Hindawi Publishing Corporation
International Journal of Mathematics and Mathematical Sciences
Volume 2012, Article ID 174372, 10 pages
doi:10.1155/2012/174372

Research Article

Properties of Carry Value Transformation

Suryakanta Pal,[1] Sudhakar Sahoo,[2] and Birendra Kumar Nayak[3]

[1] *Department of Basic Science and Humanities, Silicon Institute of Technology, Silicon Hills, Patia, Bhubaneswar 751024, India*
[2] *Department of Computer Science, Institute of Mathematics and Applications, Andharua, Bhubaneswar 751003, India*
[3] *P.G. Department of Mathematics, Utkal University, Bhubaneswar 751004, India*

Correspondence should be addressed to Sudhakar Sahoo, sudhakar.sahoo@gmail.com

Received 28 December 2011; Accepted 20 January 2012

Academic Editor: V. R. Khalilov

Carry Value Transformation (CVT) is a model of discrete deterministic dynamical system. In the present study, it has been proved that (1) the sum of any two nonnegative integers is the same as the sum of their CVT and XOR values. (2) the number of iterations leading to either CVT = 0 or XOR = 0 does not exceed the maximum of the lengths of the two addenda expressed as binary strings. A similar process of addition of modified Carry Value Transformation (MCVT) and XOR requires a maximum of two iterations for MCVT to be zero. (3) an equivalence relation is shown to exist on $Z \times Z$ which divides the CV table into disjoint equivalence classes.

1. Introduction

The notion of transformation is very important in mathematics. Accordingly, in the literature, one finds many kinds of transformations with interesting properties. Carry Value Transformations (CVTs) and Modified Carry Value Transformations (MCVTs) are two challenging transformations which currently have assumed much significance because of their applications in fractal formation [1], designing new hardware circuits for arithmetic operations [2], and so forth. Similar kind of transformations such as Extreme Value Transformation (EVT) [3], 2-Variable Boolean Operation (2-VBO) [4], Integral Value Transformation (IVT) [5] are also used to manipulate strings of bits and applicable in pattern formations [3, 4], solving Round Rabin Tournaments problem [6], Collatz-like functions [5], and so forth. All these applications in diversified domain motivated us to study the mathematical properties of these kinds of transformations.

The hardware circuit for arithmetic operations as designed in [2] is based on a result that after finite number of iterations, either CVT of the two nonnegative integers is equal

$$\begin{array}{rlll} \text{carry value} = a_n \wedge b_n & a_{n-1} \wedge b_{n-1} \ldots\ldots\ldots\ldots\ldots & a_1 \wedge b_1 & 0 \\ a = & a_n & a_{n-1} \ldots\ldots\ldots\ldots\ldots\ldots & a_1 \\ b = & b_n & b_{n-1} \ldots\ldots\ldots\ldots\ldots\ldots & b_1 \\ \hline a \oplus b = & a_n \oplus b_n & a_{n-1} \oplus b_{n-1} \ldots\ldots\ldots & a_1 \oplus b_1 \end{array}$$

Figure 1: Carry generated in ith column counted from LSB is saved in $(i+1)$th column.

Carry:	1	0	0	1	1	0
Augend:		1	0	1	1	1
Addend:		1	1	0	1	1
XOR:		0	1	1	0	0

Figure 2: Carry generated in ith column is saved in $(i+1)$th column for $i = 0, 1, 2, \ldots, n$.

to 0 or their XOR value is equal to 0. But no mathematical proof regarding this result was discussed in [2]. This important result has been proved in this paper. Section 2 provides the basic concepts of CVT, MCVT, and XOR earlier defined in [1, 2]. In Section 3, it is proved that addition of any two nonnegative integers expressed as binary numbers is the same as addition of their CVT and their XOR values. This result is also shown to be true for any base of the number system. In Section 4, it is proved that in a successive addition of CVT and XOR of any two nonnegative integers, the maximum number of iterations required to get either CVT = 0 or XOR = 0 is equal to the length of the bigger integer expressed as a binary string. Further, in the same section, it is shown that MCVT of any two nonnegative integers = 0 requires a maximum of two iterations. In Section 5, an equivalence relation is defined using the concept of CVT, and the equivalence classes obtained due to it are presented.

