Pedaqoji Universitetin Xəbərləri. Riyaziyyat və təbiət elmləri seriyası – 2022, C. 70, №2, s. 20-30
Transactions of Pedagogical University. Series of mathematics and natural sciences – 2022, V. 70, №2, pp. 20-30

UOT 51-7

A.P.Orucəliyev, N.M.Məmmədova
Dövlət Təhlükəsizliyi Xidmətinin Heydər Əliyev adına Akademiyası
akiforuc@rambler.ru

SÜRƏTLİ VURMA ALQORİTMLƏRİ VƏ ONLARIN İNFORMASİYANIN KRİPTOQRAFİK MÜHAFİZƏSİNDƏ ROLU

Açar sözlər: *hesablama alqoritmləri, hesablama mürəkkəbliyi, kriptoqrafik mühafizə, kriptosistem, faktorizasiya, kvant kompüterləri, sadə ədəd*

Məqalədə kifayət qədər böyük natural ədədlər üçün sürətli vurma alqoritmlərindən olan Karatsuba, Şönhare-Ştrassen və Fürer alqoritmlərinin hesablama mürəkkəbliyi araşdırılır və bu alqoritmlərin informasiyanın kriptoqrafik mühafizəsində rolu tədqiq edilir.

А.П.Оруджалиев, Н.М.Маммедова

АЛГОРИТМЫ БЫСТРОГО УМНОЖЕНИЯ И РОЛЬ ИХ В КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЕ ИНФОРМАЦИИ

Ключевые слова: *вычислительные алгоритмы, вычислительная сложность, криптографическая защита, криптосистем, факторизация, квантовые компьютеры, простая числа.*

В статье рассматривается вычислительная сложность алгоритмов быстрого умножения Каратцуба, Шенхаре-Штрассен, Фюрера для достаточно больших натуральных чисел и исследованы роль этих алгоритмов в криптографических защиты информации.

A.P.Orujaliev, N.M.Mammadova

FAST MULTIPLICATION ALGORITHMS AND THEIR ROLE IN CRYPTOGRAPHIC PROTECTION OF INFORMATION

Keywords: *computational algorithms, computational complexity, cryptographic protection, cryptosystems, factorization, quantum computers, prime numbers.*

The article reviewers the computational complexity of the fast multiplication algorithms of Karatsuba, Shenhare-Strassen, Fuhrer for sufficiently large natural numbers and investigates the role of these algorithms in cryptographic information protection.

Müxtəlif real məsələlərin həlli zamanı, o cümlədən informasiyanın

kriptoqrafik mühafizəsi üsullarının realizasiyası zamanı kifayət qədər böyük ədədlərin hasillərinin hesablanması tələb olunur. Belə hasillərin tapılması kifayət qədər böyük kompüter resursları (yaddaş, vaxt) tələb etdiyindən onların realizasiyası xüsusi hesablama üsulları olmadan səmərəsiz bir iş olardı. Böyük ədədlərin hasillərinin xüsusi hesablama üsullarını öyrənən elm sahəsi sürətli hesablama nəzəriyyəsidir. Sürətli hesablama nəzəriyyəsi hesablama riyaziyyatının verilmiş funksiyanın verilmiş dəqiqliklə qiymətlərinin ən az əməliyyatlardan istifadə etməklə hesablanmasını öyrənən sahəsidir. Bu nəzəriyyənin əsasını təşkil edən alqoritmlər bəzən kompüterin hesablama sürətini əhəmiyyətli dərəcədə artırır, bəzən isə ölçüsü adi hesablama üsullarının tətbiqini qeyri-mümkün edən məsələlərin həllinə imkan verir.

Sürətli hesablama alqoritmlərinin yaradılması ideyasının tarixi 1960-cı ildən başlayır. Məhz 1960-cı ildən başlayaraq informasiya nəzəriyyəsi üzrə tədqiqatlar aparan məşhur sovet riyaziyyatçısı akademik A.N. Kolmoqorovun (1903-1987) rəhbərlik etdiyi seminarda hesablama mürəkkəbliyi ilə bağlı ilk məsələ müzakirəyə çıxarılmış və bu məsələ digər alim və tədqiqatçıların böyük marağına səbəb olmuşdur. Kompüter texnikasının tətbiqinin yeni-yeni sahələrinin meydana çıxdığı bu dövrdə hesablama alqoritmlərinə olan böyük ehtiyac belə alqoritmlərin yaradılması zərurətini artırdı. Bu zaman hesablama alqoritmləri içərisində polinomial mürəkkəbliyə malik alqoritmlərin yaradılması xüsusilə aktual idi. Belə ki, yalnız məhdud həcmdə hesablama və mövcud kompüter texnikasının imkanları daxilində ehtiyatlar tələb edən belə alqoritmlərin yaradılması qarşıya çıxan mühüm əhəmiyyətli məsələlərin realizasiyasını mümkün edə bilərdi. Ona görə də hesablama mürəkkəbliyi məsələləri xüsusi tədqiqat sahəsi kimi riyaziyyatçıları çox düşündürürdü.

