J. Appl. Math. & Informatics Vol. **26***(2008), No. 1 - 2, pp. 307 - 316*
Website: http://www.kcam.biz

A COMPUTATIONAL EXPLORATION OF THE CHINESE REMAINDER THEOREM

AMOS O. OLAGUNJU

ABSTRACT. Real life problems can be expressed as a congruence modulus n and split into a system of congruence equations in modulus factors of n. A system of congruence equations can be combined into a congruence equation under certain conditions. This paper uniquely presents and critically reviews the generalized Chinese Remainder Theorem (CRT) for combining systems of congruence equations into single congruence equations. Sequential and parallel implementation strategies of the generic CRT are outlined. A variety of unique applications of the CRT are discussed.

AMS Mathematics Subject Classification : 11Y02, 11Y40, 68W02, 68W10
Key words and phrases : Chinese Remainder Theorem; parallel algorithm; cryptography.

1. Introduction

The Chinese Remainder Theorem (CRT) is a valuable mathematical algorithm invented in ancient China. The first problem related to the CRT appeared in the classic book, "Sun Tzu Suan Ching" or "Master Sun's Mathematical Manual" of Sun Zi. There is no consensus about the publication date of the problem and the book. There are experts who allege that Sun Zi constructed the problem in the 4th century [6]. Other experts argue that the book, "Sunzi Suanjing" was published between 280 A.D and 473 A.D because the Chinese tax by family unit "hu diao" expressed in terms of silk floss "mian" was established in 280A.D, and the measuring scale between "chi" and "duan" was modified in 473 A.D [5], and Sun Zi used old Chinese scale [2, 3]. The book, "Sunzi Suanjing" introduces the basic mathematical tables and operations. In particular, the book provides the computation process with rod numerals and the famous Chinese remainder problem.

Received September 20, 2006.

The first published CRT problem by Sun Zi focused on the astonishing question: what number can be divided separately by 3, 5 and 7 to obtain 2, 3, and 2 respectively as the remainders? Sun Zi obtained the solution to this problem by computing the (a) multiples 70, 21, and 15 of (5 x 7), (3 x 7) and (3 x 5) respectively, (b) sum of (2 x 70, 3 x 21, 2 x 15) equals 233, and (c) smallest positive number by eliminating multiples of (3 x 5 x 7) from 233 to obtain 23. Let "mod" be the operator that produces the remainder of a division between two integers, and "=" be the congruence operator such that, y = n mod m implies y mod m = n mod m. That is, y is congruent to n modulus m implies that both y and n produce the same integer remainder when each is divided by m. Thus, the properties implicit in Sun Zi's solution can be presented as:

70 = 1 mod 3 = 0 mod 5 = 0 mod 7,
21 = 1 mod 5 = 0 mod 3 = 0 mod 7,
15 = 1 mod 7 = 0 mod 3 = 0 mod 5, and
233 = 2 x 70 + 3 x 21 + 2 x 15 satisfies each of the congruence equations. Moreover, any multiple of 105 = 3 x 5 x 7 is divisible by 3, 5 and 7. Consequently, 210 = 2 x 105 is subtracted from 233 to obtain 23 as the smallest positive integer.

The dramatic story from Brahma-Sphuta-Siddhanta created by Brahmagupta [4] is an astounding puzzle allied with the CRT. A horse stepped on and broke a basket full of eggs at a market. The horse rider offered to compensate the old woman and asked her for the number of eggs. Lacking memory, she only remembered that, the time she obtained two eggs out at a time from the basket, one egg was left; she recalled the same one egg was left when she removed three, four, five and six eggs at a time; however the basket was empty when she picked out seven eggs each time. What is the smallest possible number of eggs she had? The 301 eggs can be derived from the system of congruence equations y = 1 mod 2 = 1 mod 3 = 1 mod 4 = 1 mod 5 = 1 mod 6 = 0 mod 7.