2. Definitions of CVT and MCVT in Binary Number System

Let "a" and "b" be decimal representations of the binary strings $(a_n, a_{n-1}, \ldots, a_1)_2$ and $(b_n, b_{n-1}, \ldots, b_1)_2$, respectively, where each $a_i, b_i \in B_2 = \{0,1\}$ for all $i = 1, 2, \ldots, n$ and B_2^n be the set of all possible binary strings of length n on the set B_2. In binary number system, CVT as discussed in [1] is a mapping $\text{CVT} : B_2^n \times B_2^n \to B_2^n \times \{0\}$ defined by $\text{CVT}(a,b) = (a_n \wedge b_n, a_{n-1} \wedge b_{n-1}, \ldots, a_1 \wedge b_1, 0)_2$, whereas MCVT in [1] is a mapping $\text{MCVT} : B_2^n \times B_2^n \to B_2^n$ defined by $\text{MCVT}(a,b) = (a_n \wedge b_n, a_{n-1} \wedge b_{n-1}, \ldots, a_1 \wedge b_1)_2$. That is, to find out CVT, we perform the bit wise XOR operation of the operands to get a string of sum-bits (ignoring the carry-in while performing the addition of a and b) and simultaneously the bit wise ANDing of the operands to get a string of carry-bits, the latter string is padded with a "0" on the right is called the CVT of these operands as shown in Figure 1, and MCVT is only the ANDing values except the bit "0" padded on the right, and thus the relation between these two operation is $\text{CVT}(a,b) = 2 \times \text{MCVT}(a,b)$.

For example, suppose we want to find out the CVT of two numbers say 23 and 27. First of all, we have to find out the binary representation of these numbers, that is, $(23)_{10} \equiv (10111)_2$ and $(27)_{10} \equiv (11011)_2$.

The carry value is computed as in Figure 2.

Table 1: Shows the contributions in calculating the sum in different cases.

a_k	b_k	Sum of contributions of a_k and b_k in $a+b$	$c_k = a_k \wedge b_k$	Contribution of c_k in $\mathrm{CVT}(a,b)$	Contribution of $a_k \oplus b_k$ in $(a \oplus b)$	Sum of contributions of c_k and $a_k \oplus b_k$ in $\mathrm{CVT}(a,b)+(a\oplus b)$
0	1	2^{k-1}	0	0	2^{k-1}	2^{k-1}
1	0	2^{k-1}	0	0	2^{k-1}	2^{k-1}
0	0	0	0	0	0	0
1	1	2^k	1	2^k	0	2^k

Thus, $\mathrm{CVT}(23,27) = \mathrm{CVT}(10111,11011) = (1\wedge 1, 0\wedge 1, 1\wedge 0, 1\wedge 1, 1\wedge 1, 0)_2$ $(100110)_2 = (38)_{10}$, and $\mathrm{MCVT}(23,27) = (19)_{10}$. It may be noted that in any number system, CVT and MCVT are mapping from $Z \times Z$ to Z, where Z is set of nonnegative integers.

2.1. Extensions of CVT, MCVT, and XOR Operations for Arbitrary Number System

For any number system in base β, CVT of any two nonnegative integers $a = (a_n, a_{n-1}, \ldots, a_1)_\beta$ and $b = (b_n, b_{n-1}, \ldots, b_1)_\beta$ is defined by an integer $c = (c_n c_{n-1} \cdots c_1 0)_\beta$, where $c_i = \begin{cases} 1, \text{ if } a_i+b_i \geq \beta \\ 0, \text{ if } a_i+b_i < \beta \end{cases}$ for $i = 1,2,3,\ldots,n$. Similarly, MCVT of a and b in base β is the CVT value $c = (c_1 c_2 \cdots c_n)_\beta$ except the padding bit 0 in the least significant bit position. That is $\mathrm{CVT}(a,b) = \beta \times \mathrm{MCVT}(a,b)$ and the definition of XOR operation in binary number system can be extended for any number system in base β as $a \oplus b = ((a_n + b_n) \bmod \beta, (a_{n-1}+b_{n-1}) \bmod \beta, \ldots, (a_1+b_1) \bmod \beta)$, where + is the usual addition in decimal number system.

For example, in ternary number system, $\mathrm{CVT}(466,458) = \mathrm{CVT}(122021,121222) = (110110)_3 = 336$, $\mathrm{MCVT}(466,458) = \mathrm{MCVT}(122021,121222) = (11011)_3 = 112$, $\mathrm{XOR}(466, 458) = \mathrm{XOR}(122021,121222) = (210210)_3 = 588$.

3. Properties of CVT and XOR

We have observed in the last example that $\mathrm{CVT}(23,27) = 38$ and $\mathrm{XOR}(23,27) = 12$. Now $23+27 = 38+12$, that is, $23+27 = \mathrm{CVT}(23,27) + (23 \oplus 27)$. In general, we prove the following.