Məlumdur ki, kompüterdə realizə olunan istənilən bir əməliyyatın əsas göstəricisi onun hesablama mürəkkəbliyidir. Hesablama mürəkkəbliyinə görə hesab əməlləri içərisində vurma əməli daha mürəkkəbdir. Bu mürəkkəblik onunla bağlıdır ki, istənilən çoxrəqəmli ədədlərin vurulması zamanı vurma əmələri ilə bərabər ən azı bir toplama əməli də yerinə yetirilməlidir. Bununla belə bu əmələn mürəkkəbliyinin qiymətləndirilməsi zamanı yerinə yetirilən vurma əməllərinin sayı əsas götürülür. Vurma əməlinin hesablama mürəkkəbliyinin qiymətləndirilməsi üçün ənənəvi olaraq elementar əməliyyatların miqdarı istifadə olunur. Elementar və ya bit əməliyyatı dedikdə iki bitin toplanması, çıxılması və vurulması əməliyyatları nəzərdə tutulur. İxtiyari iki n rəqəmli ədədin hasilinin hesablanması üçün kifayət edən bit əməliyyatlarının miqdarının tapılması problemi və ya $M(n)$ mürəkkəblik funksiyasının limitinin $(n \to \infty)$ hesablanması problemi hal-hazırda sürətli hesablama nəzəriyyəsinin qeyri-aşkar problemidir.

Məlum olduğu kimi ən sadə vurma alqoritmi ibtidai məktəbdən hər kəsə tanış olan "sütunlu vurma"dır. İki ədədi bir-birinə vurmaq üçün birinci ədədin hər bir rəqəmini ikinci ədədin hər bir rəqəminə vurmaq və sonra müəyyən sayda

toplama əməli yerinə yetirmək və nəticələri düzgün ardıcıllıqda yerləşdirmək lazımdır. Bu zaman əsas iş rəqəmlərin vurulmasıdır. Əgər ədədlər üç rəqəmlidirsə, onda vurma cədvəlinə 9 dəfə, beş rəqəmli olduqda isə 25 dəfə istinad edilməlidir. Ümumiyyətlə, bu cür prosedurların sayı ədədin rəqəmlərinin sayına mütənasib olaraq artır. Başqa sözlə desək, "sütunlu vurma"da ədədlər böyüdükcə, hesablama alqoritminin addımlarının sayı da sürətlə artır. Bunu aşağıdakı kimi izah edək. Vurma cədvəlini bilməyi tələb edən birrəqəmli ədədlərin vurulmasını ən sadə vurma hesab edək və onun mürəkkəbliyini 1 ədədi ilə işarə edək. İndi n rəqəmli ədədin birrəqəmli ədədə vurulmasına baxaq. Bu zaman n dəfə bir rəqəmli ədədləri vurmaq lazımdır. Toplamanın mürəkkəbliyinin vurmaya nəzərən azlığına görə nəticəni almaq üçün lazım olan toplama əməllərinin sayı nəzərə alınmır. Deməli n rəqəmli ədədin bir rəqəmli ədədə vurulması birrəqəmli ədədlərin vurulmasından n dəfə mürəkkəbdir, yəni belə vurmanın mürəkkəbliyi $n * 1 = n$-dir. İndi n rəqəmli ədədin n rəqəmli ədədə vurulmasına baxaq. Əgər sütunlu vurma halına baxsaq bu əvvəlkindən n dəfə mürəkkəbdir, yəni onu n dəfə təkrar etmək lazımdır. Nəticədə mürəkkəblik $n * n = n^2$ olar. Ona görə də bu "məktəb" üsulunun mürəkkəbliyi üçün yuxarıdan qiymətləndirmə olaraq $M(n) = O(n^2)$ münasibəti alınır. Başqa sözlə desək, n rəqəmli iki ədədin hasilinin hesablanması n^2 elementar əməliyyatlar tələb edir. Misal üçün, 4 rəqəmli iki ədədin vurulması üçün 16 sayda elementar vurma əməlindən başqa xeyli sayda aralıq nəticələrin toplanması və axırdan sıfırların əlavə edilməsi kimi əməliyyatların da yerinə yetirilməsinə baxmayaraq bu əməliyyatın mürəkkəbliyi 16 tərtibində olacaqdır. Akademik A.N.Kolmoqorov öz mülahizələrində istənilən vurma üsulunda M(n) mürəkkəblik funksiyasının aşağıdan qiymətləndirilməsi üçün də eyni tərtibli münasibətin doğru olduğunu fərz edirdi. Başqa sözlə desək, akademik A.N.Kolmoqorova görə elə bir ümumiləşmiş vurma üsulu ola bilməzdi ki, bu üsul iki n rəqəmli ədədin hasilini n^2-dan az sayda elementar əməliyyatla yerinə yetirsin. Belə bir fərziyyənin həqiqətə oxşarlığı o faktdan qaynalanırdı ki, "sütunlü vurma" üsulu 4000 ildən çox bir vaxt ərzində mövcud olmuşdur (misal üçün bu üsuldan bəşəriyyət tarixinin ilk sivilizasiyalarından biri olan şumerlər istifadə etmişlər) və əgər daha sürətli bir üsul olsaydı çox ehtimal ki, o artıq tapılmalı idi. Lakin 1960-cı ildə Anatoli Karatsuba adlı rus riyaziyyatçısı ixtiyari iki n rəqəmli ədədin vurulması üçün polinomial mürəkkəbliyə malik, yəni mürəkkəblik funksiyası $M(n) = O\left(n^{\log_2 3}\right)$ olan bir vurma alqoritmi kəşf etmişdir [5]. Bununla da "n^2" fərziyyəsini inkar etmişdir. Karatsubanın bu elmi işi istər keçmiş sovet məkanında, istərsə də qərb dünyasında sürətli alqoritmlər sahəsində tədqiqatların başlanğıcı oldu. Sonralar Karatsuba üsulu "parçala hökm sür" paradiqmasınadək ümumiləşdirilmiş və onun təsiri altında ikili axtarış, parçanın yarıya bölünməsi və digər bu kimi hesablama üsulları meydana çıxmışdır. Karatsuba üsulunun mahiyyətininə baxaq.