The question naturally arises on how to solve the CRT problems using modern mathematics notation. Consider a number of jolly friends who plan to celebrate next Christmas at Libby Hill Seafood restaurant. Guests at the restaurant are seated by a waitress in rows of chairs with diner tables of various sizes. When the waitress decides to sit three to a row, one friend is left over; when she decides to sit five to a row, two friends are left over; when she decides to sit seven or eleven to a row, three friends are left over. How many jolly friends are in the party? Suppose y is the number of jolly friends in the party. The problem of determining the number of jolly friends can be expressed by the following four congruence equations.

$$y = 1 \mod 3 \tag{1}$$

$$y = 2 \mod 5 \tag{2}$$

$$y = 3 \mod 7 \tag{3}$$

$$y = 3 \mod 11 \tag{4}$$

Clearly, the congruence equations (1), (2), (3) and (4) must be solved simultaneously to obtain the single value that satisfies each equation. The greatest common divisor (GCD) of 3 and 5 is 1; thus, 3 and 5 are relatively prime numbers. Moreover, by definition of modulus, numbers congruent to 1 mod 3 are of the form y = 1 + 3k, for integer values k. Substituting this expression into equation (2) gives:

1 + 3k = 2 mod 5, 3k = 1 mod 5, k = (3-1 mod 5) mod 5, and k = 2.

y = 1 + 3k = 7, and the solution to equations (1) and (2) is y = 7 mod 15. Again, numbers congruent to 7 mod 15 are of the form y = 7 + 15n, for integers n. Substituting this expression into equation (3) gives:

7 + 15n = 3 mod 7, 15n = -4 mod 7 = 3 mod 7, n = 3(15-1 mod 7) mod 7 = 3(1) mod 7, and n = 3. y = 7 + 15n = 52, and the solution to equations (1), (2) and (3) is y = 52 mod 105. Again, numbers congruent to 52 mod 105 are of the form y = 52 + 105m, for integer values m. Substituting this expression into equation (4) gives:

52 + 105m = 3 mod 11, 105m = -49 mod 11 = 6 mod 11, m = 6(105-1 mod 11) mod 11 = 6 (2) mod 11, and m =1. y = 52 + 105m = 157, and the solution to equations (1), (2), (3) and (4) is y = 157 mod 1155. Note that 157 = 1 mod 3, 157 = 2 mod 5, 157 = 3 mod 7, and 157 = 3 mod 11. Thus, the smallest number of jolly friends in the party is 157.

2. The Chinese remainder theorem

Theorem 1. *Let m_1, m_2 be relatively prime integers. Given integersa_1, a_2 then $y \equiv c \, mod \, m_1m_2$ is the single solution to the simultaneous congruence equations $y \equiv a_1 \, mod \, m_1$ and $y \equiv a_2 \, mod \, m_2$, where $c = a_1 + m_1(m_1^{-1}$ $mod \; m_2)(a_2 - a_1)$.*

Proof. By definition, m_1 *and* m_2 are relatively prime implies$GCD(m_1, m_2) = 1$, and by the extended Euclidean algorithm there exists integers p, q such that $m_1p + m_2q = 1$. Then $m_1p = 1 \, mod \, m_2$, and $m_2q = 1 \, mod \, m_1$. Let $y = a_2m_1p + a_1m_2q$.

Then $y \equiv a_2m_1p \equiv a_2 mod \, m_1andy \equiv a_1m_2q \equiv a_1 \, mod \, m_2$, and so there is a solution. Suppose z is another solution; then$y \equiv z \, mod \, m_1$ and$y \equiv z \, mod \, m_2$, so that y-z is a multiple of both m_1 *and* m_2. Let w = y-z. By definition w is a multiple of both m_1 *and* m_2 implies there are integers s, t such that$w = m_1s = m_2t$.

