Cycle Counting For Information Security in I@I Swarm Internet Generating Q Swarm Memory

Prof. Dr. Fevzi Ünlü
Retired Professor of Computer Science and Applied Mathematics
Ege University & Yasar University, Izmir

Özet—*Bir Q^+ kümesinden gelişigüzel alınan bir Q sürü belleği, LOS ve RDS ile adlandırılan iki farklı durumda bulunur. LOS durumunda Q belliğinin, bildirişim yapaya uygun olan tamra ve ramta anten çiftleri bulunur ve bildirişim yaparken kullanacakları frekanslar onlara atanır. Atanan bu frekanslar Q sürü belleği RDS durumuna geçtiğinde kullanılır. Devirli sayma CC bir Q sürü belleği LOS durumunda iken, onun yapısında bulunan ve bildirişim yapmaya muktedir olan, tamra ve ramta anten çiftlerini bularak eşleyen ve hangi frekanslarda bildirişim yapacaklarını bulan bir sayma işlemi biçimidir. Bir Kip[n], n ∈ N, dilinde CC ile bulunan frekans kodları, bir I@I sürü internet modeli üreten Q sürü bellek modeli RDS durumunda iken; onun yapısında bulunan tamra ve ramta antenlerinin bildirişim yapmada kullanacağı en uygun frekanslardır. Yazar bunun için, bu bildiride, bir Q sürü belleği LOS durumunda iken; CC tekniği ile onun RDS durumunda güvenli bildirişim yapmaya muktedir olacak tamra ve ramta anten çiftlerine atanacak olan, en uygun bildirişim frekanslarını bulma çalışmasını yapmaktadır.*

Abstract— A Q of Q^+ swarm memory closure can be in two distinguishable states**.** They are namely called LOS and RDS of Q. Cycle counting CC is a process for pairing and marking the communicating tamra and ramta antennas appearing in a Q of Q^+ while Q in LOS by the same frequency codes in a Kip[n], n ∈ N. Frequency codes are used to realize the observe able and secure optimal communicational activities in the I@I swarm internet model generating Q swarm memory model while it is in RDS. Author is studding CC techniques in order to code securely communicating pairs of the tamra and ramta antennas while Q is in LOS. The basic supporting information can be found in Ünlü at all and [1-14].

Index Terms— Cycle counting, internet, I@I, ITC, IRC, KBO, LOS, memory, N, Q, RCR, RDS, security, swarm.

I. INTRODUCTION

I@I and Q are tic RCR formatted KBOs with a pair of ITC and IRC systems. Tamra and ramta antenna systems in the rator and rand of a distributed communicating I@I swarm internet model generating Q swarm memory model are responsible from ITC and IRC. They were studied in Ünlü at all. I@I and Q can be in two different states. Namely they are called randomly distributed state RDS and linearly ordered state LOS. For designing formally optimal communicational algorithms to realize information processing in a distributed Q swarm memory model software chip, some cycle counting CC models were needed for pairing and marking the communicating tamra and ramta antennas while Q in LOS. A set of suitable CC models are found to mark tamra and ramta antennas on Q in LOS in ordered to be used in RDS in this paper. Basic background information is available in [1-14].

II. FOUNDATION

In this section the main tools of communicating antenna resource cycle counting models in a distributed Q of Q^{+n} swarm memory closure for generating an distributed I@I of $I@I^+$ swarm internet closure are shortly introduced by using DTS: Descriptive Text Sequence [1-14] for the sake of easy readability. The reader should read each DTS in the given order in order to grasp the notion of different cycle counting process types.

DTS-1: Brief words

01. KBO: “Knowledge Based Object,”
02. ITC: “Information Transmitting Channel,”
03. IRC: “Information Receiving Channel,”
04. TIC | tic: “undividable tiny compact and dense information construct,”
05. CIC | cic: “Coarse easily dividable into other compact and dense tic information construct,”
06. Rator: “Operator” | “Operator register,”
07. Rand: “Operand” | “Operand register,”
08. @ | CAP | cap: “Communicational Attraction and Process Controlling Center,”
09. RCR: Rator Cab Rand. It is used to generate RCR formatted symmetric KBO software model.
10. I@I: Internet @ Internet.
11. N: The set of natural number.
12. LOS: Linearly Ordered State.
13. RDS: Randomly Distributed State.
14. CC: Cycle Counting.
15. ISTS: Information Sensing and Transmitting State.
16. ISRS: Information Sensing and Receiving State.

DTS-2: Tics, IDC, P_1 and P_2

1. (i) t: tic tamra antenna resource, (ii) x or y: tic control variable resource, (iii) @: cab: tic cap resource, (iv) c: tic constant resource, and (v) r: tic ramta antenna resource. They are used in the construction of a tic RCR formatted Q KBO construction.

2. IDC = {0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 01000, 1001, 1010, 1011, 1100, 1101, 1110, 1111}. Where IDC stands for "Image Dress Code" in which "1" represents observe able component while "0" represents none observe able component in a vector representation of a tic KBO.

