# Vertex-Transitivity and Routing for Cayley Graphs in GCR Representations

K. Wendy Tang* and Bruce W. Arden†

*Department of Electrical Engineering
SUNY at Stony Brook, Stony Brook, NY 11794.

†Department of Electrical Engineering
University of Rochester, Rochester NY 14627.

## Abstract

Dense, symmetric graphs are good candidates for effective interconnection networks. Cayley graphs, formed by Borel subgroups, are the densest, symmetric graphs known for a range of diameters [1]. Every Cayley graph can be represented with integer node labels by transforming into another existing topology, Generalized Chordal Ring (GCR) [2]. However, generally speaking, GCR graphs are not fully symmetric. In this paper, we provide a framework for the formulation of the complete symmetry (or vertex-transitivity) of Cayley graphs in the integer domain of GCR representations. Successful realization of such formulation offers a simple, iterative routing algorithm that is capable of determining multiple, shortest paths between any source and destination pairs. An example from a Borel Cayley graph is used to illustrate this concept.

## Introduction

*Multiprocessors* and *multicomputers* are two major categories of parallel computers [3]. In the former, processors communicate via shared memory whereas in the latter, each processor has its own local memory (hence a computer) and communication is via message passing. Whether it is a shared-memory multiprocessor or a message-passing multicomputer, an efficient *interconnection network* to interconnect the communicating elements is critical to the performance of the parallel computer [4]. In the design of an interconnection network, there are two major issues: the interconnection *topology* and related *routing algorithms*.

An interconnection topology can be modeled as a graph. To model a multicomputer system, we consider *regular*, *undirected* graphs with no *multiple edges* between any pair of nodes. A graph is *regular* when it has the same number of incident edges, or *degree*, at every node [5]. Nodes of the graph correspond to processors with local memory and the edges represent connections between these elements. Due to the limited number of connections that can be made to real chips, we are interested primarily in regular graphs of small degree. For a given small degree, we are interested in *dense graphs* [6]. A *dense graph* is one with a large number of nodes for a given *diameter*. The diameter is the maximum *distance* between all node pairs. Here *distance* between two nodes refers to the smallest number of hops between the two nodes. Obviously, a dense graph allows the interconnection of a large number of processing elements with relatively small communication delay.



Besides density, *symmetry* or *vertex-transitivity* is another desirable attribute of an efficient interconnection network topology [7]. Informally, a symmetric or vertex-transitive graph looks the same from any node [8]. This property allows the use of identical routing algorithms at every node. The goal of *routing* is to send messages between pairs of nodes. There are two aspects. First, we have to identify a path between non-adjacent nodes. Second, there is the problem of conflicts when multiple messages at a node have the same optimal outgoing link. In this paper, we discuss the first aspect of routing: path identification.

Path identification is a trivial problem for graphs with *path-defining* labels that implicitly define shortest paths between vertices. In this case, *optimal routing* or *shortest-path identification* can be achieved computationally with an algorithm that has a space requirement independent of graph size, i.e., its space complexity is O(1). The toroidal mesh [9], hypercube [10] and cube-connected cycles [11] are examples of such graphs. Clearly, these graphs have more efficient routing algorithm than graphs without path-defining labels. However, for massively parallel computer systems, with thousands of processing elements, density is an important factor; and unfortunately these graphs are far from the densest.

Among the many symmetric interconnections being proposed, *Cayley graphs* are attractive because a subclass, *Borel Cayley graphs*, are the densest known degree-4 graphs for a range of diameters [1]. However, these graphs do not have path-defining labels. They are defined over a group of matrices, the Borel matrices. That is, vertices are labeled as matrices. There is no inherent, simple ordering of node labels and no known computational routing algorithm with a constant or O(1) space commitment.

For graphs without path-defining labels, routing can be done by tables [7]. At every node, a table of size $(n-1)$ is stored to record the optimal out-going links from that node to all other nodes in the network. In the general case, these *routing tables* differ at each node. Furthermore, routing is achieved in a progressive, step at-a-time approach because only the immediate next node is identified at any node in the path. However, it should be possible for symmetric networks, such as Cayley graphs, to use an identical routing table at every node, and hence an entire path could be obtained at any node.