Multiply $m_1p + m_2q = 1$ by w to obtain

$w = wm_1p + wm_2q = (m_2t)m_1p + (m_1s)m_2q = m_1m_2(tp + sq)$. Therefore, w is a multiple of m_1m_2 and$y \equiv z \, mod \, m_1m_2$, and any two solutions y to the system of congruence equations are congruent mod m_1m_2. Consequently,$y \equiv c \, mod \, m_1m_2$, and $c = a_1 + m_1(m_1^{-1} \, mod \, m_2)(a_2 - a_1)$ is the single solution to $y \equiv a_1 \, mod \, m_1$ and $y \equiv a_2 \, mod \, m_2$. □

Theorem 2. *Let $m_1, m_2, \ldots, m_n$be relatively prime integers. Given integers $a_1, a_2, \ldots, a_n$ then the congruence equations$y \equiv a_i \, (mod \, m_i)$,$1 \leq i \leq n$, has exactly one solution $y \equiv d \, (mod \, m_1m_2 \ldots m_n)$, where*

$d = p_n$, $p_1 = a_1$, $p_j = p_{j-1} + \prod m_h((\prod m_h)^{-1} \bmod m_j)(a_j - p_{j-1})$, $j =$ 1 to n, and the product $\prod m_h$ is computed from 1 to j -1.

Proof. Let CR(N) denote the statement $y \equiv a_1 \,(mod\, m_1) \equiv a_2 \,(mod\, m_2) \equiv \ldots \equiv a_n \,(mod\, m_n)$ has exactly one solution $y \equiv d\,(mod\, m_1 m_2 \ldots m_n)$.We need to show that CR(N) is true for all $N \geq 1$.

N=1, to prove CR(1) : i.e., $y \equiv a_1 \,(mod\, m_1)$ has a single solution. Clearly, $y\,(mod\, m_1) = a_1\,(mod\, m_1)$, and either$y = a_1$, *or* $y = a_1 + m_1 t$ for integers t that produce the same single solution in modulus m1, so the statement is true.

N=2, to prove CR(2): i.e., $y \equiv a_1 \,mod\, m_1 \equiv a_2 \,mod\, m_2$ has exactly one solution $y \equiv d\,(mod\, m_1 m_2)$. The proof is provided in Theorem 1, and the statement is true.

$k \geq 1$ and CR(k) the induction hypothesis is $y \equiv a_1 \,(mod\, m_1) \equiv a_2 \,(mod\, m_2) \equiv \ldots \equiv a_k \,(mod\, m_k)$ has exactly one solution $y \equiv d\,(mod\, m_1 m_2 \ldots m_k)$, where $d = p_k$, $p_1 = a_1$,

$p_j = p_{j-1} + \prod m_h((\prod m_h)^{-1} \bmod m_j)(a_j - p_{j-1})$, $j = 1$ to k, and the product $\prod m_h$ is computed over h = 1 to j. For the induction step CR(k+1), $y \equiv a_1 \,(mod\, m_1) \equiv a_2 \,(mod\, m_2) \equiv \ldots \equiv a_k \,(mod\, m_k) \equiv a_{k+1} \,(mod\, m_{k+1})$ has exactly one solution $y \equiv d\,(mod\, m_1 m_2 \ldots m_k m_{k+1})$, where $d = p_{k+1}$, $p_1 = a_1$,

$p_j = p_{j-1} + \prod m_h((\prod m_h)^{-1} \,mod\, m_j)(a_j - p_{j-1})$, j = 1 to k+1, the product $\prod m_k$ is computed over h = 1 to j, and the proof stems from

$$y \equiv a_1(modm_1) \equiv a_2(modm_2) \equiv \ldots \equiv a_k(modm_k) \equiv a_{k+1}(modm_k + 1)$$

$$= (y \equiv a_1(modm_1) \equiv a_2(modm_2) \equiv \ldots \equiv a_k(modm_k)) \equiv a_{k+1}(modm_{k+1})$$
$$= y \equiv d(modm_1 m_2 \ldots m_k) \equiv a_{k+1}(modm_{k+1}) \text{ (by induction hypothesis)}$$
$$= y \equiv d(modm_1 m_2 \ldots m_k m_{k+1}) \text{(by Theorem 1), where}$$
$$\text{d} = p_k + \prod m_h((\prod m_h)^{-1} modm_k + 1)(a_{k+1} - p_k), h = 1 \text{ to } k$$ □