3. P_1: A property vector which has only 1 control variable x in its structure. P_2: A property vector which has only 2 control variables x and y in its structure.

DTS-3: P_1 and P_2 form design and meaning of P_1 and P_2

1. P_1 = <C, T; K = K[n] = Kip[n], x, n; 4B: bbbb ∈ IDC> is a form design.

Meaning of P_1 = <C: Communicating, T: TASIM genetic; K = Kip[n] = {0, 1, 2 ... n-1}, x: 1 control variable named by x, n ≥ 1 and n ∈ N; 5B: bbbb ∈ IDC >.

2. P_2 = <C, T; < K = Kip[m], K = Kip[n] >, <x, y>, <m, n>: m, n ≥ 1 and m, n ∈ N; 5B: bbbb ∈ IDC > is a form design.

The meaning of P_2 = <C: Communicating, T: TASIM genetic; < K = K[m] = Kip[m] ={0, 1, 2, ..., m-1}, K = K[n] = Kip[n] ={0, 1, 2, ..., n-1} >, <x, y>: 2 control variables named by x and y, <m, n>: 2 natural number named by m and n such that m, n ≥ 1 and m, n ∈ N; 4B: bbbb ∈ IDC > is a form design.

2.1 Cycle Counting Models

Different cycle counting models are introduced in the following different text boxes.

DTS-4: FRCC

Definition 1: A proper forward counting is generated by using only numerals appearing in **K** = K[n] = Kip[n] such a way that all numbers are generated in length 1 in the order of n-tuple cycles is called a **forward regular cycle counting** on K = K[n] = Kip[n]. It is briefly represented by FRCC[n]. Where, "n-tuple" meaning is "a sequence that it contains n element in it".

Theorem 1: There is one forward regular n-tuple cycle counting FRCC[n].

Proof: 0, 1, 2, 3 ... (n-1), 0, 1, 2, 3 ... (n-1), 0, 1, 2, 3 ... (n-1), 0, 1, 2, 3 ... is forward regular cycle counting FRCC[n].

Example 1: If n = 8, then 8–tuple K = K[8] = Kip[8] = {0, 1, 2, 3, 4, 5, 6, 7} ≡ "0 | 1 | 2 | 3 | 4 | 5 | 6 | 7" is obtained. FRCC [8] is carried in the length 1 vocabularies in the order of 8-tuple cycles by using 8 numerals in K = K[8] = Kip[8]. Hence 0, 1, 2, 3, 4, 5, 6, 7, 0, 1, 2, 3, 4, 5, 6, 7 ... is a forward regular cycle counting FRCC[8].

DTS-5: BRCC

Definition 2: The reverse counting process of FRCC[n] is called backward regular cycle counting by n-tuple on K = Kip[n]. It is briefly represented by BRCC[n].

Theorem 2: There is one backward regular n-tuple cycle counting BRCC[n].

Proof: "(n-1), (n-2) ... 3, 2, 1, 0, (n-1), (n-2) ... 3, 2, 1, 0, (n-1), (n-2) ... 3, 2, 1, 0..." is backward regular n-tuple cycle counting BRCC[n] on K = K[n] = Kip[n] .

Example 2: If n = 8. K = Kip[8] = {0, 1, 2, 3, 4, 5, 6, 7} ≡ "0 | 1 | 2 | 3 | 4 | 5 | 6 | 7", then "7, 6, 5, 4, 3, 2, 1, 0, 7, 6, 5, 4, 3, 2, 1, 0 ... " is a BRCC[8] on K = Kip[8].

DTS-6: Root Cycle of FRCC

Definition 3: The first n-tuple of FRCC[n] or BRCC[n] is called the root cycle of FRCC[n] or BRCC[n]. They are briefly represented by RC-FRCC[n] and RC-BRCC[n].

Theorem 3: There is only one root cycle of FRCC[n] or BRCC[n] on K = K[n] = Kip[n].

Proof: **(a)** "0, 1, 2, 3 ... (n-1)" is the root cycle of FRCC[n] on K = K[n] = Kip[n]. (b) "(n-1), (n-2) ... 3, 2, 1, 0" is the root cycle of BRCC[n] on K = K[n] = Kip[n].

Example 3: **(a)** "0, 1, 2, 3, 4, 5, 6, 7" is the root cycle of FRCC [8] on K = K [8] = Kip [8]. (b) "7, 6, 5, 4, 3, 2, 1, 0" is the root cycle of BRCC[8] on K = Kip[8].

DTS-7: Root cycle of FBRCC

Definition 4: The concatenation of the root cycles of RC-FRCC[n] and RC-BRCC[n] on a K = Kip[n] is briefly represented by RC-FBRCC [2×n].

Theorem 4: There is only one RC-FBRCC [2×n] on K = K[n] = Kip[n].

Proof: "0, 1, 2, 3 ... (n-2), (n-1)". "(n-1), (n-2) ... 3, 2, 1, 0" = "0, 1, 2, 3 ... (n-2), (n-1), (n-1), (n-2) ... 3, 2, 1, 0" is RC-FBRCC [2×n] on K = Kip[n].