Tutaq ki, $2n$ uzunluqlu X və Y ədədlərinin hasilinin tapılmasına baxırıq

(tək uzunluqlu ədədlər halında soldan sıfırlar əlavə edilir). "Sütunlu vurma" halında mürəkkəblik $2n * 2n = 4n^2$ olar. Bu ədədlərin hər birini yarıya bölək:

$$X = a_1a_2\ldots..a_na_{n+1}\ldots a_{2n-1}a_{2n} \quad Y = b_1b_2\ldots b_{n-1}b_n\, b_{n+1}\ldots b_{2n-1}b_{2n}.$$

$$X = x_1\, x_2 \quad Y = y_1\, y_2.$$

$$x_1 = a_1a_2\ldots..a_n, \qquad x_2 = a_{n+1}\ldots a_{2n-1}a_{2n}\ ,$$

$$y_1 = b_1b_2\ldots b_{n-1}b_n, \qquad y_2 = b_{n+1}\ldots b_{2n-1}b_{2n}.$$

Onda aşağıdakıları yaza bilərik:

$$X = x_1 \cdot 10^n + x_2,\ Y = y_1 \cdot 10^n + y_2.$$

$$X \cdot Y = (x_1 \cdot 10^n + x_2)(y_1 \cdot 10^n + y_2)$$

$$= x_1 \cdot 10^{2n} \cdot y_1 + (x_1 \cdot y_2 + x_2y_1) \cdot 10^n + x_2y_2. \quad (1)$$

Onu da qeyd edək ki, ədədi 10-un qüvvətlərinə vurmaq ədədə sağ tərəfdən qüvvət üstü qədər sıfırların yazılmasından ibarət sadə bir əməliyyatdır və mürəkkəbliyin hesablanmasında nəzərə alınmaya bilər.

Beləliklə, biz *n* rəqəmli iki ədədin hasilini hesablamaq üçün 4 vurma əməliyyatını yerinə yetirməliyik:

$$x_1 \cdot y_1\ , x_1 \cdot y_2\ , x_2 \cdot y_1\ , x_2 \cdot y_2.$$

Bu vurmaların hər biri n^2 mürəkkəbliyinə malikdir. Ona görə də toplam mürəkkəblik $4n^2$olar. Yəni biz nəticədə başqa yolla elə adi sütunlu vurmanın nəticəsini alırıq. Karatsuba vurmasında $x_1 \cdot y_2\ , x_2 \cdot y_1$vurmalarının əvəzinə aşağıdakı 3 vurma təklif olunur:

$$x_1.y_1\ , x_2 \cdot y_2, (x_1 + x_2) \cdot (y_1 + y_2).$$

Bəs onun köməyilə bizə lazım olan hasilləri necə ala bilərik?

Axırıncı hasildən x_1y_1 *və* $x_2y_2$hasillərini çıxaq:

$$(x_1 + x_2)(y_1 + y_2) - x_1y_1 - x_2y_2 = x_1y_2 + x_2y_1. \quad (2)$$

Göründüyü kimi (1) bərabərliyində tələb olunan $x_1y_2 + x_2y_1$ cəmi alındı. Əlavə olaraq biz yalnız toplama əməlindən istifadə etdik. Deməli $4n^2$ mürəkkəbliyi $3n^2$mürəkkəbliyinə qədər azaldılmış oldu.

Karatsubanın təklif etdiyi 4 hasilin hesablanmasını misal üçün 128 rəqəmli ədədlər halında belə izah etmək olar. 128 rəqəmli ədədlərin hər biri iki 64 rəqəmli ədədə bölünür. Alınmış 64 rəqəmli ədədlərində hər biri yarıya, yəni 32 rəqəmli ədədlərə bölünür. Eyni qayda ilə 32 rəqəmli ədədlər 16 rəqəmliyə, 16 rəqəmli 8 rəqəmliyə və s. bölünür. Rekursiv olaraq aparılan bu proses sonda 1 rəqəmli ədədlərin vurulmasında saxlanılır ki, bu mərhələdə artıq sadəcə toplama əməli icra olunur. Məhz bu qayda ilə n rəqəmli ədədlərin vurulmasının mürəkkəbliyi n^2-dan aşağı salına bilinir. Yuxarıda qeyd etdiyimiz kimi verilmiş ədədlərin yarıya bölünməsi zamanı tək rəqəmli ədədlər alınarsa, onda bu ədədlərin hər biri soldan eyni sayda sıfırlar əlavə edilməklə cüt rəqəmli ədədlərə çevrilir. Beləliklə rekursiv alqoritmdən ibarət Karatsuba vurmasında n rəqəmli

ədədlərin vurulmasının mürəkkəbliyi n^2-dan $n^{1,585}$ - dək azaldılmış olur. Daha dəqiq desək, mürəkkəblik $O(n^{\log_2 3})$ olur. Bu qiymətləndirmədə loqarifmanın arqumenti hər addımda vurmaların sayını bildirir. Xüsusi halda, yəni *n* rəqəmli ədədlərin sütunlu vurulmasında hər addımda 4 sayda vurma əməli icra olunduğuna görə mürəkkəblik funksiyası $O(n^{\log_2 4}) = O(n^2)$ olar, yəni göründüyü kimi adi sütunlu vurmadakı dəqiq nəticə alınır.