In an earlier report[2], we have proved that all Cayley graphs can be represented in the integer domain as Generalized Chordal Rings (GCR). The GCR representation is isomorphic to and retains all the properties of the original Cayley graphs. However, as its definition (see section 2) indicates, generally speaking, a GCR graph is only partially symmetric. In this paper, we provide a framework to express the fully symmetric or vertex-transitive property of Cayley graphs in GCR representations. In the case of Borel Cayley graphs, this framework is further simplified into a single equation. We then exploit this inherent symmetry of Cayley graphs to develop an *optimal* routing algorithm, *Vertex-Transitive routing*, that uses identical routing tables to determine an entire path at any node.

This paper is organized as follows: In section 2, we review the definitions of GCR, Cayley graphs and Borel Cayley graphs. In section 3, we discuss the vertex-transitive property of Cayley graphs in GCR representations. Section 4 discusses the special case, Borel Cayley graphs. Section 5 consists of a routing algorithm that exploits the vertex-transitivity of Cayley graphs in the GCR domain. A Borel Cayley graph is then used to demonstrate the routing algorithm. Finally in section 6, we summarize and conclude this paper.

## Review

In this section we review the definitions of Generalized Chordal Rings (GCR) [12], Cayley graphs in general and Borel Cayley graphs in particular [13] [1].

### GCR

**Definition 1** *A graph* **R** *is a Generalized Chordal Ring (GCR) if vertices of* **R** *can be labeled with integers* $\{0, \ldots, n-1\}$*, and there is a divisor of* $n$*, say* $q$*, such that vertex* $i$ *is connected to vertex* $j$ *iff vertex* $i+q$ *(mod* $n$*) is connected to vertex* $j+q$ *(mod* $n$*), where* $n$ *is the number of vertices.*

According to this definition, vertices of a GCR are labeled from 0 to $n-1$ and are classified into $q$ classes, each class with $n/q$ elements. The classification is based on modulo $q$ arithmetic. Two vertices having the same residue (mod $q$) are considered to be in the same class. That is, class $i$ consists of the following nodes: $i$, $i+q$, $i+2q$, $\ldots$, $i+mq$ (mod $p$), where $m = n/q - 1$; and node $i$ is the *representing element* of class $i$. Since $i$ connects to $j$ implies $i+q$ connects to $j+q$ (mod $n$), nodes in the same class have the same connection rules. In other words, connections of the entire graph is defined if connections of each class representing element are known.

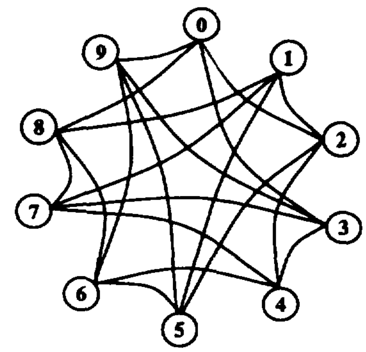


Fig. 1. *A Degree-4 GCR ($n = 10$ $q = 2$)*

The connection rules of each class representing element are defined by *connection constants*, the GCR constants. For degree 4 graphs, there are four GCR constants in each class, defining the 4 neighbors. For example, Figure 1 shows a degree 4 GCR with 10 nodes and $q = 2$ classes.

The connection rules for these classes can be defined as: Let $\mathbf{V} = \{0, 1, \ldots, 9\}$. For any $i \in \mathbf{V}$, if $i$ mod 2 =:

"0" : i is connected to $i+2$, $i+3$, $i-1$, $i-2$ (mod 10);
"1" : i is connected to $i+1$, $i+4$, $i-4$, $i-3$ (mod 10).

In this case, the vertices of the graph are numbered from 0 to 9 and are divided into even and odd classes. For the even vertices, the connection constants are $+2$, $+3$, $-1$, and $-2$; and for the odd vertices, the connection constants are $+1$, $+4$, $-4$ and $-3$. The addition of these connection constants to the node label is done in modulo $n$ arithmetic.