Example 4: (a) "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0" is RC-FBRCC [2×8] on K = Kip [8].

Definition 5: RC-FBRCC [2×n] = "0, 1, 2, 3, ..., n-2, n-1, n-1, n-2 ... 3, 2, 1, 0 " is also called the root cycle of the forward and backward mirror cycle counting RC-FBMCC [2×n] by 2 × n-tuple on K = Kip[n].

2.2 Image Designs for Forward Regular Mirror Cycle Counting

DTS-8: Numerical Image A, B and C

(a)"IA: Image A" is a RC-FRCC[n] on K = Kip[n]. It observes itself in the mirror as RC-BRCC[n] on K = Kip[n]. "IB: Image B" is RC-BRCC[n] on K = Kip[n]. It observes itself in the mirror as RC-FRCC[n] on K = Kip[n]. "IC: Image C" is RC-RMCC[n] on K = Kip[n]. It observes itself in the mirror as a RC-RMCC[n] on K = Kip[n]. Hence:

(b) IA = "0, 1, 2, 3... n-2, n-1" ‖ "n-1, n-1, n-2 ... 3, 2, 1, 0 " = IB ⟺ "IA ‖ IB and IB ‖ IA". It implies that IC = RC-RMCC[n] = "0, 1, 2, 3 ... n-2, n-1, n-1, n-2 ... 3, 2, 1, 0" ‖ "0, 1, 2, 3 ... n-2, n-1, n-1, n-2 ... 3, 2, 1, 0" = RC-RMCC[n] = IC ⟺ IC ‖ IC.

Where, "**‖" is double sided mirror.**

DTS-9: Observer O

Assuming "O" is a remote observer to observe the action of IA and IB together on its own mirror, while IA and IB are looking into their mirrors. Following cases are possible:

(i) If IA is looking into its mirror with its eye-detectors, then it observes itself as IB.

(ii) If IB is looking into the mirror with its eye-detector then it observes itself as IA.

(iii) If O, observes IA and IB together while they are looking into mirror, then O observes IA and IB images in its own eye mirror as RC-RMCC[n] on K = Kip[n].

DTS-10: IC as an observer

Theorem 5: Assume that IC is RC-RMCC[n] on K = Kip[n]. IC observes itself in its mirror.

Proof: Let us assume that "‖" is a two sided eye detectible mirror. Assume that IC is RC-RMCC[n] on K = Kip[n]. It implies that IC = "0, 1, 2, 3, … (n-1), (n-1), (n-2) … 3, 2, 1, 0" ‖ "0, 1, 2, 3 … (n-1), (n-1), (n-2) … 3, 2, 1, 0" = IC⟺ IC ‖ IC.

Example 5: Let us assume that "‖" is a two sided eye detectible mirror and IC = "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0" is RC-RMCC [8] on K = Kip [8]. This implies that IC = "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0" ‖ "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0" = IC⟺ IC ‖ IC.

DTS-11: Relation between RC-FRCC and RC-BRCC

Theorem 6: If RC-FRCC[n] is the root cycle of a forward regular cycle counting by n-tuple on K = Kip[n] and RC-BRCC[n] is the root cycle of a backward regular cycle counting by n-tuple on K = Kip[n]. Then

(i) RC-FRCC[n] observes RC-BRCC[n] in the mirror.

(ii) RC-BRCC[n] observes RC-FRCC[n] in the mirror.

Proof: Assume that RC-FRCC[n] is the root cycle of a forward regular cycle counting by n-tuple on K = Kip[n] and RC-BRCC[n] is the root cycle of a backward regular n-tuple cycle counting by n-tuple on K = K[n] = Kip[n]. Then:

(i) RC-FRCC[n] = "0, 1, 2, 3 … (n-2), (n-1)" ‖ "(n-1), (n-2), …, 3, 2, 1, 0" = RC-BRCC[n] in the mirror.

(ii) RC-BRCC[n] = "(n-1), (n-2) … 3, 2, 1, 0" ‖ "0, 1, 2, 3, …, (n-2), (n-1)" = RC-FRCC[n] in the mirror.

Example 6: (i) If IA = RC-FRCC [8] = "0, 1, 2, 3, 4, 5, 6, 7" ‖ "7, 6, 5, 4, 3, 2, 1, 0" = RC-BRCC [8] = IB.

(ii) If IB = RC-BRCC [8] = "7, 6, 5, 4, 3, 2, 1, 0" ‖ "0, 1, 2, 3, 4, 5, 6, 7" = RC-FRCC [8] = IA.