Aşağıdakı misallara baxaq.

Misal 1. Tutaq ki, $1234 \cdot 5678$ hasilini hesablamaq tələb olunur.

Həlli. Karatsuba üsuluna əsasən bu ədədlərin hər biri iki rəqəmli hissələrdən ibarət daha kiçik ədədlərə ayrılır və tələb olunan hasil simvolik olaraq $(12 + 34) \cdot (56 + 78)$ şəklində təsvir olunur. Alqoritmə əsasən aşağıdakı hasillər yerinə yetirilməlidir:

$$1)12 \cdot 56 \quad 2)\ 34 \cdot 78 \quad 3)\ 12 \cdot 78 \quad 4)\ 34 \cdot 56.$$

Əvvəlcə 1-ci və 2-ci hasillər hesablanır:

I addım. 1-ci hasili hesablayırıq: $12 \cdot 56 = 672$.

II addım. 2-ci hasili hesablayırıq: $34 \cdot 78 = 2652$.

III addım. Birinci hasilin axırına 4 ədəd sıfır əlavə edib alınmış ədədi II addımım nəticəsi ilə toplayırıq: 6720000+2652=6722652.

Tələb olunan 3-cü və 4-cü hasillərin cəmi aşağıdakı IV və V addımlarla hesablanılır:

IV addım. $(12 + 34) \cdot (56 + 78) = 46 \cdot 134 = 6164$.

V addım. IV addımın nəticəsindən I və II addımların nəticələri çıxılır:

6164-(672+2652)=2840.

VI addım. V addımın nəticəsinin axırına iki sıfır əlavə etməklə alınan ədədi III addımın nəticəsi ilə toplayırıq: 284000+6722652= 7006652.

Alınmış nəticə axtarılan hasildir.

Cavab: $1234 \cdot 5678 = 7006652$.

Baxılan bu misal halında verilmiş 4 rəqəmli ədədlərin vurulması adi məktəb vurmasında 16 elementar (rəqəmin rəqəmə) vurma əməlini, eləcə də aralıq ədədlərin sürüşdürlməsi və toplanması əməliyyatlarını tələb etdiyi halda Karatsuba üsulunda 12 elementar vurmadan istifadə edilmişdir. Bu zaman 12 vurmadan əlavə 5 toplama, 2 sıfırların əlavə edilməsi və 1 inversiya əməlindən də istifadə edilməsinə baxmayaraq, bu əməllərinin vurma əməlinə nisbətdə az mürəkkəb əməllər olduğunu nəzərə alsaq cüzi də olsa hesablama mürəkkəbliyinin azaldılması müşahidə olunur.

Digər bir misala baxaq.

Misal 2. Tutaq ki, 12345678 x 17654321 hasilini tapmaq tələb olunur.

Həlli. Əvvəlki misalda olduğu kimi 8 rəqəmli bu ədədlərin hər biri 5 rəqəmli hissələrə ayrılır və tələb olunan hasil $(1234 + 5678) \cdot (1765 + 4321)$ şəklində təsvir olunur. Birinci misalda olduğu kimi aşağıdakı kimi hesablamalar

aparılır:

$1)1234 \cdot 1765$ $2)\ 5678 \cdot 4321$ $3)\ 1234 \cdot 4321$ $4)5678 \cdot 1765.$

I addım. 1-ci hasil Misal 1-ə uyğun olaraq hesablanır:

$$1234 \cdot 1765 = 2178010.$$

II addım. 2-ci hasildə Misal 1-ə uyğun olaraq hesablanır:

$$5678 \cdot 4321 = 24534638.$$

III addım. I addımın nəticəsinin axırına 8 sıfır əlavə edib II hasilin nəticəsi ilə toplanılır:

217801000000000+24534638=217801024534638.

IVaddım. $(1234 + 5678) \cdot (1765 + 4321)$ hasili hesablanır:

$$(1234 + 5678) \cdot\ (1765 + 4321) = 42066432.$$

V addım. IV addımın nəticəsindən I və II addımların nəticələrinin cəmi çıxılır:

42066432-(2178010+24534638)=15353784.

VI addım. V addımım nəticəsinin axırına 4 sıfır əlavə edib alınan nəticəni III addımım nəticəsi ilə toplanılır:

153537840000+217801024534638=217954562374638.

Cavab: 12345678 x 17654321=217954562374638.

Misal 2 halında verilmiş 8 rəqəmli ədədlərin vurulması adi məktəb vurmasında 64 elementar (rəqəmin rəqəmə) vurma əməlini, eləcə də aralıq ədədlərin sürüşdürlməsi və toplanması əməliyyatlarını tələb etdiyi halda Karatsuba üsulunda 36 elementar vurmadan, 15 toplama, 6 sıfırların əlavə edilməsi və 3 inversiya əməlindən istifadə edilmişdir. Hesablamalardan göründüyü kimi ilk baxışda Karatsuba vurma üsulu adi məktəb vurma üsulundan mürəkkəb görünür. Məsələ ondadır ki, mütəxəssislərin qiymətləndirmələrinə görə Karatsuba vurma üsulunun səmərəsi ən azı 10000 rəqəmli natural ədədlər halında ortaya çıxır. Doğrudanda, N=10000 rəqəmli iki natural ədəd halında Karatsuba vurması 2187761 sayda elementar vurma ilə yerinə yetirildiyi halda, adi məktəb vurması 100 mln sayda (ola bilər 2 və ya 3 dəfə çox bu o qədər də əhəmiyyətli deyildir) elementar vurma tələb edir. Göründüyü kimi Karatsuba vurma alqoritmi adi məktəb vurma alqoritminə nisbətdə 2 tərtib daha səmərəlidir.

