The Recursive Transpose-Connected Cycles (RTCC) Interconnection Network for Multiprocessors

M. Hoseiny Farahabady\textsuperscript{1, 2}, H. Sarbazi-Azad\textsuperscript{2, 1}

\textsuperscript{1} IPM School of Computer Science, Tehran, Iran.
\textsuperscript{2} Department of Computer Engineering of Sharif University of Technology, Tehran, Iran.

Hoseiny@{ipm.ir, ce.sharif.edu}, Azad@{ipm.ir, sharif.edu}

ABSTRACT

In this paper, we propose a new modular topology for interconnection networks, the Recursive Transpose-Connected Cycles (RTCC). The RTCC has a recursive definition quite similar to that of fractal graphs having interesting topological characteristics, making it suitable for utilization as the base topology of large-scale multicomputer interconnection networks. We study important properties of this topology such as diameter, bisection width and issues related to implementation, such as routing algorithms and the average message latency under VLSI layout constraints. In addition, we prove that the RTCC is a Hamiltonian graph. We conclude that, insight of most of the above-mentioned properties, the RTCC is superior to conventional topologies such as the mesh and k-ary n-cube.

Keywords

Multicomputer, Interconnection network, Hamiltonian properties, Performance evaluation, Implementation constraints.

1. INTRODUCTION

An interconnection network can be modeled by an undirected graph in which a processor is represented by a node, and a communication channel between two nodes is represented by an edge between corresponding nodes. The tree, mesh (especially the 2-dimensional mesh or $M_{n\times n}$), torus, hypercube, k-ary n-cube, star graphs [1], and chordal rings [2] (cycles in which chords have been added) and pruning networks (in which certain links are removed from the $K_n$ network such that overall implementation cost is reduced) are examples of common interconnection network topologies. Desirable properties of interconnection networks include symmetry, small node degree, diameter, bisection cost (the product of node degree and network diameter, $dD$) and bisection width $B$, and high connectivity, scalability, modularity, and fault tolerance. The most straightforward physical connection pattern for the nodes of a multiprocessor system is that of an n-node ring, $R_n$ (or cycle $C_n$). In such a setting, each node has only two communication links, one to either of the two adjacent nodes.

Many efficient algorithms for traditional networks mentioned above have been developed and important topological properties derived. However, a major deficiency of these networks is that they are not truly expansible. A network can be defined as expansible if no changes with respect to node configuration and link connection are necessary when it is expanded. The present paper introduces a new topology, the definition of which is based on the recursive connection of cyclic networks. This network preserves almost all of the desirable properties of mesh networks and displays even better topological properties in some cases. For example, while being of low diameter, this network possesses a small number of edges. In addition, a mesh network can easily be embedded onto this network. In the following sections, we illustrate a number of these properties. This paper is organized as follows. In Section 2, a new interconnection network topology is proposed and various topological properties, such as network bisection width, are studied. A routing algorithm is suggested and with respect to this algorithm, an upper bound for the average network distance is determined. In Section 3, proof is presented of the fact that this network is Hamiltonian. A comparison of this network and other popular networks is presented in Section 4. Finally, Section 5 concludes this study.

2. THE RTCC NETWORK

In this section, we formally define the RTCC and derive some of its topological properties.

2.1 Definition and Topological Properties

An L-level RTCC, denoted as $RTCC(C, L)$, is recursively constructed by connecting a number of C distinct $RTCC(C, L-1)$ networks such that each $RTCC(C, L-1)$ is directly connected to a node of every other $RTCC(C, L-1)$. Note that the $RTCC(C, 1)$ is a C-node cycle.

Definition 1: A C-node cycle consists of a set of nodes \{0, 1, 2, 3… $C-1$\} and set of edges \{e\}_0, e\_1, e\_2… e\_C-1\} such that e\_i = (i, i+1 mod C).

Definition 2: The definition of the $RTCC(C, L)$ is based on a C-node cycle. We name all the nodes in this cycle, ‘Extern nodes’ or ‘Open nodes’. An $RTCC(C, 2)$, consists of a number of C discrete $RTCC(1, 1)$ networks, or C-node cycles, numbered 0 to C-1. Each external node i of each C-node cycle j is connected to node j of C-node cycle i. It is obvious that a node whose number is equal to the number of the $RTCC(C, 2)$ in which it resides, is not directly connected to any other cycle, and is of...
node degree one less than other nodes in the network. There is one such node in each RTCC(C, 1) used to construct an RTCC(C, 2), and thus a total of C such nodes. We name these nodes as the external nodes of the RTCC(C, 2). In a similar manner, the RTCC(C, L) can be defined as C discrete RTCC(C, L-1) networks that are connected in such a way that each external node i in RTCC(C, L-1) number j is connected to external node j in RTCC(C, L-1) number i. Once again, a node whose number is equal to the address of the RTCC(C, L-1) in which it resides, is not directly connected to any other cycle, and is of node degree one less than other nodes in the network. We name these nodes, of which there is a total of C, the external nodes of the RTCC(C, L). In Figure 1, the connection between nodes of an RTCC (4, 2), RTCC (5, 3) and an RTCC (6, 1) are displayed.