DTS-12: Rator and rand part of a RC-RMCC

Definition 6: Let us assume that RC-RMCC[m] = "0, 1, 2, 3, …, m-1, m-1 … 3, 2, 1, 0" is given for m ≥ 1 and m ∊ N. (a) The first m section of RC-RMCC[m] is called a rand part of RC-RMCC[m]. (b) The second m section of RC-RMCC[m] is called a rator part of RC-FRMCC[m]. (c) If an "o_i" is the i[th] tic tab memory component, injected in front of each i[th] numeral for using "i → < o_i i >" transformation, in the rand part of the RC-RMCC[m]; then the new RC-RMCC[m] is called as an **m black hole tab tic memories injected into RC-RMCC[m]**. It is briefly represented by RC-BHIRMCC[m]. Where, "o" is an extendable and contractible tic black hole memory. It can't be observed by any technology for time being. Only T can store very dense, observable, T genetic, communicating, forward and backward regular mirror cycle countable tic information in it.

Theorem 7: There is at least one RC-BHIRMCC[m] for m ≥ 1 and m ∊ N.

Proof: Let us assume that "0, 1, 2, 3 … m-1, m-1 … 3, 2, 1, 0" is a RC-RMCC[m] for each m ≥ 1 and m ∊ N. The construction of "0, 1, 2, 3 … m-1, o_{m-1}, m-1, o_{m-2}, m-2 … o_3, 3, o_2, 2, o_1, 1, o_0, 0" is an m black hole tab memories injected RC-BHIRMCC[m].

Example 7: "0, 1, 2, 3 … 7, o_7, 7, o_6, 6 … o_3, 3, o_2, 2, o_1, 1, o_0, 0" is 8 black holes injected RC-BHIRMCC [8].

DTS-13: Mixed and Smooth Complex RMCC

Definition 7: Let assume that A = A [m, o] = "0, 1, 2, 3 … m-2, m-1, o_{m-1}, m-1, o_{m-2}, m-2 … o_3, 3, o_2, 2, o_1, 1, o_0, 0" is a RC-BHIRMCC[m] and B_i = B[n_i] is a RC-RMCC [n_i]. C = C [m, n = <n_0, n_1, n_2 … n_{m-1}>] = A [m, o_i← B_i = B[n_i]] is called a **mixed complex RMCC[<m, n= <n_0, n_1, n_2 … n_{m-1}>>]**. Further: If ∀n_i = s, then RMCC [<m, n = <m x s >] is called **a smooth complex RMCC [m, n = <m x s>]**. The first one is briefly represented by MCRMCC [<m, n= <n_0, n_1, n_2 … n_{m-1}>>]. The second is briefly represented by SCRMCC [m, n = <m × s>].

Theorem 8: (a) There is at least one SCRMCC [m, n = <m × s>].

Proof: Consider: (i) A = A[m, o] = "0, 1, 2, 3 … m-1, o_{m-1}, m-1, o_{m-2}, m-2 … o_3, 3, o_2, 2, o_1, 1, o_0, 0" is a SCRMCC[m, n = <m × s>] for each m ≥ 1 and m ∊ N. Consider ∀B_i [s ← n] = B[n] = "0, 1, 2, 3 … n-1, n-1 … 3, 2, 1, 0" is a RC-RMCC[n] for each n ≥ 1 and n ∊ N. Construct C = C[m, <m × n>] = A[m, o:∀o_k ← B[n]] = "0, 1, 2, 3 … m-2, m-1, o_{m-1}←B [n], m-1, o_{m-2}←B[n], m-2 … o_3←B[n], 3, o_2←B[n], 2, o_1←B[n], 1, o_0←B[n], 0" = "0, 1, 2, 3, …, m-2, m-1, "0, 1, 2, 3, ..., n-1, n-1, …, 3, 2, 1, 0", m-1, "0, 1, 2, 3, ..., n-1, n-1, …, 3, 2, 1, 0", m-2, …, "0, 1, 2, 3, ..., n-1, n-1, …, 3, 2, 1, 0", 3, "0, 1, 2, 3, ..., n-1, n-1, …, 3, 2, 1, 0", 2, "0, 1, 2, 3, ..., n-1, n-1, …, 3, 2, 1, 0", 1, "0, 1, 2, 3, ..., n-1, n-1, …, 3, 2, 1, 0", 0 " is a SCRMCC[<m, <m x n >>].

DTS-14: Smooth and complex RMCC

Example 7: Consider A[8, o] = "0, 1, 2, 3, 4, 5, 6, 7, o_7, 7, o_6, 6, o_5, 5, o_4, 4, o_3, 3, o_2, 2, o_1, 1, o_0, 0" is 8 black hole tab memories injected RC-BHIRMCC [8]. Let assume that ∀B_i [s ← n = 8] = B[8] = "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0" is a RC-RMCC[8]. Construct C = C[<8, <8×8>] = A[8, ∀o_i ← B[8]] = "0, 1, 2, 3, 4, 5, 6, 7, "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0", 7, "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0", 6, "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0", 5, "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0", 4, "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0", 3, "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0", 2, "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0", 1, "0, 1, 2, 3, 4, 5, 6, 7, 7, 6, 5, 4, 3, 2, 1, 0", 0" is a SCRMCC [8, <8 × 8>].

Theorem 8: There is at least one MCRMCC [<m, n = <n_0, n_1, n_2 … n_{m-1}>].