This class-structure of a GCR provides a *regular layout*, and a *concise* and *simple* way of describing connectivity in the integer domain and therefore making GCR an attractive representations. Furthermore, GCR are obviously symmetric within a class. Nodes of the same class have the same GCR constants and hence identical connection structure. However, symmetry between nodes of different classes is not a necessary property of a GCR. One of our contributions in this paper is to provide a framework for expressing the complete symmetry of Cayley graphs in GCR representations.

### Cayley Graphs

**Definition 2** *A graph* $\mathbf{C} = (\mathbf{V}, \mathbf{G})$ *is a Cayley graph with vertex set* $\mathbf{V}$ *if two vertices* $v_1, v_2 \in \mathbf{V}$ *are adjacent* $\Leftrightarrow v_1 = v_2 * g$ *for some* $g \in \mathbf{G}$ *where* $(\mathbf{V}, *)$ *is a finite group and* $\mathbf{G} \subset \mathbf{V} \setminus \{I\}$. $\mathbf{G}$ *is called the generator set of the graph.*

Note that the identity element $I$ is excluded from $\mathbf{G}$. This prevents the graph from having self-loops. A Cayley graph is undirected if $\mathbf{G}$ is closed under inversion, and the graph's degree is $\mid \mathbf{G} \mid$. Because of the availability of degree-4 transputer chips [14], we are interested in undirected, degree-four Cayley graphs. In other words, we are concerned with Cayley graphs whose generator set consists of two group elements $\mathbf{A}$, $\mathbf{B}$ and their inverses.

### Borel Cayley Graphs

The definition of a Cayley graph requires vertices to be elements of a group but does not specify a particular group. A family of Cayley graphs that includes some of the densest degree 4 graphs are formed from a subgroup, the Borel subgroup $\mathbf{BL}_2(\mathbf{Z}_p)$, of the general linear $2 \times 2$ matrices $\mathbf{GL}_2(\mathbf{Z}_p)$ [1]. The definition of the Borel subgroup is:

**Definition 3** *If* $\mathbf{V}$ *is a Borel subgroup,* $\mathbf{BL}_2(\mathbf{Z}_p)$, *of* $\mathbf{GL}_2(\mathbf{Z}_p)$ *with a parameter* $a$, $a \in \mathbf{Z}_p \setminus \{0, 1\}$, *then*

$$\mathbf{V} = \left\{ \begin{pmatrix} x & y \\ 0 & 1 \end{pmatrix} : x = a^t \pmod{p},\ y \in \mathbf{Z}_p,\ t \in \mathbf{Z}_k \right\}$$

*where* $p$ *is prime and* $k$ *is the smallest positive integer such that* $a^k = 1 \pmod{p}$.

The vertices of *Borel Cayley graphs* are linear $2 \times 2$ matrices that satisfy the definition of Borel subgroup, and modular matrix multiplication is chosen as the group operation $*$. Note that $n = \mid \mathbf{V} \mid = p \times k$, where $k$ is a factor of $p - 1$. By choosing specific generators, Chudnovsky et al. [1] produced Borel Cayley graphs that are the densest, nonrandom degree 4 graphs currently known for diameters 7 to 13 [1].

In our earlier report, we proved that all Cayley graphs can be transformed into GCR and provided an explicit algorithm to generate a GCR from a Cayley graph [2]. In essence, the algorithm involves choosing an arbitrary element from the group, called the the *transform elment* $T$. The well known Lagrange theorem [15] guarantees that $T^m = I \pmod{p}$ implies $n = mq$, where $I$ is the identity element, $n$ is the number of elements and $q$ is an integer. The elements of the group are then partitioned into $q$ classes by premultiplying the transform element with the representing element of each class, which is arbitrarily chosen. The following is an example of a Borel Cayley graph in a GCR representations:

### An Example

As an example, we consider the Cayley graphs over $\mathbf{BL}_2(\mathbf{Z}_p)$ with $a = 2$ and $p = 7$. Since we are interested in undirected, degree-4 graphs, there are four generators: $\mathbf{A}$, $\mathbf{B}$, $\mathbf{A}^{-1}$ and $\mathbf{B}^{-1}$. Let $\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ and $\mathbf{B} = \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}$. Note that in this case, we have $k = 3$, $p = 7$, $n = p \times k = 21$ and the diameter is 3. We arbitrarily choose the transform element $\mathbf{T} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ with $\mathbf{T}^7 = \mathbf{I} \pmod{p}$ to produce a GCR representation. Furthermore, we choose the representing element of class $i$ to be $a_i = \begin{pmatrix} a^i & 0 \\ 0 & 1 \end{pmatrix}$, $i = 0, \ldots, q-1$. Since $m = 7$, the divisor $q = n/m = 3$. Let $\mathbf{V} = \{0, 1, \ldots, 20\}$. For any $i \in \mathbf{V}$, if $i$ mod 3 =:

> "0" : i is connected to $i+3$, $i-3$, $i+4$, $i-10$ (mod 21);
> "1" : i is connected to $i+6$, $i-6$, $i+7$, $i-4$ (mod 21);
> "2" : i is connected to $i-9$, $i+9$, $i+10$, $i-7$ (mod 21).

Note that, the modular integer labels have imposed an order on the original vertices of the Cayley graph and then divided these vertices into 3 classes according to their residue (modulo 3). The connection rules of each class are summarized above. For instance, any node $i$ that belongs to class 0 connects (mod 21) to $i+3$, $i-3$, $i+4$ and $i-10$.

### Vertex-Transitivity

In the introduction, we mentioned that all Cayley graphs share the following property:

**Theorem 1** *All Cayley graphs are vertex-transitive.*

The proof of the theorem can be found in [13] and is not repeated here.

Mathematically this implies that for any two nodes $a$ and $b$ in the graph there exists an automorphism of the graph that maps $a$ to $b$. This property is very useful for practical implementation of interconnection networks. Most of the well known interconnection graphs, such as the toroidal mesh, hypercube and cube-connected cycle, exhibit this property [9, 10, 11].

A useful interpretation of the vertex-transitive property can be summarized in the following corollary:

**Corollary 1** *Let* $\mathbf{C} = (\mathbf{V}, \mathbf{G})$ *be a Cayley graph as defined. Assume* $a, b, c \in \mathbf{V}$. *If* $a$ *and* $b$ *are connected through a sequence of generators, then* $c$ *and* $c * a^{-1} * b$ *are connected through the same sequence of generators.*

Proof:
Assume $g_1, g_2, \ldots, g_i$ is a sequence of generators connecting $a$ and $b$. We have

$$\begin{array}{rrcl} & b & = & a * g_1 * g_2 * \ldots * g_i \\ \Rightarrow & g_1 * g_2 * \ldots * g_i & = & a^{-1} * b \\ \Rightarrow & c * g_1 * g_2 * \ldots * g_i & = & c * a^{-1} * b \end{array}$$

□

The above interpretation of the vertex-transitive property simplifies the problem of finding a path between two arbitrary nodes to one of finding a path from a fixed node, say the identity, to another node. In other words, routing between nodes $a$ and $b$ can be determined by finding paths between the identity and $a^{-1} * b$. This property is the basis for our routing algorithm described in the routing section. However it is expressed here in the group theoretic domain of Cayley graphs. As mentioned in [2], elements in such domains are usually not simply ordered. By transformation to a GCR, the nodes of a Cayley graph are necessarily given integer labels and are henced ordered.