We now focus on the topological properties of this network. The node set of an RTCC(C, L) can be expressed as \{ (a_1, a_2, …, a_{L-1}) \} a_i ∈ γ, 1 ≤ i ≤ L, where γ = {0, 1, …, C-1}. Therefore, the number of nodes of an RTCC(C, L) network is \( C × |RTCC(C, L)| = C × |RTCC(C, L-1)| + (C-1)/2 = (3^L - C) / 2 \). The degree of external nodes of an RTCC(C, L) is 2 and the degree of all other nodes is 3. Therefore, the degree of network is fixed and equal to 3. The address of the neighbors of a node with address (a_1, a_2, …, a_{L-1}) are: 

1. \((a_1, a_2, …, (a_{L-1}+1) mod C)\)
   and \((a_1, a_2, …, (a_{L-1}-1) mod C)\) in the same lowest-level sub-graph, which we refer to as sister nodes,
2. \((a_1, a_2, …, a_{j-1}+1, a_{j+1}, …, a_{L-1})\), 1 ≤ j ≤ L-1, where a_j = a_{j+1} = … = a_{L-1} ≠ a_{j-1}, corresponding to the connection between the external nodes of level-j sub-graphs. Notation \((a')\) denotes j consecutive a’s. We refer to this adjacent node as a cousin node. It is obvious that a node whose address is of the form \((a_1)\) \(L-1\), i.e. the address of all sub-graphs to which the node belongs to are the same, is of no cousin. We refer these nodes as external node.

2.2 Routing Algorithm in the RTCC

A recursive routing algorithm can be defined for routing a message from node S = (a_1, a_2, …, a_{L-1}) to D = (b_1, b_2, …, b_L) in an RTCC(C, L) network. When a_i = b_i, the source and destination nodes belong to the same sub-graph. In this case, the same routing algorithm as that used to route the message from (a_1, a_2, …, a_{L-1}) to (b_1, b_2, …, b_L) in the RTCC(C, L-1) can be utilized. If however a_i ≠ b_i, the message is recursively routed from S to the node S' = (a_1, b_1, …, b_L) within the source RTCC(C, L-1) sub-graph. It is then routed, over a single link, from S' to D' = (b_1, a_1, …, a_{L-1}), the cousin of S'. Finally from D', which resides within the destination sub-graph, the message is routed to D. At the lowest level (where L=1), the same routing algorithm used for routing within a C-node cycle can be utilized. This algorithm does not necessarily result in a shortest routing path between two nodes, as the length d(S, S') + d(D, D') + 1 may be longer than another alternative path. In Fig. 1, for example, the route R_1 traversed by this algorithm is longer than route R_2. The definition of the mentioned algorithm is however straightforward and can easily be implemented.

Lemma 1: The diameter of an RTCC(C, L) is \(2^{L-1} × \lceil C/2 \rceil + 2^{L-2} + 1\).

Proof: We first show that the two most distant nodes in an RTCC(C, L) network correspond to addresses A=(α)\(^L\) and B=(β)\(^L\), where |α-β| = |C/2| and 1 ≤ α, β ≤ C. We prove this by induction on L. When L=1, this proposition is true as the maximum distance in a C-node cycle obviously occurs between two nodes of the cycle that fulfill the above-mentioned condition. Assuming this proposition to be true for L=ℓ+1, we prove that it is also true for L=ℓ+1. According the mentioned routing algorithm, the distance between nodes A and B is determined as one plus the sum of the distance between A=(α)\(^{L-1}\) and A'=(α (β))\(^L\), and the distance between B'=(β)\(^L\) and B=(β)\(^L\). However, the distance between A and A' within sub-graph a is the maximum distance in an RTCC(C, L-1), and so is the distance between B and B'. Since the routing algorithm (either the proposed or any other efficient one) does not route a message through more than two RTCC(C, L-1) sub-graphs, the distance between nodes A and B is the maximum of an RTCC(C, L). As a result, the diameter of RTCC(C, L) is \(2^{L-1} × \lceil C/2 \rceil + 2^{L-2} + 1\).