Proof: Assume A = A [m, o] = "0, 1, 2, 3 … m-2, m-1, o_{m-1}, m-1, o_{m-2}, m-2 … o_3, 3, o_2, 2, o_1, 1, o_0, 0" is a RC-BHIRMCC [m] for m ≥ 1 and m ∊ N. Assume B_{m-1}[1] = "0, 0", B_{m-2}[2] = "0, 1, 1, 0" … B_2[m-2] = "0, 1, 2, 3 … m-3, m-3 … 3, 2, 1, 0", B_1[m-1] = "0, 1, 2, 3 … m-2, m-2 … 3, 2, 1, 0", B_0[m] = "0, 1, 2, 3 … m-1, m-1 … 3, 2, 1, 0". Construct C[m, <1, 2, 3 … m>] = "0, 1, 2, 3 … m-2, m-1, "0, 0", m-1, "0, 1, 1, 0" m-2 … "0, 1, 2, 3 … m-3, m-3 … 3, 2, 1, 0", 2, "0, 1, 2, 3, …, m-2, m-2, …, 3, 2, 1, 0", 1, "0, 1, 2, 3, …, m-1, m-1, …, 3, 2, 1, 0" 0". It is a MCRMCC [<m, <1, 2, 3 … m>>].

Example 8: "0, 1, 2, 3, "0, 1, 2, 2, 1, 0", 3, "0, 1, 1, 0", 2, "0, 1, 2, 3, 3, 2, 1, 0", 1, "0, 1, 2, 2, 1, 0", 0" is a MCRMCC [4, <3, 2, 4, 3>>].

III. APPLICATION

In this section, SCRMCC [<m, <m × n >] and MCRMCC [<m, n = <n_0, n_1, n_2 … n_{m-1}>] are applied for counting the pairs of communicating tamra and ramta antenna systems in a Q of Q^+ in the following DTS[15-24].

3.1 Cycle Counting on Communicating Tamra and Ramta Antennas in Q

DTS-15a: Constructing a L [K, G] type formal language

Definition 8: In a communicating T genetic Q swarm memory constructing Environment; "t" is called a tic tamra antenna, "r" is called a tic ramta antenna, "c" is called a tic KBO constant, "x or y" is called a tic KBO variable and K = K[n] = Kip[n] = {0, 1, 2, …, n-1} is called a numerical alphabet for supporting generation of a communicating formal language L = L[K, G] .

Where, G is a formal grammar for generating formal language L from tic numerals of K.

DTS-15b: Q [K, G] formal language

Definition 9: In a communicating T genetic Q swarm memory constructing Environment: If A = {t, x, y, @, c, r, k} is a TASIM genetic alphabet and G is a TASIM genetic formal grammar, then Q[A , G] is a TASIM genetic formal Q swarm memory language.

DTS-16: Elements in Q [A, G]

Assume that

1) "$_k$t" is a ramta antenna that it has a capacity to sense, receive and transmit any tic information coded into the signal "k" while k □ L[K, G].

2) "r_k" is A ramta antenna that it has a capacity to sense, receive and transmit any tic information coded into the signal "k" while k □ L[K, G].

3) "$_k$t … r_k" is a < t: tamra, r: ramta> antenna pair in a word Q swarm memory that it has capacity to communicate with each other by sensing, receiving and transmitting tic information coded into the signal "k" while k □ L[K, G].

Communication Rule: Each of t and r antennas in the pair has a capacity to be in two different states namely ISRS and ISTS. In the initial state, tamra antenna becomes in the state of ISRS. In the next state the pair changes its state as pair wise. Tamra antenna becomes in the state of ISRS and ramta antenna becomes in the state of ISTS in the pair. A communication process takes place between the members of the pair continuously this way, until the process of communication between tamra and ramta antenna system is over.

DTS-17: KRA[n] alphabet

Consider Kip[n] as an alphabet. An alphabet obtained by changing the numeral symbols in Kip[n] by some tic symbols or pictures is called a KRA[n] alphabet. A KRA[n] alphabet is in the same equivalence class of Kip[n]. The other properties are also exactly same as Kip[n].

Example 9: $\mathcal{A}$ = {t ≡ "(", r ≡ ")", @, c, x} is a KRA [5] alphabet ≡ $\mathcal{B}$ = {"(", x, @, c, ")"} TASIM alphabet. If alphabet $\mathcal{B}$ of TASIM is substituted by alphabet $\mathcal{A}$ then the formal language L obtained by this substitution on $\mathcal{A}$ is called a TASIM genetic language.

DTS-18: TA antenna system

TA = TA[n] = "$_0$t $_1$t …$_k$t…$_{n-2}$t $_{n-1}$t" construction is called a T genetic, communication partner antenna system. It is designed for communicating in formal language L [K, G] on the alphabet of K = K[n] = Kip[n] for communicating in n different channels. The tic elements of TA = TA[n] is realized by an "IDC: Image Dress Code" of 0011. Observe that the tic elements of TA = TA[n] is countable in the RC-FRCC[n].