3. Algorithms for Implementing the CRT

The CRT algorithm entails calculating the inverse of numbers in modulus arithmetic. Given congruence equations whose constant and modulus coefficients are stored in vectors A and M respectively, the following sequential algorithm generates the solution to these simultaneous equations by repeatedly invoking the function "MODINVERSE". The algorithm uses the temporary variables c, a1, a2, m1, m2, n and INV as counter, first number, second number, first modulus, second modulus, number of congruence equations, and the inverse of m1 and m2 respectively.

3.1. Sequential CRT algorithm

```
START
    Initialize c =1, a1 = A[c], m1 = M[c];
    WHILE (c < n) DO
      BEGIN c = c + 1;
```

```
                a2 = A[c];
                m2 = M[c];
               INV = MODINVERSE (m1, m2);
                a1 = a1 + (m1 * INV * (a2 -- a1));
                m1 = m1 * m2;
        END;
        a1 = a1 modulus m1;
        PRINT a1, m1;
STOP

MODINVERSE (m1, m2)
BEGIN
   Initialize t0 = 1, t1 = 0, m = m2;
   WHILE (m2 ? 0) DO
   BEGIN
     q = Floor (m1/m2);
    t2 = m1-(q*m2);
    t3 = t0 - (q*t1);
    t0 = t1;
    t1 = t3;
    m1 = m2;
    m2 = t2;
   END
  IF (t0 < 0) SET t0 = m + t0;
  RUTURN MODINVERSE=t0;
END
```

Consider the congruence equations y = 1 mod 3 = 1 mod 4 = 1 mod 5 = 0 mod 7. These equations are stored in vectors A and M, and here is the trace of execution of the algorithm.

C	A	M	a1	m1	a2	m2	INV
1	1	3					
2	1	4	1	3	1	4	3-1 mod 4 = 3
3	1	5	1	12	1	5	12-1 mod 5 = 3
4	0	7	1	60	0	7	60-1 mod 7 = 2

The final values are: a1 = 1 + 60(2(0-1)) = -119, m1 = 60(7) = 420, a1 mod m1 = -119 mod 420 = 301. The solution to the congruence equations is y = 301 mod 420.

The sequential CRT algorithm uses instructions in succession to manipulate the elements of vectors A and M, and to compute inverse values. Thus, the chronological CRT algorithm is inefficient for computing the solution to numerous systems of congruence equations. An efficient algorithm ought to split the elements of vectors A and M for processing by several processors. Moreover, an resourceful algorithm for computing the inverse in modulus arithmetic should take advantage of concurrent processors.

The parallel CRT algorithm below assumes the existence of multiple processors. The algorithm is suitable for even and odd numbers of congruence equations. The number of available processors will determine the partitioning of parallel computational tasks. In particular, given P processors and N congruence equations, it is important to explore the cases when both P and N are either odd or even, or one of P and N is odd. However, processors are usually assigned to computer program segments designed to execute concurrently based on availability. Thus, the algorithm below illustrates how the sequential CRT algorithm may be coded for parallel execution.