It is clear that in the GCR representation, nodes of the same class carry the vertex-transitive property of the original Cayley graphs. For example, if $a$ and $b$ are connected through a sequence of generators, then $a+q$ are connected to $b+q$ through the same sequence of generators. However it is unclear that the general vertex transitive property of Cayley graphs is as readily expressed after the transformation to GCR form. In order to formulate the property in the integer domain, we need more information on the particular group. Recall that in [2], choices of the transform element $T$ ($T^m = I$) and the representing element ($a_i$ for class $i$, $i = 0, 1, \ldots, q-1$) of each class are arbitrary and hence a GCR representation of a Cayley graph is not unique. To formulate the vertex-transitive property we need more information on these choices. In particular, we need three tables or alternately three functions of the indices appearing on the left:

1. a table of the inverse of the representing elements;
$$a_i^{-1} = T^{l_i} * a_{r_i} \qquad i = 0, 1, \ldots, q-1 \qquad (1)$$
2. a multiplication table of the representing elements:
$$a_i * a_j = T^{\hat{l}_{ij}} * a_{\hat{r}_{ij}} \qquad i, j = 0, 1, \ldots, q-1 \quad (2)$$
3. a multiplication table of $a_i$ and $T^h$ ($i = 0, 1, \ldots, q-1$; $h = 0, 1, \ldots, m-1$).
$$a_i * T^h = T^{\tilde{l}_{ih}} * a_{\tilde{r}_{ih}} \qquad (3)$$

Here $l_i, \hat{l}_{ij}$ and $\tilde{l}_{ih}$ range from 0 to $m-1$; whereas $r_i, \hat{r}_{ij}$ and $\tilde{r}_{ih}$ range from 0 to $q-1$. Once we have these tables, Corollary 1 in the integer domain of GCR can be stated as:

**Corollary 2** *For a Cayley graph in the GCR domain, given $i = m_1 q + c_1$, $j = m_2 q + c_2$ and $i' = m'q + c'$. If $i$ is connected to $j$ through a sequence of generators, $i'$ is connected to $j'$ through the same sequence of generators, then $j'$ can be found according to equations (1) to (3) with the following mapping or functional composition:*

$$j' \Leftrightarrow T^{m'} * a_{c'} * a_{c_1}^{-1} * T^{m_2 - m_1} * a_{c_2} \qquad (4)$$

The proof can be obtained by noting that $i$, $j$, and $i'$ correspond to elements $T^{m_1} * a_{c_1}$, $T^{m_2} * a_{c_2}$ and $T^{m'} * a_{c'}$ in the group domain. Using equations 1 to 3, the right hand side of equation 4 can be simplified to $T^t * a_c$, where $0 \le t < m$ and $0 \le c < q$ and $j' = t * q + c$.

For some finite groups, we can make simplifying choices of the transform element $T$ and the representing elements $a_i$ ($i = 0, 1, \ldots, q-1$) such that these tables can be expressed as equations. We use Borel Cayley graphs to illustrate this concept.

## Borel Cayley Graphs

In this section we discuss Cayley graphs obtained from the Borel subgroup. Let $a$ and $k$ be integers as defined in Definition 3. In this group, we can choose the transform element $\mathbf{T} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ where $\mathbf{T}^p = \mathbf{I}$. We then have $q = n/p = k$ classes. We choose the representing element of each class as $a_i = \begin{pmatrix} a^i & 0 \\ 0 & 1 \end{pmatrix}$. These choices make it possible to reduce the tables to simple equations:

$$\begin{array}{rcl} a_i^{-1} & = & a_{q-i} \\ a_i * a_j & = & a_{<i+j>_q} \\ a_i * \mathbf{T}^h & = & \mathbf{T}^{<a^i h>_p} * a_i, \end{array} \qquad (5)$$

where $< \ >_p$ signifies the operation within the bracket $< \ >$ is done in modulo $p$. As usual, contiguous variables indicate multiplication. That is, $< a^i h >_p$ means $a^i$ times $h$ modulo $p$ and $< i + j >_q$ means $i$ plus $j$ modulo $q$. The derivation of equations 5 is as follows:

We observe that with such $\mathbf{T}$ and $a_i$, a matrix $\begin{pmatrix} a^t & y \\ 0 & 1 \end{pmatrix} = \mathbf{T}^y * a_t$. In other words, $\begin{pmatrix} a^t & y \\ 0 & 1 \end{pmatrix}$ is mapped to $yq + t$. With this mapping in mind,

$$a_i = \begin{pmatrix} a^i & 0 \\ 0 & 1 \end{pmatrix} \Rightarrow a_i^{-1} = \begin{pmatrix} a^{q-i} & 0 \\ 0 & 1 \end{pmatrix} = a_{q-i}$$

$$a_i * a_j = \begin{pmatrix} a^{<i+j>_q} & 0 \\ 0 & 1 \end{pmatrix} = a_{<i+j>_q}$$

$$\begin{array}{rcl} a_i * \mathbf{T}^h & = & \begin{pmatrix} a^i & 0 \\ 0 & 1 \end{pmatrix} * \begin{pmatrix} 1 & h \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} a^i & < a^i h >_p \\ 0 & 1 \end{pmatrix} \\ & = & \mathbf{T}^{<a^i h>_p} * a_i \end{array}$$

The vertex transitivity in the integer domain can then be summarized:

Given source $i = m_1 q + c_1$ and destination $j = m_2 q + c_2$.
While $(i \neq j)$
Step 1: Find $j'$ according to equations (1) to (4).
Step 2: Choose an outgoing link according to row $j'$ of routing table.
Step 3: Determine the new source $i'$ according to the GCR constants associated with each link.
Step 4: $i = i'$ and $j = j'$.

Table 1: *A Pseudo-Code for the Routing Algorithm*

**Corollary 3** *Assume a Borel Cayley graph in GCR representation with transform element* $T = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ *and representing element of each class $i$ as* $a_i = \begin{pmatrix} a^i & 0 \\ 0 & 1 \end{pmatrix}$. *Let $i = m_1\ q + c_1$, $j = m_2\ q + c_2$ and $i' = m'q + c'$. If $i$ is connected to $j$ with a sequence of generators then $i'$ is connected to $j'$ with the same sequence of generators, where*

$$j' = < m' + a^{<c'-c_1>_q}(m_2 - m_1) >_p \ q + < c' - c_1 + c_2 >_q . \quad (6)$$

The proof of equation 6 is a simple substitution of equations 5 to equation 4. The corollary above provides an explicit formula to identify $j'$. With this formula, an iterative routing algorithm that involves transforming the original problem of finding a path between nodes $i$ and $j$ to a new problem of routing between node 0 and $j'$ is feasible. We present such a routing algorithm in the next section.

## Routing

Routing is an important practical aspect of communication networks. Our goal has been to develop time and space efficient routing algorithms for large, dense graphs. So far we have produced different routing algorithms for Cayley graphs in general and Borel Cayley graphs in particular [2] [16] [17]. In this section, we present a routing algorithm that exploits the vertex-transitive property of Cayley graphs.

Recall that for an irregular network, which has arbitrary connection rules, routing can be achieved by suitable table lookup [3]. The idea is to produce a *routing table* of size $(n-1) \times \delta$ at each node ($\delta$ is the degree and $n$ is the number of nodes). The table consists of $(n-1)$ rows and there are $\delta$ bits in each row, corresponding to the number of available links at each node. At node $i$, the 1-bit locations on row $j$ of the table indicate the communication links that lead to shortest paths from nodes $i$ to $j$. Consider a degree-4 graph with links **A**, **B**, $\mathbf{A}^{-1}$ and $\mathbf{B}^{-1}$. Suppose there are two shortest paths from node $i$ to node $j$, one path corresponds to taking link **A** and the other path corresponds to taking link **B** from node i. At node $i$, row $j$ of the routing table would then have the entries corresponding to link **A** and **B** marked as 1 and the other two entries marked 0.

This table lookup scheme finds multiple, shortest path(s) between any two nodes and works for any network. Routing is achieved through a routing table, which identifies appropriate outgoing links for any incoming message according to its destination. The space complexity of the algorithm is of $O(n^2)$ for the entire network because each node needs $(n-1) \times \delta$ bits and there are $n$ nodes in the network. The time complexity for a table lookup scheme is constant or $O(1)$ at a single node and $O(D)$ for a complete path. One drawback of this algorithm is that it is strictly progressive which means it is generally not possible to determine the entire, multi-step path at a single node.