DTS-19: RA antenna system

RA = RA[n] = "r_{n-1} r_{n-2} …r_k… r_1 r_0" KBO construction is called a T genetic, communication partner antenna system. It is designed for communicating in formal language on the alphabet of K = K[n] = Kip[n] for communicating L[K, G] in n different channels. The tic elements of RA = RA[n] is realized by an IDC of 0011. Observe that the tic elements of RA = RA[n] is countable in the RC-BRCC[n].

DTS-20: TASIM genetic Q swarm memory

1) "c" is a tic constant representation. This tic constant c can be a tic KBO of tic KBOs. It is realized by an IDC of 0010.

2) "x" or "y" is a tic variable representation. Each x or y can be a tic KBO of KBOs. Both are realized by an IDC of 0010.

3) R = R[n] = "$_0$t $_1$t …$_k$t…$_{n-2}$t $_{n-1}$t x" KBO construction is called a **rator R** controlled by controlling variable x. Where, x has a capacity to sense, receive and transmit any tic information coded into any signal "k" while k □ Kip[n].

4) "@cr_k" is a united tic that it is constructed from 3 different tics of tic @, tic c and tic r. They are realized by a common IDC = 0011. Where, $@_k$ c_k r_k = @cr_k. This construction is solely called as a c-union. A c-union uses only one leg component in this article. Where, they are appearing to share one leg together.

5) CR = CR[n] = " "@cr_k" "kip-n" = " "@cr_{n-1}" "@c r_{n-2}" … "@cr_k" … "@cr_1" " @cr_0" " KBO construction is called a **rand CR**. Where, each tic c has a capacity to sense, receive and transmit any tic information coded into any signal "k" sent by x using a signal carrier k □ K = K[n] = Kip[n] to r_k.

6) Q = Q[x, n] = "$_0$t $_1$t … $_k$t… $_{n-2}$t $_{n-1}$t x @cr_{n-1} @cr_{n-2}… $_{@c}r_k$…@cr_1 @cr_0" KBO construction is a **swarm memory**. It is shortly represented in this paper by Q = P_1 Q. Observe that the pair of antennas of Q is countable in a RC-RMCC[n].

DTS-21: TASIM genetic smooth complex Q swarm memory

$Q = Q[<x, y>, <n, <n \times n>>]$ = “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{n-2}t\ {}_{n-1}t$ x @ “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{n-2}t\ {}_{n-1}t$ y $@cr_{n-1}\ @cr_{n-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” r_{n-1} @ “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{n-2}t\ {}_{n-1}t$ y $@cr_{n-1}\ @cr^{n-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” $r_{n-2} \ldots$ @ “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{n-2}t\ {}_{n-1}t$ y $@cr_{n-1}\ @cr_{n-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” $r_k \ldots$ @ “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{n-2}t\ {}_{n-1}t$ y $@cr_{n-1}\ @cr_{n-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” r_1 @ “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{n-2}t\ {}_{n-1}t$ y $@cr_{n-1}\ @cr_{n-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” r_0” KBO construction is a **smooth complex swarm memory**. It is shortly represented in this paper by $Q = P_2\ Q$. Observe that the antennas appearing in Q is countable in the SCRMCC $[n, <n \times n>]$.

DTS-22: TASIM genetic mixed complex Q swarm memory

$Q = Q[<x, y>, <m, <n_0, n_1, \ldots, n_k, \ldots, n_{m-2}, n_{m-1}>>]$ = “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{m-2}t\ {}_{m-1}t$ x @“${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<n0>-2}t\ {}_{<n0>-1}t$ y $@cr_{<n0>-1}\ @cr_{<n0>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” r_{m-1} @“${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<n1>-2}t\ {}_{<n1>-1}t$ y $@cr_{<n1>-1}\ @cr_{<n1>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$”$r_{m-2} \ldots$ @“${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<nk>-2}t\ {}_{<n2>-1}t$ y $@cr_{<nk>-1}\ @cr_{<nk>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” $r_k \ldots$ @“${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<nm-2>-2}t\ {}_{<nm-2>-1}t$ y $@cr_{<nm-2>-1}\ @cr_{<nm-2>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” r_1 @“${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<nm-1>-2}t\ {}_{<nm-1>-1}t$ y $@cr_{<nm-1>-1}\ @cr_{<nm-1>-2} \ldots @cr_k \ldots @cr_1$ $@cr_0$”r_0” KBO construction is a **mixed complex swarm memory**. It is shortly represented in this paper by $Q = P_2\ Q$.