1960-cı ildən başlayaraq Karatsuba alqoritmindən də daha səmərəli, başqa sözlə desək, daha eleqant polinomial mürəkkəbliyə malik vurma alqoritmlərinin axtarışı istiqamətində aparılan tədqiqatlar nəticəsində on ildən sonra, yəni 1971-ci ildə Arnold Şönhare və Folker Ştrassen adlı iki alman riyaziyyatçısının təklif etdikləri vurma alqoritmi meydana çıxdı. Sonralar onların adı ilə adlandırılan Şönhare-Ştrassen alqoritmi N rəqəmli iki ədədin vurulması üçün $N \cdot \log_2 N \cdot \log_2(\log_2 N)$ - dan artıq addım tələb etmirdi [8]. Kifayət qədər böyük N ədədi üçün Şönhare-Ştrassen alqoritmi Karatsuba alqoritminə nisbətən daha böyük sürətli hesablamanı təmin edir. Belə ki, 1mln rəqəmli iki ədədin hasilini adi vurma üsulu 10^{12} addıma (elementar vurmaya) yerinə yetirdiyi halda, Karatsuba alqoritmi bu

hesablamanı 3 235 936 569 addıma (elementar vurmaya), Şönhare-Ştrassen alqoritmi isə bu hesablamanı 85287084 addıma (elementar vurmaya) yerinə yetirə bilir. Göründüyü kimi, Şönhare-Ştrassen alqoritmi adi məktəb vurmasına nisbətdə 4 tərtib, Karatsuba alqoritminə nisbətdə isə 2 tərtib daha səmərəlidir. Bir elementar vurma əməlinin, yəni iki bitin vurulması əməlinin kompüterdə yerinə yetirilməsi (7-10 takt) üçün təqribən 10^{-9} san zaman lazım olduğu fərziyyəsində belə hasilin hesablanmasına müasir kompüterə adi məktəb vurmasında 16,7 dəqiqə, Karatsuba üsulunda 3,23 san, Şenhare-Ştrassen alqoritmi ilə hesablamada isə 0,08 saniyə vaxt lazım gələr.

1971-2007-ci illərdə Şönhare-Strassen alqoritmi asimptotik sürətli alqoritm kimi qəbul edilmişdir. Bununla belə Şönhare-Strassen alqoritmini optimal hesab etməyən müəlliflər, N rəqəmli iki ədədin vurulması üçün addımlarının sayı $N * log_2 N$-dən çox olmayan alqoritmin mümkünlüyü fərziyyəsini irəli sürdülər. Bu fərziyyə ona əsaslanırdı ki, vurma əməli kimi fundamental bir əməl üçün məhdudiyyət $N \cdot \log_2 N \cdot \log_2(\log_2 N)$ hasilinə nisbətdə daha eleqant olmalıdır. Bu zaman isveçrəli riyaziyyatçı Martin Fürerin sözlərini xatırlamaq yerinə düşər. O demişdir: " Təcrübələrə əsasən biz bilirik ki, riyaziyyat elmində baza elementləri nəticə etibarilə həmişə eleqant olurlar. Bu mənada əksər alimlərin mövqeyinə görə vurma əməli də əhəmiyyətli baza əməliyyatı kimi sırf estetik nöqteyi-nəzərcə daha gözəl mürəkkəblik funksiyasına malik olmalıdır" [4]. Bu mülahizəyə "Gözəllik qaydası"nda hökm edilən "Bütün başqa parametrlər üzrə eyni olan iki modeldən ən gözəlini seçmək lazımdır" mülahizəsini də əlavə etsək doğrudanda $O(N \cdot \log_2 N \cdot \log_2(\log_2 N))$ mürəkkəbliyinə nisbətdə daha eleqant $O(N * log_2 N)$ mürəkkəbliyinə malik vurma alqoritminin varlığına inanmaq olar. Məhz bu fikirləri əsas tutan Martin Fürer 2007-ci ildə, yəni Şönhare-Ştrassen alqoritminin meydana çıxmasından 36 il sonra hesablama mürəkkəbliyi $O(N \cdot log_2 N \cdot 2^{O(log^* N)})$ olan alqoritm təklif etmişdir [4]. Burada $log^* N$ - N-in iterativ loqarifması başa düşülür:

$$log^* N = \begin{cases} 0, & əgər\ N \leq 1, \\ 1 + log^*(log_2 N), & əgər\ N > 1. \end{cases}$$

İnformatikada log N-in əsası olaraq 2 ədədi tətbiq edildiyi halda, $log^* N$-loqarifmasının əsası olaraq $b > e^{1/e}$ şərtini ödəyən ədədlər götürülür. Bu loqarifmanın maraqlı xüsusiyyəti ondadır ki, o artan, lakin zəif sürətlə artan bir hesablamadır. Belə ki, praktiki əhəmiyyətli bütün hesablamalarda bu loqarifmanın qiyməti demək olar ki, sabit olaraq qalır. Misal üçün, $N \in (65536, 2^{65536}]$ halında $log^* N \in (4; 5]$. Martin Fürerin alqoritminin Şönhare-Ştrassen alqoritmindən üstünlüyü 10 trilyondan artıq rəqəmli ədədlər halında aşkar görünür.