3.2. Parallel CRT algorithm

```
START
      IF (n mod 2 = 0) THEN
         SET s = n/2 +1; \{even n\}
      ELSE SET s = (n+1)/2; \{odd n\}
      PARBEGIN \{initialize variable\}
                  c1 =1; a11 = A[c1]; m11 = M[c1];
                  c2 = s; a21 = A[c2]; m21 = M[c2];
              PAREND;
     WHILE (c1 < s) DO
     BEGIN \{compute solution using parallel instructions\}
             PARBEGIN c1 = c1 + 1;
                      c2 = c2 + 1;
                   PAREND;
             PARBEGIN a12 = A[c1];  m12 = M[c1];
                      A22 = A[c2];  m22 = M[c2];
                    PAREND;
             PARBEGIN
                    INV1 = MODINVERSE (m11, m12);
                    INV2 = MODINVERSE (m21, m22);
                PAREND;
             PARBEGIN
                 a11 = a11 + (m11 * INV1 * (a12 -- a11));
                 a21 = a21 + (m21 * INV2 * (a22 -- a21));
                PAREND;
             PARBEGIN
                    m11 = m11 * m12;
                    m21 = m21 * m22;
                PAREND;
      END;
      INV = MODINVERSE (m11, m12);
      a11 = a11 + (m11*INV*(a22 --a11));
```

```
        m11 = m11*m21;
        IF (n mod 2 \ne 0) THEN
        BEGIN
             INV = MODINVERSE (m11, M[n]);
             a11 = a11 + (m11*INV*(A[n] -- a11));
             m11 = m11*M[n];
          END;
        a11 = a11 modulus m11;
        PRINT a11, m11;
STOP

MODINVERSE (m1, m2)
BEGIN \{compute the inverse of m1 mod m2 with parallel
        instructions\}
    PARBEGIN
           t0 = 1;
           t1 = 0;
            m = m2;
        PAREND;
        WHILE (m2 \ne 0) DO
        BEGIN
            q = Floor (m1/m2);
          PARBEGIN
              p1 = q*m2;
              p2 = q*t1;
            PAREND;
          PARBEGIN
               t2 = m1-(q*m2);
               t3 = t0 - (q*t1);
               t0 = t1;
             PAREND;
          PARBEGIN
                t1 = t3;
                m1 = m2;
                m2 = t2;
             PAREND;
         END
        IF (t0 $<$ 0) SET t0 = m + t0;
        RUTURN MODINVERSE=t0;
 END
```

The sequential and parallel CRT algorithms require relatively prime modulo numbers in vector M. When modulo numbers in equations are not relatively prime, the equations may be preprocessed without any loss of information. For instance, consider y = 1 mod 2 = 1 mod 3 = 1 mod 4 = 1 mod 5 = 1 mod 6 = 0 mod 7. Clearly, 2, 3, 4, 5, 6 and 7 are not relatively prime numbers. However, 3, 4, 5, and 7 are relatively prime numbers. The information in 1 mod 6 can be derived from 1 mod 3 and 1 mod 2, and the information in 1 mod 2 is covered in 1 mod 4. Consequently, z = 1 mod 3 = 1 mod 4 = 1 mod 5 = 0 mod 7 will generate the same solution as y. Accordingly congruence equations that contain no relatively prime modulo numbers require factoring and combining of equations.

4. Application of the CRT

The CRT is evolving in new contexts and types of applications notwithstanding its existence for two centuries. The usefulness and applications of the CRT are apparent in abundant aspects of algorithms and modular computations, particularly in the theory of codes and cryptography [1]. For example, the CRT can be used to solve y2 = 133 mod 143 by factoring 143 into a product of primes (11 x 13) and rewriting the congruence as two equations.

$$y2 = 133 \mod 11 = 1 \mod 11 \tag{5}$$

$$y2 = 133 \mod 13 = 3 \mod 13 \tag{6}$$

The solutions to equation (5) are y = 1 mod 11 and
y = -1 mod 11 = 10 mod 11.
The solutions to equation (6) are y = 4 mod 13 and
y = -4 mod 13 = 9 mod 13.
The CRT is then used to combine the pairs of congruence equations.
y = 1 mod 11 and y = 4 mod 13 to obtain y = 56 mod 143,
y = 1 mod 11 and y = 9 mod 13 to obtain y = 100 mod 143,
y = 10 mod 11 and y = 4 mod 13 to obtain y = 43 mod 143, and
y = 10 mod 11 and y = 9 mod 13 to obtain y = 87 mod 143.