Theorem 3.1. *$a+b = CVT(a,b) + (a \oplus b)$, where a and b are any two nonnegative integers.*

Proof. Suppose $a = a_n a_{n-1} \cdots a_{k-1} a_k a_{k+1} \cdots a_2 a_1$ and $b = b_n b_{n-1} \cdots b_{k-1} b_k b_{k+1} \cdots b_2 b_1$ are the binary representations of a and b both expressed using n bits. Then, $\mathrm{CVT}(a,b) = c_n c_{n-1} c_{n-2} \cdots c_1 0$ for $i = 1,2,\ldots,n$. We will prove that sum of the contribution of a_k and b_k in $a+b$ is the same as the sum of the contribution of c_k and $a_k \oplus b_k$ in $\mathrm{CVT}(a,b) + (a \oplus b)$, where $k = 1,2,3,\ldots,n$. The place values of a_k and b_k in a and b are $a_k \times 2^{k-1}$ and $b_k \times 2^{k-1}$, respectively. So the total contributions of both a_k and b_k in $a+b$ is $(a_k+b_k)2^{k-1}$. The binary variable a_k and b_k can have four choices, and their place values are shown in Table 1.

From third column and seventh column, it can be verified that the total contribution of a_k and b_k in $a+b$ is the same as the sum of the contribution of c_k and $a_k \oplus b_k$ in $\mathrm{CVT}(a,b)+(a\oplus b)$ for $k = 1,2,\ldots,n$. Therefore, $a+b = \mathrm{CVT}(a,b) + (a \oplus b)$. □

Table 2: Shows the contributions in calculating the sum for two possible cases.

Cases	Conditions	Sum of contributions of a_k and b_k in $a+b$	c_k	Contribution of c_k in CVT(a,b)	Contribution of $a_k \oplus b_k$ in $(a \oplus b)$	Sum of contributions of c_k and $a_k \oplus b_k$ in CVT$(a,b)+(a\oplus b)$
Case 1	$a_k + b_k < \beta$	$(a_k + b_k)\beta^{k-1}$	0	0	$(a_k + b_k)\beta^{k-1}$	$(a_k + b_k)\beta^{k-1}$
Case 2	$a_k + b_k \geq \beta$	$(a_k + b_k)\beta^{k-1}$	1	β^k	$(a_k + b_k)\beta^{k-1} - \beta^k$	$(a_k + b_k)\beta^{k-1}$

Table 3: Generated sequences of CVT and XOR values.

Initial guess (x_0, y_0)	Generated sequences (x_{n+1}, y_{n+1})
$(1,8)$	$(0,9)$
$(12,10)$	$(16,6)$, $(0,22)$
$(17,11)$	$(2,26)$, $(4,24)$, $(0,28)$
$(1,23)$	$(2,22)$, $(4,20)$, $(8,16)$, $(0,24)$
$(1,15)$	$(2,14)$, $(4,12)$, $(8,8)$, $(16,0)$, $(0,16)$
$(27,5)$	$(2,30)$, $(4,28)$, $(8,24)$, $(16,16)$, $(32,0)$, $(0,32)$
$(127,65)$	$(130,62)$, $(4,188)$, $(8,184)$, $(16,176)$, $(32,160)$, $(64,128)$, $(0,192)$

(A) General Proof for an Arbitrary Number System

Let $a = \sum_{k=1}^{n} a_k \times \beta^{k-1}$ and $b = \sum_{k=1}^{n} b_k \times \beta^{k-1}$ be two numbers from a number system with base β, and let $\text{CVT}(a,b) = c_n c_{n-1} \cdots c_1 0$. We will prove that sum of the contribution of a_k and b_k in $a+b$ is the same as the sum of the contribution of c_k and $a_k \oplus b_k$ in $\text{CVT}(a,b) + (a \oplus b)$ for $k = 1,2,3,\ldots,n$.

Note that the individual place values of a_k and b_k in a and b are $a_k \times \beta^{k-1}$ and $b_k \times \beta^{k-1}$, respectively. So the total contributions for both a_k and b_k in $a+b$ is $(a_k + b_k)\beta^{k-1}$. Two cases arise: case 1, $a_k + b_k < \beta$ and case 2, $a_k + b_k \geq \beta$. In both cases, the contributions from a_k and b_k in $a+b$ remain $(a_k+b_k)\beta^{k-1}$, whereas in $a \oplus b$ they differ; for the first case it is $(a_k+b_k)\beta^{k-1}$ and for the second case it is $(a_k+b_k)\beta^{k-1} - \beta^k$. The value of c_k is zero in the first case, whereas in the second case it is 1 so that the respective contributions of c_k in $\text{CVT}(a,b)$ are 0 and β^k. It can be summarised in a table as shown in Table 2.

From third column and last column of Table 2, it can be seen that sum of the contribution of a_k and b_k in $a+b$ is the same as the sum of the contribution of c_k and $a_k \oplus b_k$ in $\text{CVT}(a,b) + (a \oplus b)$ for $k = 1,2,3,\ldots,n$. Therefore, $a + b = \text{CVT}(a,b) + (a \oplus b)$.