Lemma 2: The average node distance in RTCC(C, L) is less than \(2^{L-2} × \lceil C/2 \rceil\).

Proof: With induction on L. The RTCC(C, 1) is a cycle and as we know, the average distance in a cycle is approximately C/4. We denote two arbitrary nodes in the network as u=(u_1, u_2, …, u_L) and v=(v_1, v_2, …, v_L). When u_1 ≠ v_1, the nodes that connect the related sub-graphs of these two nodes are u'=(u_2(v_2)^{L-1}) and its cousin, v'=(v_2(u_2)^{L-1}). The nodes u and v' belong to the same sub-graph, u_1 and v' belong to the v_1 sub-graph. According to our recursive routing algorithm, the distance between u and v, d(u, v), is equal to d(u, u') + d(v', v) + 1. Thus, if u and v do not belong to same sub-graph, \(d_L(u, v) = d_L(u, u') + d_L(v, v') + 1\). If, however, u_1 = v_1 then \(d_L(u, v) = d_L(u, v)\). Since the RTCC(C, L-1) is almost a symmetric network, the average distance between external and an arbitrary node, \(d_L(u, u')\), is approximately equal to \(d_L(u, z)\), where z is an arbitrary node in the sub-graph u_L. So:

\[d_L(u, u') = (d_L(u, u') + d_L(v, v')) × (C^{L-1} - C^{L-2}) + d_L(u, v) × C^{L-1}\]
\[
\leq \left( \left( 2^{1.5} \times C \right) + 2^{1.5} \times C \right) + \left( C^2 \times C \right) / C^2 \\
\leq \left( 2^{1.5} \times C \right) + \left( C^2 \times C \right) / C^2 \\
\leq \left( 2^{1.5} \times C \right) + \left( C^2 \times C \right) / C^2 \\
\leq \left( 2^{1.5} \times C \right) + C^2 / C^2.
\]

Since our routing algorithm does not necessarily result in a shortest path, the average distance between \( u \) and \( v \) is less than \( 2^{1.5} \times C / C^2 \) and the lemma is proven.

**Definition 3:** The minimum number of links must be cut in which the network divided into two equal parts is the Bisection Width [1].

**Lemma 3:** The bisection width, \( B_{RTCC(C,L)} \), of the RTCC(C, L) is equal to \( \left\lfloor \frac{C}{2} \right\rfloor \times \left\lfloor \frac{C}{2} \right\rfloor = \left( \frac{C^2}{4} \right) \), and is equal to 2 when \( L = 1 \).

**Proof:** When, in an RTCC(C,L) network, each of the RTCC(C, L-1) sub-graphs is assumed as a super-node, the whole network looks like a graph \( K_C \), which is of a bisection width equal to \( \left\lfloor \frac{C}{2} \right\rfloor \times \left\lfloor \frac{C}{2} \right\rfloor = \left( \frac{C^2}{4} \right) \). The lemma is clearly hold for \( L = 1 \).

An important result from Lemma 3 is that the bisection width of the RTCC is constant regardless of network size. This property is a valuable topological characteristic.

### 2.3 Related Work

To overcome the problem of unbounded node complexity in large hypercube and star graphs, constant-degree variants, such as the cube-connected cycles, de Bruijin graphs, butterfly networks[1], WK-Recursive networks[3], Recursive Hierarchical Swapped network[4] and OTIS networks[5], have been proposed in the literature and have been shown to possess several desirable properties.

The most similar networks to the RTCC(C, L) are the OTIS networks, the swapped networks, and WK-recursive networks. The OTIS network [5], is constructed of \( |G| \) copies (clusters) of an \(|G|\)-node nucleus or basic graph \( G \), numbered from 0 to \(|G|-1\).

**Definition 4:** A Hamiltonian path between nodes \( u \) and \( v \) in graph \( G \) is a path that leads from \( u \) to \( v \) and traverses every node of \( G \) exactly once. A Hamiltonian cycle in \( G \) is a Hamiltonian path from one node to itself. A graph, \( G \), is Hamiltonian if a Hamiltonian cycle can be found in the graph, and it is Hamiltonian-connected if a Hamiltonian path exists between every pair of nodes in \( G \). We denote a Hamiltonian cycle and path in an RTCC(C, L) as \( HC_{RTCC(C,L)} \) and \( HP_{RTCC(C,L)} \), respectively.