Observe that the antennas appearing in Q is countable in the MCRMCC $[<m, n = < n_0, n_1, \ldots n_k \ldots n_{m-2}, n_{m-1}>>]$. Remember that:

$Q = Q[x, m]$ = “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{m-2}t\ {}_{m-1}t$ x $@cr_{m-1}\ @cr_{m-2} \ldots @cr_k \ldots @cr_1\ @cr_0$”,

$Q = Q[y, n_0]$ = “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<n0>-2}t\ {}_{<n0>-1}t$ y $@cr_{<n0>-1}\ @cr_{<n0>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$”,

$Q = Q[y, n_1]$ = “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<n1>-2}t\ {}_{<n1>-1}t$ y $@cr_{<n1>-1}\ @cr_{<n1>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$”,

…

$Q = Q[y, n_k]$ = “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<nk>-2}t\ {}_{<n2>-1}t$ y $@cr_{<nk>-1}\ @cr_{<nk>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$”,

…

$Q = Q[y, n_{m-2}]$ = “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<nm-2>-2}t\ {}_{<nm-2>-1}t$ y $@cr_{<nm-2>-1}\ @cr_{<n0>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$” and

$Q = Q[y, n_{m-1}]$ = “${}_0t\ {}_1t \ldots {}_kt \ldots {}_{<nm-1>-2}t\ {}_{<nm-1>-1}t$ y $@cr_{<nm-1>-1}\ @cr_{<nm-1>-2} \ldots @cr_k \ldots @cr_1\ @cr_0$”. They are used as sub structures in the construction of $Q = Q[<x, y>, <m, <n_0, n_1 \ldots n_k \ldots n_{m-2}, n_{m-1}>>]$.

Observe also that the antennas appearing in Q is countable in the MCRMCC $[<m, n = <n_0, n_1, n_2 \ldots n_{m-1}>]$.

DTS-23: Examples

Example 10: In a communicating T genetic Q swarm memory constructing Environment:

1) $Q = Q[x, 4]$ = “${}_0t\ {}_1t\ {}_2t\ {}_3t$ x $@cr_3@cr_2@cr_1@cr_0$” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in RC-RMCC [4] in this construction.

2) $Q = Q[y, 5]$ = “${}_0t\ {}_1t\ {}_2t\ {}_3t\ {}_4t$ y $@cr_4@cr_3@cr_2@cr_1@cr_0$” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in RC-RMCC [5] in this construction.

3) $Q = Q[y, 8]$ = “${}_0t\ {}_1t\ {}_2t\ {}_3t\ {}_4t\ {}_5t\ {}_6t\ {}_7t$ y $@cr_7@cr_6@cr_5\ @cr_4@cr_3@cr_2@cr_1@cr_0$” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in RC-RMCC [8] in this construction.

4) $Q = Q[x, 3]$ = “${}_0t\ {}_1t\ {}_2t$ x $@cr_2@cr_1@cr_0$” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in RC-RMCC [3] in this construction.

5) $Q = Q[y, 2]$ = “${}_0t\ {}_1t$ y $@cr_1@cr_0$” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in RC-RMCC [2] in this construction.

DTS-24: Examples

Example 11: In a communicating T genetic Q swarm memory constructing Environment:

1) $Q = Q[y, 3]$ = “${}_0t\ {}_1t\ {}_2t$ y $@cr_2@cr_1@cr_0$” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in RC-RMCC [3] in this construction.

2) $Q = Q[<x, y>, <4, <2, 2, 2, 2>>]$ = “${}_0t\ {}_1t\ {}_2t\ {}_3t$ x @“${}_0t\ {}_1t$ y $@cr_1@cr_0$” r_3@“${}_0t\ {}_1t$ y $@cr_1@cr_0$” r_2@“${}_0t\ {}_1t$ y $@cr_1@cr_0$” r_1@“${}_0t\ {}_1t$ y $@cr_1@cr_0$” r_0” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in SCRMCC $[4, <2, 2, 2, 2>]$ in this construction.

3) $Q = Q[<x, y>, <3, <3, 3, 3>>]$ = “${}_0t\ {}_1t\ {}_2t$ x @“${}_0t\ {}_1t\ {}_2t$ y $@cr_2@cr_1@cr_0$” r_2@“${}_0t\ {}_1t\ {}_2t$ y $@cr_2@cr_1@cr_0$” r_1@“${}_0t\ {}_1t\ {}_2t$ y $@cr_2@cr_1@cr_0$” r_0” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in SCRMCC $[3, <3, 3, 3>]$ in this construction.

4) $Q = Q[<x, y>, <4, <5, 8, 8, 3>>]$ =“${}_0t\ {}_1t\ {}_2t\ {}_3t$ x @“${}_0t\ {}_1t\ {}_2t\ {}_3t\ {}_4t$ y $@cr_4@cr_3@cr_2@cr_1@cr_0$” r_3@“${}_0t\ {}_1t\ {}_2t\ {}_3t\ {}_4t\ {}_5t\ {}_6t\ {}_7t$ y $@cr_7@cr_6@cr_5\ @cr_4@cr_3@cr_2@cr_1@cr_0$” r_2@“${}_0t\ {}_1t\ {}_2t\ {}_3t\ {}_4t\ {}_5t\ {}_6t\ {}_7t$ y $@cr_7@cr_6@cr_5\ @cr_4@cr_3@cr_2@cr_1@cr_0$” r_1@“${}_0t\ {}_1t\ {}_2t$ y $@cr_2@cr_1@cr_0$” r_0” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in MCRMCC $[<4, <5, 8, 8, 3>>]$ in this construction.