4. Convergence Behavior of CVT and MCVT

4.1. Convergence of CVT

Let $f : Z \times Z \to Z \times Z$ be defined as $f(a,b) = (\text{CVT}(a,b), (a \oplus b))$ for all $(a,b) \in Z \times Z$. Consider the iterative scheme $(x_{n+1}, y_{n+1}) = f(x_n, y_n)$, $n = 0,1,2,3,\ldots$. In this section, we will prove an important theorem which states that the sequence generated by the iterative scheme $(x_{n+1}, y_{n+1}) = f(x_n, y_n)$, $n = 0,1,2,3,\ldots$ converges to $(0, x_0 + y_0)$. The convergence behavior of CVT and XOR values of different order pairs are shown in Table 3.

The sequences generated from the ordered pair $(127,65)$ in Table 3 may be interpreted as $127 + 65 = 130 + 62 = 4 + 188 = 8 + 184 = 16 + 176 = 32 + 160 = 64 + 128 = 0 + 192$. These

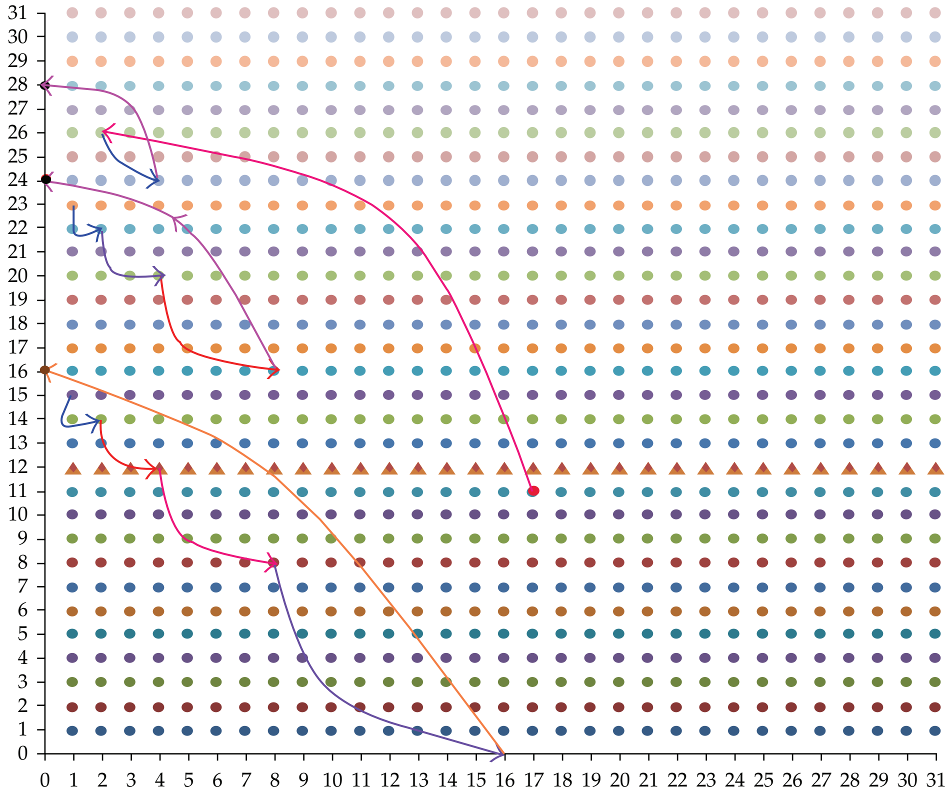

Figure 3: Showing the state transition diagrams (STDs) of three order pairs $(1,23)$, $(1,15)$, and $(17,11)$ as shown in Table 3.

generated sequences are named as the orbit of the order pair $(127,65)$. Figure 3 shows the state transition diagrams (STDs) of some of the points and their orbits.

Observations:

(1) in any number system, CVT = 0 in any iteration ⇔ in its previous iteration the sum of the corresponding bits of CVT and XOR is always less than the base of that number system;

(2) if two numbers expressed in binary are complement to each other, then their CVT = 0. But the converse is not true;

(3) if XOR value is 0 in any iteration, then CVT = 0 in the next iteration;

(4) the points in a single orbit are collinear as shown in Figure 3.

According to the definition of CVT for any two n-bit numbers, CVT will be of at most $(n + 1)$ bits. It seems that the recursive procedure of the CVT + XOR of two nonnegative integers always increases the length of the CVT by 1 in each iteration but it is not true, which is clear from the next proof.

