation of minimal length addition chains. SIAM J. Comput., 28: 1247-1263.

Yao, A.C.C., 1976. On the evaluation of powers. SIAM J. Comput., 5: 100-103.