Martin Fürerin alqoritminin mürəkkəbliyi Şönhare-Ştrassen alqoritminin mürəkkəbliyi ilə fərz olunan $O(N \cdot \log_2 N)$ mürəkkəblik arasındakı boşluğu

doldursa da o vaxtdan, yəni 2007-ci ildən 2019-cu ilə qədər $O(N \cdot \log_2 N)$ mürəkkəblikli alqoritmin mövcudluğu fərziyyəsinin doğruluğunu göstərən heç bir konkret üsul kəşf edilməmiş və ya bu fərziyyənin isbatı mümkün olmamışdır. Məhz bu məsələni Devid Harvi və Yoris Huven adlı riyaziyyatçılar həll etdilər. Onlar addımlarının sayı $N \cdot log_2 N$-dən çox olmayan alqoritm təklif etdilər [3]. Sidneyin New South Wells Universitetinin əməkdaşı Devid Harvi və onun fransalı həmkarı Yoris Huven bu elmi işlərində yarım əsr bundan əvvəl riyaziyyatçıların qoyduğu problemi, yəni $\boldsymbol{O(N * log_2 N)}$ mürəkkəbliyə malik vurma alqoritminin varlığı problemini həll etmiş oldular. Bu problem Şönhare-Strassen fərziyyəsinin isbatından ibarətdir. Bu fərziyyəyə əsasən, N-rəqəmli ədədlərin vurulmasının elə bir alqoritmi mümkündür ki, ədəd böyüdükcə alqoritmin addımlarının sayı $\boldsymbol{N \cdot log_2 N}$-dan daha sürətli artmayacaqdır. Harvinin sözlərinə görə bir milyard onluq rəqəmli iki ədədin hasilini müasir kompüter bir aya, Şönhare-Strassen alqoritmindən istifadə ilə isə 130 saniyə ərzində hesablaya biləcəkləri halda onların təklif etdiyi alqoritm bu hesablamanı daha sürətli həll edə bilər. Harvi və Huven bu fərziyyəni qəti şəkildə sübut edə bilməsələr də təklif etdikləri alqoritmin sürətinin nəzəri cəhətdən mümkün həddi olduğunu düşünürlər. Əgər bu doğrudan da beladirsə, onda iki böyük ədədin hasilinin ən sürətli hesablanması üzrə riyazi problemi nəhayət ki, həll edilmiş hesab etmək olar.

Yuxarıda qeyd etdiyimiz kimi Karatsuba alqoritmi ən azı 10000 rəqəmli ədədlər (onluq) üçün səmərəli olduğu halda Harvi və Huvenin alqoritmi hansı uzunluqlu ədədlər üçün daha səmərəli ola bilər sualına Harvi səmimi olaraq "Heç nə deyə bilmərəm" cavabını vermişdir. Bununla belə Harvi öz elmi işində alqoritminin nəticəsi olaraq $10^{214857091104455251940635045059417341952}$ hasilinin alındığı vurma nümunəsini gətirir. Sual olunur: Fantastik böyüklükdə olan belə hasili səmərəli hesablaya bilən alqoritmin mövcudluğu bizə nələr vəd verə bilər?

Hər şeydən əvvəl qeyd edək ki, vurma əməli bölmə və kökalma tipli hesablama proseduralarının ayrılmaz bir tərkib hissəsidir. Başqa sözlə desək, kifayət qədər böyük ədədlərin bölünməsi və ya müxtəlif dərəcələrdən kökünün alınması məsələlərinin həlli vurma əməli vasitəsilə daha səmərəli mümkün olur. Bu mənada Harvi və Huvenin bu elmi işinin nəticəsi $\sqrt[n]{2}$, $\sqrt[n]{3}$ tipli irrasional ədədlərin, eləcə də müxtəlif elmi-texniki məsələlərin həllində istifadə olunan $\boldsymbol{\pi}$ və $\boldsymbol{e}$ ədədlərinin də daha çox sayda qiymətli rəqəmlərinin səmərəli şəkildə tapılmasına imkan verir. Məsələ ondadır ki, elə ədədlərdən, daha dəqiq desək onların kifayət qədər böyük sayda qiymətli rəqəmlərindən müxtəlif kriptoqrafik alqoritmlərdə geniş istifadə edilir. Misal üçün, kriptoqrafik heş-funksiyalardan olan SHA-1, SHA-2 (*Secure Hash Algorithm*) və bu ailədən olan digər alqoritmlərdə tətbiq olunan sabitlərin təyinində ilk 8 sadə ədədin kvadrat kökünün kəsr hissəsindən istifadə olunur. Bu irrasional ədədlər həmçinin psevdotəsadüfi ədədlər generasiya edən müxtəlif generatorların start qiymətlərinin (*seed value*) seçimində də geniş

istifadə edilir.

Digər tərəfdən Harvi və Huvenin bu elmi işi bu vaxtadək elmə məlum olan **sadə ədədlər** siyahısının genişləndirilməsi və faktorizasiya məsələlərinin həlli üçün də çox faydalı olacaqdır. Belə ki, informasiya mühafizəsinin istər simmetrik, istərsə də asimmetrik kriptosistemlərində açar elementlərinin təyinində kifayət qədər böyük, məlum cədvəllərə daxil olmayan və bir sıra ciddi şərtləri ödəyən sadə ədədlərdən istifadə olunur. Misal üçün, konfidensial informasiyanın mübadiləsi prosesində daha çox tətbiq olunan və faktorizasiya məsələsinin (verilmiş kifayət qədər böyük ədədin sadə vuruqlarına ayrılması məsələsi) çətinliyinə əsaslanan **RSA** (***R**ivest, **S**hamir, **A**dleman*) tipli müasir asimmetrik kriptosistemlərdə şifrləmə alqoritmini realizə edən f funksiyası belə təsvir oluna bilər:

$$f: x \to x^e \ (mod\ N).$$

Burada *N* (modul) və *e* müəyyən natural ədədlərdir.