Lemma 4.1. *If the maximum length of two nonnegative integers in binary representation is n then the CVT and XOR values in each iteration expressed in binary strings must be of length at most* $(n + 1)$*.*

Proof. Let a and b be two nonnegative integers with length at most n in their binary representations. Let c and d be two numbers to be added in kth iteration while performing the repeated sum of CVT and XOR. Suppose the number of (valid) bits in $\text{CVT}(c,d) \geq n+2$ (rejecting the zeros in the left of the first nonzero bit) in an iteration. The smallest number with valid $(n+2)$ bits is $100\cdots0 = 1 \times 2^{n+1} = 2^{n+1}$. So, $\text{CVT}(c,d) \geq 2^{n+1} \Rightarrow \text{CVT}(c,d) + (c \oplus d) \geq 2^{n+1}$.

Since $\text{CVT}(c,d) + (c \oplus d) = c + d$ (from Theorem 3.1), $c + d \geq 2^{n+1}$. Since $c + d = a + b$, so

$$a + b \geq 2^{n+1}. \tag{4.1}$$

The maximum number with $n+2$ bits is $111\cdots11 = 1 \times 2^{n-1} + 1 \times 2^{n-2} + \cdots + 1 \times 2^1 + 1 \times 2^0 = 1 + 2 + 4 + 8 + \cdots + 2^{n-2} + 2^{n-1}$.

Maximum value of $a + b$ is $2(1 + 2 + 4 + \cdots + 2^{n-2} + 2^{n-1}) = 2(2^n - 1)/(2-1) = 2^{n+1} - 2$:

$$\Rightarrow a + b \leq 2^{n+1} - 2. \tag{4.2}$$

From (4.1) and (4.2), we get $2^{n+1} \leq a + b \leq 2^{n+1} - 2$ which is absurd. Thus, our assumption was wrong, and hence all CVTs will be of at most $(n+1)$ bits in every iteration.

Same logic can be applied to XOR operation also, that is, if we write CVT in place of XOR in above proof, we also get an absurd result for XOR. Therefore, all XOR operations are of at most $(n+1)$ bits in every iteration. □

Lemma 4.2. *In any iteration if there is a "0" in CVT at kth position (counted from right), then there must be a "0" in $(k+1)$th position in the next iteration while forming the subsequent CVTs. The number of zeros in the CVT increases by at least one in each iteration.*

Proof. Suppose a CVT contains 0 at kth position in any iteration. In the next iteration, this 0 will be added to either 0 or 1 of XOR value obtained in the previous iteration. When we form CVT, $(k+1)$th position of CVT will be either $0 \wedge 1 = 0$ or $0 \wedge 0 = 0$. Thus, we get a 0 in the $(k+1)$th position of the newly formed CVT. Thus, once a "0" appears in a CVT in any iteration, then "0" appears in all subsequent CVT's in all subsequent iterations, but the position will be shifted by one in each iteration. By definition of CVT, one additional zero is added to the rightmost position in each iteration. So number of zero increases by at least one in a CVT in each iteration. □

Lemma 4.3. *If a and b are of maximum n binary bits, then the number of iterations required to get $\text{CVT} = 0$ is at most $(n+1)$.*

Proof. By Lemma 4.1, all CVTs will be of at most $(n+1)$ bits in all iterations.

By Lemma 4.2, once a "0" appears in a CVT in any iteration, then this zero will appear in all the subsequent CVT's in all subsequent iterations, but the position will be shifted by one in each iteration.

Also the number of zero in CVT increases by at least one in each iteration, the $(n+1)$ bits in CVT will be converted to $(n+1)$ zeros in at most $(n+1)$-iterations. □

Note. If a and b are of maximum n binary bits and Hamming distance between a and b is n, then $\text{CVT} = 0$ in one iteration. Otherwise, if Hamming distance between two selected numbers is k for $k < n$, then number of iterations required to get $\text{CVT} = 0$ is at most $(k+2)$.

Table 4: Showing calculation of MCVT.

a_i	b_i	$a_i \wedge b_i$	$a_i \oplus b_i$	$(a_i \wedge b_i) \wedge (a_i \oplus b_i)$
1	1	1	0	0
1	0	0	1	0
0	1	0	1	0
0	0	0	0	0

Lemma 4.4. *If a and b are of maximum n binary digits and $CVT = 0$ in $(n+1)$th iteration, then $XOR = 0$ in the nth iteration.*

Proof. Let us assume that CVT $= 0$ in the $(n+1)$th iteration and suppose XOR $\neq 0$ in the nth iteration. Then at least, one bit of the XOR in nth iteration must be "1". It is sure that in the kth iteration (where $k = 1, 2, 3, \ldots$ or $(n-1)$) of successive addition, XOR bit must be 1, and the corresponding carry bit must be 1 which is impossible. So our assumption was wrong. Thus, XOR $= 0$ in the nth iteration. Hence proved. □

Combining Lemmas 4.3 and 4.4, we have proved the following theorem.