**Theorem 1:** The RTCC(C, L) is Hamiltonian.

**Proof:** When \( L = 1 \), the RTCC(C, L) is a \( C \)-node cycle which is Hamiltonian. When \( L = 2 \), the RTCC network is equivalent to an OTIS(C) network. The proof of an OTIS network being Hamiltonian is based on proof of the nucleus of the OTIS network being Hamiltonian [8]. When \( L \geq 2 \), if we assume that there is a Hamiltonian path between any pair of external nodes, we can show that this theorem is true. A Hamiltonian cycle in RTCC(C, L), \( HC_{RTCC(C,L)} \), is constructed according to the following sequence:

\[
HC_{RTCC(C,L)} \leq \left( 0(L-1)^{2} \right) \parallel \left( 1(L-1)^{2} \right) \parallel \left( 2(L-1)^{2} \right) \parallel \left( 3(L-1)^{2} \right) \parallel \left( 4(L-1)^{2} \right) \parallel \left( 5(L-1)^{2} \right) \parallel \left( 6(L-1)^{2} \right) \parallel \left( 7(L-1)^{2} \right) \parallel \left( 8(L-1)^{2} \right) \parallel \left( 9(L-1)^{2} \right) \parallel \left( 10(L-1)^{2} \right) >.
\]

In the notation used to describe this cycle, the symbol \( \parallel \) concatenates two set of nodes, and the \( HP \) symbol represents the Hamiltonian path between two pairs of external nodes in the mentioned sub-graph.

Now, let us prove that there exists a Hamiltonian path between any pair of external nodes in a RTCC(C, L) for any \( L \geq 2 \).

**Theorem 2:** There exists a Hamiltonian path, in the RTCC(C, L), \( L \geq 2 \), between any two external nodes.

**Proof:** We prove this by induction on \( L \). The base of induction is
when \( L=2 \). We consider two different cases, when \( C \) is odd, and when \( C \) is even.

Case 1: \( C=2S \). Without loss of generality, we accomplish this case with establishing a Hamiltonian path between nodes \((0,0)\) and \((S-j,S-j)\), \(1 \leq j \leq S \). Consider the following path.

**HP(0, 0),j\_path:** \( <(0,0)|| (2S-1,0)||(2S-1,2)||(2S-2,1)||...||(1,2)||(1,1)||...||(0,0) \). Consider networks with constant channel bandwidth. Under this assumption each channel is one bit wide (\( 1 \text{ bit} \)). There are two components, latency, the distance and message aspect ratio. The expected distance \( \bar{d} \), is the number of hops required to reach the destination from the source node. The message aspect ratio, \( M = \frac{B}{C} \), is the ratio of the number of channel cycles required to transmit the message across one channel. The following formulas show the \( \bar{d} \) for packet and wormhole switching. (Average message latency is reduced in wormhole as a result of its pipeline nature.)

When \( L \geq 2 \). Let the theorem be true for \( L \leq 1 \). So, we construct a Hamiltonian path between any given external nodes in \( RTCC(C, L) \). Without loss of generality, assume given nodes are \((0^{(-1)}, \text{ and } (j_{+1}) \). The following path is desirable path between \((0^{(-1)}, \text{ and } (j_{+1}) \) which contain all of nodes in the \( RTCC(C, L) \).

**HPRTCC(C,L):** \(<(0^{(-1)}),\text{HPRTCC(C,L) in sub-graph 0}, ..., (0^{(-1)}), \text{HPRTCC(C,L) in sub-graph 1}, ..., (0^{(-1)}), \text{HPRTCC(C,L) in sub-graph j}, ...\). Consider the performance of these networks under constraints such as equal channel and bisection bandwidth, and pin-out. These restrictions are of importance in the implementation of such interconnection networks. When systems are implemented on a single VLSI-chip, the wiring density of the network determines the overall cost and performance [6]. For instance, Dally [7] has used the bisection width as a rough measure of the network wiring density in a pure VLSI implementation. Due to the limitation on channel bandwidth, imposed by implementation technology, a message is broken into channel words (or phits), each of which is transferred in one cycle. Let the channel width, \( C \), be \( C \) bits, and thus a message of \( B \) bits is divided into \( M = \frac{B}{C} \) phits [6] for transmission over a channel. We consider networks that use wormhole switching or store-and-forward switching [1].