Əks proses, yəni deşifrləmə prosesi isə *y=f*(*x*)-a görə *x*-ın tapılması

$$x \equiv y^d \ (mod\ N)$$

müqayisəsinin həlli kimi aparılır. Nəzərə alaq ki, bu proseslərdə istifadə edilən *N* modulu kifayət qədər iki böyük ***p*** və ***q*** sadə ədədlərinin (300 və daha çox rəqəmli) hasilindən ibarət olan ədəd, uyğun olaraq açıq və məxfi açar adlandırılan ***e*** və ***d*** parametrləri isə kifayət qədər böyük natural ədədlərdir. Misal üçün, müasir smartfonlardakı açıq açarlı autentifikasiya vasitələrinin istifadəçiləri üçün olduqca əhəmiyyətli olan 1024 bitli RSA-1024 kriptosistemi 309 rəqəmli natural ədədlərdən istifadə edir. Digər kriptosistemlərdə, məsələn elliptik əyri üzərində diskret loqarifm məsələsinin (ECDLP -*Elliptic Curve Discrete Logarithm Problem*) həllinin çətinliyinə əsaslanan kriptosistemlərdə standart parametrlərlə iş zamanı qarşıya çıxan birinci məsələ bu parametrlərdə səhvlərin olub-olmamasının araşdırılması məsələsidir [7]. Bu məsələnin həlli isə qeyd etdiyimiz alqoritmlərdə istifadə olunan sadə ədədlərin tələb olunan ağır şərtləri ödəyib ödəməməsinin yoxlanmasını tələb edir. Bu sadə ədədlər nə qədər unikal xüsusiyyətlərə malik olarsa kriptoqrafik sistemlərin dözümlülüyü də bir o qədər yüksək olar. Kriptoqrafların gəldiyi qənaətə görə asimmetrik kriptosistem istifadəçilərinin əksəriyyəti əvvəlcədən xüsusi yolla konstruksiyalanmış sadə ədədlərdən istifadəyə üstünlük verdikləri üçün kriptosistemdə boşluqların yaranma ehtimalı böyükdür. Belə boşluqlar isə digər məxfi elementlərin aşkarlanmasına imkan verməklə son nəticədə trafikin deşifrə edilməsi ilə nəticələnə bilər. Bəs məxfiliyin əsas parametrləri olan belə p və q böyük sadə ədədləri haradan götürülməlidir? Məhz bu məsələnin həllində Harvi və Hüvenin elmi işinin böyük faydası ola bilər. Belə ki, böyük sadə ədədlərin axtarışı daha dəqiq desək, bu ədədlərin generasiyası kifayət qədər böyük uzunluqlu ədədlərin generasiyası və onların sadəlik testlərindən keçirilməsi proseduralarının yerinə yetirilməsini tələb edir ki, bu məsələlərin də həlli sürətli hesablama alqoritmləri olmadan demək olar ki, qeyri-mümkündür. RSA kriptosistemində şifrləmə və deşifrləmə proseduralarından göründüyü kimi,

böyük ədədlər üzərində qüvvətə yüksəltmə əməlinin icrası bütövlükdə kriptosistemin sürətini azaldan amil kimi Harvi və Hüvenin elmi nəticəsi sayəsində minimallaşdırılma imkanı əldə edə bilər. Bu nailiyyətlər isə öz növbəsində daha mükəmməl simmetrik şifrləmə alqoritmlərinin, bütövlükdə son nəticədə kriptoqrafiya dünyasında daha müasir kriptoqrafik mühafizə üsullarının işlənilməsinə gətirib çıxara bilər.

Digər bir məsələ. Kriptoanalitik tədqiqatlarda $\boldsymbol{d}$ məxfi açarının təyini N modulunun faktorizasiyası məsələsinin həllini tələb edir. Məsələ ondadır ki, faktorizasiya məsələsi o qədər də asan həll edilən məsələ deyildir. Misal üçün, yuxarıda qeyd etdiyimiz RSA-1024 kriptosistemində istifadə olunan 309 rəqəmli N modulunun faktorizasiyası hələ ki (2021), mümkün olmamışdır. Bu onunla bağlıdır ki, hələ ki, faktorizasiyanın polinomial sürətli alqoritmi elm aləmində məlum deyildir. Elə bu səbəbdən də müasir ədədlər nəzəriyyəsinin ən əhəmiyyətli açıq problemlərindən birini klassik kompüterlərdə polinomial mürəkkəbliyə malik faktorizasiya alqoritminin varlığı məsələsi təşkil edir. Qeyd edək ki, hazırkı dövrədək (2021) vuruqlarına ayrılmış ən böyük natural ədəd 250 rəqəmli və ya 829 bitli natural ədədlərdir. Daha dözümlü müasir asimmetrik kriptosistemlərdə iki sadə ədədin hasilindən ibarət ən azı 2048 bitli (610 rəqəmli) natural ədədlər tələb olunur ki, belə böyük ədədlərin də generasiyası xüsusi sadəlik testlərinin tətbiqini zəruri edir və bu zaman sürətli hesablama alqoritmləri olmadan lazımı nəticə əldə etmək mümkün deyildir. Ona görə də belə sadə ədədlərin axtarışı və faktorizasiya məsələsinin həlli informasiyanın kriptoqrafik mühafizəsində olduqca aktualdır.