Theorem 4.5. *Let $f : Z \times Z \to Z \times Z$ be defined as $f(a,b) = (CVT(a,b), (a \oplus b))$. Then, the iterative scheme $(x_{n+1}, y_{n+1}) = f(x_n, y_n)$, $n = 0, 1, 2, 3, \ldots$ converges to $(0, x_0 + y_0)$ for any initial choice $(x_0, y_0) \in Z \times Z$. Further, for any nonnegative integers "x_0" and "y_0" (where $x_0 \geq y_0$), the number of iterations required to get either $CVT = 0$ or $XOR = 0$ is at most the length of "x_0" when expressed as a binary string.*

4.2. Convergence of MCVT

The following theorem gives the number of iterations required for MCVT $= 0$.

Theorem 4.6. *The procedure of calculating the MCVT and XOR values of any two nonnegative integers requires a maximum of two iterations to get their MCVT $= 0$.*

Proof. Let $a = a_n a_{n-1} \cdots a_1$ and $b = b_n b_{n-1} \cdots b_1$ be two n-bits number. In the first iteration, we get MCVT$(a,\ b)$ and $a \oplus b$.

Let $x = \text{MCVT}(a,b) = (a_n \wedge b_n, a_{n-1} \wedge b_{n-1}, \ldots, a_1 \wedge b_1)$ and $y = a \oplus b = (a_n \oplus b_n, a_{n-1} \oplus b_{n-1}, \ldots, a_1 \oplus b_1)$. Then in the second iteration, we get MCVT(x,y) and $(x \oplus y)$. We will show that MCVT$(x,y) = 0$.

From Table 4, it can be verified that:

$$\begin{aligned} \text{MCVT}(x,y) &= ((a_n \wedge b_n) \wedge (a_n \oplus b_n), (a_{n-1} \wedge b_{n-1}) \wedge (a_{n-1} \oplus b_{n-1}), \ldots, (a_1 \wedge b_1) \wedge (a_1 \oplus b_1)) \\ &= (0, 0, 0, 0, \ldots, 0) = 0. \end{aligned} \tag{4.3}$$

If $a_i \wedge b_i \neq 1$ for all i, then MCVT$(a,b) = 0$ in one iteration. Hence proved. □

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	2	1	2	1	3	1	2	1	3	1	2	1	4	1	2	1	3	1	2	1	4	1	2	1	3	1	2	1	5
2	1	1	1	2	1	1	2	3	1	1	2	2	1	1	3	4	1	1	2	2	1	1	3	3	1	1	2	2	1	1	4	5
3	1	2	2	1	1	3	3	3	1	3	2	2	1	4	4	4	1	3	2	2	1	4	3	3	1	3	2	2	1	5	5	5
4	1	1	1	1	1	2	2	2	1	1	1	1	2	3	3	3	1	1	1	1	2	2	2	2	1	1	1	1	3	4	4	4
5	1	2	1	3	2	1	2	3	1	2	1	4	3	3	3	2	1	2	1	4	2	2	2	3	1	2	1	5	4	4	4	4
6	1	1	2	3	2	2	1	2	1	1	3	4	3	3	3	3	1	1	3	3	2	2	2	2	1	1	4	5	4	4	4	4
7	1	3	3	3	2	3	2	1	1	4	4	4	3	2	3	3	1	4	3	3	2	3	2	2	1	5	5	5	4	4	4	4
8	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3
9	1	2	1	3	1	2	1	4	2	1	2	3	2	2	2	4	1	2	1	3	1	2	1	5	3	3	3	2	3	3	3	4
10	1	1	2	2	1	1	3	4	2	2	1	2	2	2	3	3	1	1	2	2	1	1	4	5	3	3	3	3	3	3	2	3
11	1	3	2	2	1	4	4	4	2	3	2	1	2	4	3	3	1	3	2	2	1	5	5	5	3	2	3	3	3	4	3	3
12	1	1	1	1	2	3	3	3	2	2	2	2	1	2	2	2	1	1	1	1	3	4	4	4	3	3	3	3	3	3	3	3
13	1	2	1	4	3	3	3	2	2	2	2	4	2	1	2	3	1	2	1	5	4	4	4	4	3	3	3	4	3	3	3	3
14	1	1	3	4	3	3	3	3	2	2	3	3	2	2	1	2	1	1	4	5	4	4	4	4	3	3	2	3	3	3	4	3
15	1	4	4	4	3	2	3	3	2	4	3	3	2	3	2	1	1	5	5	5	4	4	4	4	3	4	3	3	3	3	3	3
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
17	1	2	1	3	1	2	1	4	1	2	1	3	1	2	1	5	2	1	2	3	2	2	2	4	2	2	2	3	2	2	2	5
18	1	1	2	2	1	1	3	3	1	1	2	2	1	1	4	5	2	2	1	2	2	2	3	3	2	2	2	2	2	2	4	4
19	1	3	2	2	1	4	3	3	1	3	2	2	1	5	5	5	2	3	2	1	2	4	3	3	2	3	2	2	2	5	4	4
20	1	1	1	1	2	2	2	2	1	1	1	1	3	4	4	4	2	2	2	2	1	2	2	2	2	2	2	2	3	3	3	3
21	1	2	1	4	2	2	2	3	1	2	1	5	4	4	4	4	2	2	2	4	2	1	2	3	2	2	2	5	3	3	3	3
22	1	1	3	3	2	2	2	2	1	1	4	5	4	4	4	4	2	2	3	3	2	2	1	2	2	2	4	4	3	3	3	3
23	1	4	3	3	2	3	2	2	1	5	5	5	4	4	4	4	2	4	3	3	2	3	2	1	2	5	4	4	3	3	3	3
24	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2	1	2	2	2	2	2	2	2
25	1	2	1	3	1	2	1	5	3	3	3	2	3	3	3	4	2	2	2	3	2	2	2	5	2	1	2	3	2	2	2	4
26	1	1	2	2	1	1	4	5	3	3	3	3	3	3	2	3	2	2	2	2	2	2	4	4	2	2	1	2	2	2	3	3
27	1	3	2	2	1	5	5	5	3	2	3	3	3	4	3	3	2	3	2	2	2	5	4	4	2	3	2	1	2	4	3	3
28	1	1	1	1	3	4	4	4	3	3	3	3	3	3	3	3	2	2	2	2	3	3	3	3	2	2	2	2	1	2	2	2
29	1	2	1	5	4	4	4	4	3	3	3	4	3	3	3	3	2	2	2	5	3	3	3	3	2	2	2	4	2	1	2	3
30	1	1	4	5	4	4	4	4	3	3	2	3	3	3	3	3	2	2	4	4	3	3	3	3	2	2	3	3	2	2	1	2
31	1	5	5	5	4	4	4	4	3	4	3	3	3	3	3	3	2	5	4	4	3	3	3	3	2	4	3	3	2	3	2	1