Many real life problems may be formulated as systems of congruence equations for investigation via the CRT. The CRT may be creatively used to monitor inventory at several warehouses of a corporation. Suppose a company has four warehouses w, x, y, z with different cubit storage spaces. For simplicity, suppose merchandise items such as boxes of computers or cars of equal sizes are stored in each warehouse. The CRT is practical for solving the congruence equations below used to estimate the number of merchandise items, N, in the four warehouses.
N = r1 mod w,
N = r2 mod x,
N = r3 mod y, and
N = r4 mod z
where r1, r2, r3, and r4 are the respective numbers of unoccupied cubit storage spaces in warehouses w, x, y and z. For example, the congruence equations N = 0 mod 12 = 2 mod 13 = 3 mod 25 = 3 mod 37 are solved by the CRT algorithm to yield 97,544 merchandise items in the four warehouses.

The CRT may be applied to explore the best use of powder loads for bullets and artillery of specific gunpowder grains and shell casing sizes. In particular, consider a weaponry distributor of y rounds of 9mm shell casings, each with 50 grains capacity; z rounds of 7mm rifle casings, each able to hold 71 grains; and w rounds of 40mm grenade casings, each with 153 grains storage. The total gunpowder grains may be computed with the CRT algorithm to resolve the allocation of powder for producing the optimal number of ammunitions.

The CRT may be used in steganography to estimate the optimal amount of least significant bits data to conceal, or hidden data spread over several documents or pictures. This requires the knowledge of the total usable space in each document or picture. The document or picture size may be measured in kilobytes or megabytes. The maximum data sizes of documents or pictures and the concealed data sizes might subsequently be used in the CRT algorithm to derive the amount of stored data.

5. Conclusion

The generalized Chinese Remainder Theorem is exhibited, proved and critically reviewed. Sequential and parallel algorithms are put forward for implementing the CRT. Efficiency and effectiveness of the algorithms are discussed. The practical applications of the CRT are expounded. No attempt was made to present comprehensive applications of the CRT. Nonetheless, this paper may be useful for understanding the underpinning and applications of the CRT.

References

1. Ding, D. Pei and A. Salmoaa, *Chinese remainder theorem: applications in computing, coding, cryptography*, World Scientific Publishing Co. Inc., River Edge, 1996.
2. L. Y. Lam and T. S Ang, *Fleeting footsteps: tracing the conception of arithmetic and algebra in ancient placecountry-regionChina, revised edition*, World Scientific Publishing Co. Ltd., River Edge, NJ, 2004.
3. J. Needham, *Science and civilization in China, volume 3*, Cambridge University Press, Cambridge, UK, 1959.
4. O. Ore, *Number theory and its history,* Dover Publications, Inc. placeCityNew York, StateNY, 1976.
5. L. Wang, *The Chinese origin of the decimal place-value system in the notation of numbers and the possibility of its transmission to India*, In Proceedings of the International Congress of the History of Science, Adelaide, Australia, 20-27 August, 1958.
6. D. Wells, *The penguin book of curious and interesting puzzles*, Penguin Putnam Inc. placeCityNew York, StateNY, 1992.

Amos Olagunju is a Professor of Computer Networking at placePlaceNameSt Cloud PlaceTypeState PlaceTypeUniversity. He served as the Dean of the PlaceTypeSchool of PlaceNameGraduate Studies and Chief Research Officer at placePlaceNameWinston PlaceNameSalem PlaceTypeState PlaceTypeUniversity during the 2006-2007 academic years. He is a senior member of the ACM. He has conducted research activities for the Navy and Bell Communications Research. His current interests include formal proofs of security in protocols, and network performance evaluation.

Department of Statistics and Computer Networking, PlaceNameSt Cloud PlaceTypeState PlaceTypeUniversity, placeCitySt. Cloud PostalCode56301, country-regionUSA
e-mail: aoolagunju@stcloudstate.edu