5) $Q = Q[<x, y>, <3, <2, 3, 5>>]$ = “${}_0t\ {}_1t\ {}_2t$ x @“${}_0t\ {}_1t$ y $@cr_1@cr_0$” r_2@“${}_0t\ {}_1t\ {}_2t$ y $@cr_2@cr_1@cr_0$” r_1@“${}_0t\ {}_1t\ {}_2t\ {}_3t\ {}_4t$ y $@cr_4@cr_3@cr_2@cr_1@cr_0$” r_0” is a swarm memory construction. Its communicating t: tamra and r: ramta antennas are counted in MCRMCC $[<3, <2, 3, 5 >>]$ in this construction.

IV. RESULT AND SUGGESTION

The notion of cycle counting for marking the same frequencies on a pair of tamra and ramta antennas appearing in Q of Q^+ has been studied. It is found that:

(a) $Q = P_1\ Q$ with property $P_1 = <C, T; K_n, x, n; 4B \in IDC: *> Q$ can use FRCC [x, n], BRCC[x, n], FBRCC[x, n] and RMCC[x, n] cycle counting systems for marking the pair of tamra and ramta antenna systems being in the communication with each other while information processing taking place in a Q of Q^+.

(b) $Q = P_2\ Q$ with property $P_2 = <C, T; <K_m, K_n>, <x, y>, <m, <n_0, n_1, \ldots, n_k, \ldots, n_{m-2}, n_{m-1}>>; 4B \in IDC>Q$ can use SCRMCC $[<x, y>, <n, <n, n, \ldots, n>>]$ and MCRMCC $[<x, y>, <m, <n_0, n_1, \ldots, n_k, \ldots, n_{m-2}, n_{m-1}>>]$

cycle counting systems for marking the pair of tamra and ramta antenna systems being in communication with each other while information processing taking place in a Q of Q^+. Therefore this paper is written for reporting and introducing the new found cycle counting processes to mark proper frequencies to be used by the communicating antenna resource elements while an information processing taking place in an I@I of $I@I^+$ swarm internet "+"closure generating Q of Q^+ swarm memory "+" closure communicating in every K = [n] = Kip[n]. The author is declaring that this internet closure is a securely communicating universal internet closure.

REFERENCES

[01] F. Ünlü: CITALOG: Compact and Integrated Tasim Logic Closure, *Journal of King Abdulaziz University, Science, Vol. 2 pp 117-136, Jeddah, 1990.*

[02] F. Ünlü: FTD Grammar Graph, *International Journal of Computer Mathematics,* 2003, Vol. 80(1) pp1-9.

[03] F. Ünlü: us-Crop Based Compact Memory, *International Journal of Contemporary Mathematical Science, Volume 1, No. 5-8, pp 317-325, 2006.*

[04] F. Ünlü: A Remote Programming Technology on a Remote VDM Clustering in λ-Calculus, *International Mathematical Form, Vol. 1, No. 13-16, pp 671 - 685, 2006 .*

[05] F. Ünlü: Plemvanal: A Communicating Commuting Mathematics Generator Type, *International Mathematical Form, Vol. 1, No. 13-16, pp 671 – 685, 2006* .

[06] F. Ünlü: T-genetic RCR-U Form, *INISTA 2010, ID 16, pp 1-5, Kayseri, Turkey, 2010.*

[07] F. Ünlü: Biçimsel TASIM Dilbilimi, AYSU 2010, ID 5, pp 1-5, Kayseri, 2010.

[08] F. Ünlü: A Formel Property Dependent United Business System Design and Programming, ICBME 7-9 October 2010 e-Proceeding, Yasar University, Izmir, Turkey.

[09] F. Ünlü: Bildirişimli Matematik, 9. Matematik Sempozyumu, 20-22 Ekim 2010 Karadeniz Teknik Üniversitesi, Trabzon.

[10] F. Ünlü: Q ve Q^+ Sürü Bellekli I@I Modeli Tasarımında Kullanılan T Genetik Altyapı Elamanları, *XVI. Türkiye 'de İnternet Konferansı. Bildiri No. 3, 30 Kasım-2 Aralık 2011, Ege Universitesi, İzmir.*

[11] F. Ünlü: Bildirişimli Matematiğin <T<x>, 1, n> Q Sürü Bellekli 3D I@I İnternet Sürüsü. *XVI. Türkiye'de İnternet Konferansı. Bildiri No. 4, 30 Kasım -2 Aralık 2011, Ege Universitesi, İzmir.*

[12] P. Linz, *Introduction to Formal Languages and Automata,* Fourth Edition, Jones and Bartlett Publishers, London, 2006.

[13] R. L. Causey, *Logic, Sets, and Recursion,* Second Edition, Jones and Bartlett Publishers, Inc., London, 2006.

[14] W. Lark, *LISP 1.5 Primer,* Dickenson Publishing Inc. California, 1968.