Figure 4: Showing the number of iterations required for either CVT = 0 or XOR = 0.

5. An Equivalence Relation is Defined Using the Notion of CVT

Let $A = \{0, 1, 2, 3, \ldots, 2^n - 1\}$ be a finite subset of Z for some nonnegative integer n, and let R be a relation on $A \times A$ defined as $(a, b)R(c, d) \Leftrightarrow (a, b)$ and (c, d) requiring equal number of iterations for their CVT = 0 or XOR = 0.

It can be easily verified that the relation R is reflexive, symmetric, and transitive on the set $A \times A$. Therefore, R is an equivalence relation on $A \times A$.

We have calculated the number of iterations required for the set of ordered pair in $A \times A$, where $A = \{0, 1, 2, \ldots, 31\}$ and constructed Figure 4 using a two-step procedure as follows.

Step 1. Write all the integers $0, 1, 2, 3, \ldots, 31$ in ascending order in both, uppermost row and leftmost column of Figure 4.

Step 2. Compute number of iterations required for any ordered pair (a, b) to get either CVT = 0 or XOR = 0 and store it in the position (a, b).

From Figure 4, we have observed that:

(1) the matrix is symmetric;

(2) if we consider Figure 4 as 4 quadrants, each quadrant is a symmetric matrix. Again if each quadrant is divided further into 4 smaller quadrants, then also the 1st quadrant is the same as the 3rd quadrant. Hence a self-similar fractal behaviour is noticed in Figure 4;

(3) in a block of size $(2^n - 1) \times (2^n - 1)$, there are no ordered pairs in the 2nd quadrant which transform into CVT = 0 or XOR = 0 in n-iterations.

In Figure 4 R divides the set $\{0, 1, 2, 3, \ldots, 2^n - 1\} \times \{0, 1, 2, 3, \ldots, 2^n - 1\}$ into n disjoint equivalence classes.

For $n = 1$, there is one equivalence class $[(0, 0)] = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$ and $|[0, 0]| = 4$.

For $n = 2$, there are two equivalence classes $[(0, 0)], [(1, 3)]$, where

$$[(0,0)] = \{(0,0), (0,1), (0,2), (0,3), (1,0), (1,1), (1,2), (2,0), (2,1), (2,2), (3,0), (3,3)\}, \tag{5.1}$$

$$[(1,3)] = \{(1,3), (2,3), (3,1), (3,2)\}. \tag{5.2}$$

Here, $|[0, 0]| = 12, |[1, 3]| = 4$.