Harvi və Huvenin bu elmi işinin başqa bir maraqlı nəticəsi də var. Müasir dövrdə informasiyanın kriptoqrafik mühafizəsinə ən böyük təhdid hesab olunan kvant kompüterləri asimmetrik kriptosistemlərin əsaslandığı faktorizasiya və diskret loqarifmləmə məsələlərini klassik kompüterlərə nisbətdə çox asanlıqla həll edə bilir. Məsələ ondadır ki, kvant kompüterləri kifayət qədər böyük uzunluğa malik natural ədədlərin faktorizasiya məsələsini "Şor alqoritmi" əsasında (alqoritmlər nəzəriyyəsi və kvant informatikası üzrə elmi işləri ilə məşhur olan amerikalı riyaziyyatçı Piter Şorun şərəfinə adlandırılan) çox asanlıqla həll etdiyi halda, ən sürətli klassik alqoritmlərin realizə olunduğu klassik kompüterlər bu məsələni olduqca böyük zamanda həll edə bilir [5]. Belə ki, hazırkı dövrdə faktorizasiya üçün elmə məlum olan ən sürətli üsul olan *ümumiləşmiş ədədi meydanda xəlbir üsulu (GNFS*-General Number Field Sieve) $\approx \exp\left(\left(\frac{64}{9}\right)^{\frac{1}{3}} n^{\frac{1}{3}} (\ln(n))^{\frac{2}{3}}\right)$ ($n = k \cdot log_2\,(10) -$ *verilmiş ədədin 2-lik vahidlərinin sayı, k isə bu ədədi təyin edən onluq rəqəmlərin sayıdır*) eksponensial mürəkkəbliyə malik üsul olub 500 rəqəmli natural ədədi 10^{15} $əməl/san$ sürətli super kompüterdə 5 mlrd ilə vuruqlarına ayırmağa imkan verdiyi halda, eyni məsələ $\approx n^2 \ln(n) \ln(\ln(n))$ polinomial mürəkkəbliyə malik Şor alqoritminin köməyilə 1 mln. əməl/san sürətli kvant kompüterində 18 saniyəyə həll edilə bilər

[2]. Bu zaman nəzərdə tutulur ki, klassik alqoritmlərin hesablama imkanları sabit bir qiyməti (yuxarı sərhəddi) aşmır. Yuxarıda adları çəkilən avstraliyalı və fransız riyaziyyatçılarının bu elmi işi vəziyyətin heç də belə olmadığı qənaətinə gəlməyə əsas verir. Bəlkə də gələcəyin riyaziyyatçıları verilmiş ədədin vuruqlarına ayrılması üçün bu gün mümkün hesab olunmayan daha sürətli vurma əməliyyatını yerinə yetirə bilən elə klassik alqoritm təklif edəcəklər ki, mövcud asimmetrik kriptosistemlər nəinki kvant, eləcə də klassik kompüterlərdə də asanlıqla sındırılacaqdır. Nə qədər ki, kvant və ya klassik alqoritmlərin hansının həqiqətən nəzəri olaraq daha optimal olduğu sübut edilməmişdir bu alqoritmlərin müqayisəsi doğru olmazdı. Başqa sözlə desək, yaxın vaxtlarda həm klassik, həm də kvant kompüterlərində realizə oluna biləcək kvant-klassik alqoritmlərin yaradılmasını istisna etmək hələ tezdir. Bu məsələ, yəni milyardlarla və daha artıq rəqəmli onluq ədədlərin sürətli vurulmasının müasir kvant-klassik alqoritmlərinin işlənilməsi məsələsi hazırda riyaziyyatçılar və proqramçılar üçün tədqiqat obyekti olaraq qalmaqdadır. Ona görə də klassik vurma alqoritmlərinin imkanlarının tükəndiyini fərz etmək və Şor alqoritmi tipli kvant alqoritmlərinin axtarışını daha səmərəli hesab etmək düzgün olmazdı.

ƏDƏBİYYAT

1. *Ю.И.Богданов, А.А.Кокин и др.* Квантовая механика и развитие информационных технологий. Ж-л Информационных технологии и вычислительные системы. 2012, выпуск 1, 17-31.
2. *К.А.Валиев.* Квантовые компьютеры и квантовые вычисления. Ж-л Успехи физических наук,2005,т.175, №1.
3. *David Harvey, Joris Van Der Hoeven.* İnteger multiplication in time O(nlog n). 2019. https://hal.archives-ouvertes.fr/hal-02070778
4. *Fürer M.* Faster integer Multiplication. İn Proceedings of the thirty-ninth annual ACM symposium on Theory of computing, June 11-13, 2007, p.57-66, San Diego, California.
5. *Карацуба А., Офман Ю.* Умножение многозначных чисел на автоматах // Доклады Академии Наук СССР. — 1962. — Т. 145, № 2.
6. *Кевин Хартнет.* Математики обнаружили идеальный способ перемножения чисел. https://habr.com/ru/post/451860.
7. *Б.Я.Рябко., А.Н.Фионов.* Криптографические методы защиты информации. Уч.пособие.М., 2005, 229 с.
8. *Schönhage A., Strassen V.* Schnelle Multiplikation großer Zahlen // Computing. — 1971. — № 7. — P. 281—292.

Redaksiyaya daxil olub 17.11.2021