The advantage of wormhole technique compared to store-and-forward is that it reduces impact of message distance on the latency under light traffic. There are two components of latency, the distance and message aspect ratio. The expected distance \( \bar{d} \), is the number of hops required to reach the destination from the source node. The message aspect ratio, \( M = \frac{B}{C} \), is the ratio of the number of channel cycles required to transmit the message across one channel. The following formulas show the \( \bar{d} \) for packet and wormhole switching. (Average message latency is reduced in wormhole as a result of its pipeline nature.)

\[
T_{store \ & \ forward} = T_c \times \bar{d} \times M. \quad (1)
\]

\[
T_{wormhole} = T_c \times (\bar{d} + M). \quad (2)
\]

In above equations, the \( T_c \) denotes the channel cycle time, i.e. the amount of time required to send a flit over a channel [12]. Networks have traditionally been analyzed under the assumption of constant channel bandwidth. Under this assumption each channel is one bit wide (\( 1 \text{ bit} \)) and has unit delay (\( T_c = 1 \)). However, the constant bandwidth assumption is not considered with the properties of VLSI technology. When implementing networks, high degree networks require more and longer wires than do low-degree networks. Thus, high-degree networks cost more and run more slowly than low-degree networks when physical implementation constraints are taken into account. We will use bisection bandwidth as a measure of network implementation cost. Thus, we compare networks with constant bisection bandwidth. Assuming equal bisection bandwidth for the two same-sized networks, we can calculate the ratio of their channel cycle times. Lets \( BBW_1 \) and \( BBW_2 \) be the bisection bandwidth of Network 1 and Network 2, respectively. Let \( BBW_1 \) and \( BBW_2 \) be the bisection width of the considered networks. So we have \( BBW_1 = BBW_2 \), resulting in:

\[
BBW_1 \times C_{u1} = BBW_2 \times C_{u2} \quad (3)
\]

And the ratio of channel cycle times of these networks is given by:

\[
\frac{T_{c1}}{T_{c2}} = C_{u1}/C_{u2} = BBW_1/BBW_2 \quad (4)
\]

Considering \( T_{cRTCC} = 1 \) unit cycle, we calculate the channel
Figure 3. A comparison of the average message latency in the RTCC and popular networks (a) wormhole switching (b) packet switching under the constraint of equal bisection bandwidth. (c) Wormhole switching under the constraint of equal pin out.

cycle time for any other network to be compared to the given RTCC network. We assume, for the sake of presentation, that message length is fixed and equal to 32 flits. Figure 3(a) shows the average message latency of wormhole routed hypercube, 2-D mesh, k-ary n-cube and RTCC(C, L) networks. The calculations are realized using Eq. 2 for the four considered networks, with their corresponding average inter-node distances and channel cycle times. Figure 3(b) shows the average message latency curves for these networks when packet switching is used. As can be observed, the RTCC(C, L) network has much less delay than other networks for both switching methods. The cause for this stems from the fact that the bisection width of the RTCC network remains constant as network size increases. This is while, in multiple-chip implementations, where a complete node is fabricated on a single chip, the number of I/O pins (i.e. node degree × channel width), or the pin-out, is a more suitable metric of implementation cost [6]. When the constant pin-out constraint is considered, we have Pinout1=Pinout2 resulting in:

\[ C_{w1} = \text{deg} \times C_{w2} \times \text{deg} \times C_{w2} \]

So the channel cycle times for the networks to be considered under constant pin-out constraint can be related as:

\[ T_{C1}/T_{C2} = C_{w1}/C_{w2} \times \text{deg}/\text{deg} \]

Comparison of the average latency of wormhole and packet switching in the above mentioned networks, under constant pin-out, are shown in Figure 3(c) for different network sizes. It is clear that the performance of the RTCC(C, L) network is superior to that of other famous networks. Assuming the packet switching method is even more in favor of the RTCC network, for which we have not shown the curves for the sake of brevity. It is also evident from the results that the larger C is, the better the performance of the RTCC network is.

5. CONCLUSIONS

Many network topologies have been proposed for multi-computer interconnection networks in the past. In this paper, we introduced a new interconnection topology based on cycles with a recursive definition. The low degree and bisection width of the proposed network make it a good cost-effective candidate for interconnecting processing nodes in a multi-computer.

Some important properties and performance issues of the RTCC were investigated here. The results show that the performance of the RTCC is superior to conventional topologies such as the mesh and k-ary n-cube when VLSI implementation constraints are taken into account.

The future work, in this line, includes a thorough performance evaluation of the RTCC using extensive simulation experiments for different working conditions.

6. REFERENCES