For $n = 3$, there are three equivalence classes $[(0, 0)], [(1, 3)]$, and $[(1, 7)]$:

$$|[0,0]| = 34, |[1,3]| = 18, |[1,7]| = 12. \tag{5.3}$$

For $n = 4$, there are four equivalence classes $[(0, 0)], [(1, 3)], [(1, 7)]$, and $[(1, 15)]$:

$$|[0,0]| = 96, |[1,3]| = 78, |[1,7]| = 58, |[1,15]| = 24. \tag{5.4}$$

For $n = 5$, there are five equivalence classes $[(0, 0)], [(1, 3)], [(1, 7)], [(1, 15)]$, and $[(1, 31)]$:

$$|[0,0]| = 274, |[1,3]| = 306, |[1,7]| = 263, |[1,15]| = 133, |[(1,31)]| = 48. \tag{5.5}$$

From above, we conclude that if we take a block of size $(2^n - 1) \times (2^n - 1)$, then

(1) number of ordered pairs for which CVT = 0 or XOR = 0 in one iterations is $3^n + (2^n - 1)$ for $n = 1, 2, 3, 4, \ldots$;

(2) number of ordered pairs for which CVT = 0 or XOR = 0 in n iterations is $3 \times 2^{n-1}$ for $n = 3, 4, 5, \ldots$.

6. Conclusion and Future Research Work

In the present paper, we have proved some important results on Carry Value Transformation (CVT) and Modified Carry Value Transformation (MCVT). Firstly, it has been proved that for any base of the number system, the sum of any two nonnegative integers is the same as the sum of their CVT and XOR values. This result is actually the correctness proof of the algorithm based on which the adder circuit is designed in [2]. Our second result, that is, "the number of iterations leading to either CVT = 0 or XOR = 0 does not exceed the maximum of the lengths of the two addenda expressed as binary strings" is about the efficiency at which the hardware circuit designed in [2] will produce the addition result. The state transition diagrams (STDs) and certain observations on CVT and MCVT are found out. Our third result such as addition of Modified Carry Value Transformation (MCVT) and XOR requires a maximum of two iterations for MCVT to be zero, is an interesting result for MCVT. A new equivalence relation is obtained on the set $Z \times Z$ which divides the CV Figure 4 into disjoint equivalence classes.

In future we propose to study the following aspects:

(1) investigating into the state transition diagrams (STDs) of different IVTs;

(2) extending the domain of CVT from nonnegative integers to real numbers and complex numbers;

(3) exploring the behaviour of hybrid IVTs and their applications;

(4) explaining the relationship of IVTs with cellular automata.

Acknowledgments

The authors would like to acknowledge Professor P. Pal Choudhury and Sk. Sarif Hassan, Indian Statistical Institute, Kolkata for motivating us to work further in the domain of CVT and IVT.

References

[1] P. P. Choudhury, S. Sahoo, B. K. Nayak, and Sk. S. Hassan, "Carry value transformation: application in fractal formation," in *Proceedings of the IEEE International Advance Computing Conference (IACC '09)*, pp. 2613–2618, Patiala, India, March 2009.

[2] P. P. Choudhury, S. Sahoo, and M. Chakraborty, "Implementation of basic arithmetic operations using cellular automaton," in *Proceedings of the 11th International Conference on Information Technology (ICIT '08)*, pp. 79–80, Bhubaneswar, India, December 2008.

[3] P. P. Choudhury, S. Sahoo, B. K. Nayak, and Sk. S. Hassan, "Act of CVT and EVT in the formation of number-theoretic fractals," *International Journal of Computational Cognition*, vol. 9, no. 1, pp. 1–8, 2011.

[4] S. Sahoo, I. Mohanty, G. Chowdhary, and A. Panigrahi, "2-variable Boolean operation—its use in pattern formation," *Nonlinear Sciences*. In press. http://arxiv.org/abs/1008.2530.

[5] Sk. S. Hassan, P. P. Choudhury, S. Das, R. Singh, and B. K. Nayak, "Collatz function like integral value transformations," *Alexandria Journal of Mathematics*, vol. 1, no. 2, 2010.

[6] P. P. Choudhury, S Sahoo, B. K. Nayak, and Sk. S. Hassan, "Theory of rule 6 and its application to round robin tournament," *International Journal of Computational Cognition*, vol. 8, no. 3, pp. 33–37, 2010.


Advances in
Operations Research
Advances in
Decision Sciences
Journal of
Applied Mathematics
Algebra
Journal of
Probability and Statistics
The Scientific
World Journal
International Journal of
Differential Equations
Hindawi
Submit your manuscripts at
http://www.hindawi.com
International Journal of
Combinatorics
Advances in
Mathematical Physics
Journal of
Complex Analysis
Journal of
Mathematics
Mathematical Problems
in Engineering
Abstract and
Applied Analysis
Discrete Dynamics in
Nature and Society
International
Journal of
Mathematics and
Mathematical
Sciences
Journal of
Discrete Mathematics
Journal of
Function Spaces
International Journal of
Stochastic Analysis
Journal of
Optimization