In Corollary 1, we provided an interpretation of the vertex-transitive property. Such an interpretation allows us to transform the original problem of routing between two nodes $a$ and $b$ to a new problem of routing between the identity and $a^{-1} * b$. This provides the basis for improving the table lookup scheme described above.

However the facts that both the original graph and Corollary 1 are expressed in the group domain hinder the development of practical routing algorithm. As mentioned in [2], elements in group domain are often not ordered. In [2], we have successfully transformed Cayley graphs to a GCR format, which provides integer labels for the group elements as well as a concise description of the connection rules for the graph. To properly exploit the vertex-transitive property of Cayley graph in GCR, we need to express such a property in the integer domain. In the last section we provided a framework (specifically equations (1) to (4) for general Cayley graph and equation (6) for Borel Cayley graph) for expressing the vertex-transitive property in GCR representations.

With this framework, we can identify a destination node $j'$ with origin node 0 given any source and destination pair, $i$ and $j$. This suggests a table-based routing algorithm that will permit the determination of a multi-step path at a single node. At node 0, the database consists of a routing table as described earlier. An *identical* routing table is then used at every other node. When a message

For a degree-4 Borel Cayley graph in GCR representations
with $\mathbf{T} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ and $a_i = \begin{pmatrix} a^i & 0 \\ 0 & 1 \end{pmatrix}$,
we have $q = k$ classes, where $a^k = 1 \pmod p$.
Assume the generators to be $\mathbf{A}$, $\mathbf{B}$, $\mathbf{A}^{-1}$ and $\mathbf{B}^{-1}$, where
$\mathbf{A} = \begin{pmatrix} a^{t_1} & y_1 \\ 0 & 1 \end{pmatrix}$ and $\mathbf{B} = \begin{pmatrix} a^{t_2} & y_2 \\ 0 & 1 \end{pmatrix}$.
Given source $i = m_1 q + c_1$ and destination $j = m_2 q + c_2$.

While $(i \neq j)$
Step 1: Identify new destination,
$j' = < a^{q-c_1}(m_2 - m_1) >_p q + < c_2 - c_1 >_q$
Step 2: From row $j'$ of database, determine which link to take.
Step 3: Identify new source, $i' = mq + c$ and
$m = y_1, \quad c = t_1,$ if link $A$ was chosen
$m = y_2, \quad c = t_2,$ if link $B$ was chosen
$m = p - < a^{q-t_1} y_1 >_p, \quad c = q - t_1,$ if $A^{-1}$ was chosen
$m = p - < a^{q-t_2} y_2 >_p, \quad c = q - t_2,$ if $B^{-1}$ was chosen
Step 4: $i = i'$ and $j = j'$

Table 2: *Iterative Routing for Borel Cayley Graphs*

is generated at source $i$ with destination $j$, the framework (equation (1) to (4) for general Cayley graph and equation (6) for Borel Cayley graph) described in the last section is used to identify $j'$. Then using the routing table at row $j'$, we can determine the link that corresponds to a shortest path. Once a link is identified, we can find the neighboring node by using the GCR constants. We then have a new problem of routing between this neighboring node and $j'$. This procedure is repeated until the source and destination are the same. Table 1 consists of a pseudo-code for the algorithm. We observe that this improved routing algorithm retains the ability of finding multiple, shortest path(s) and in addition is capable of determining the entire path from the source without increasing the space complexity, which is asymptotically $O(n^2)$ for the entire network of size $n$. The multi-step algorithm is iterative, and the time complexity is still $O(D)$, where $D$ is the diameter. The following example illustrates such a routing algorithm for Borel Cayley graph.

### An Example

As indicated in Corollary 3, we can choose a specific transform element, $\mathbf{T}$, and a special set of representing elements for Borel Cayley graphs such that there exists a concise formula for $j'$ (Equation 6). The existence of such a formula simplifies the use of the iterative routing algorithm. In short, the algorithm can be summarized in Table 2.

As an example, we consider the Borel Cayley graph with $p = 7$ and $a = 2$, given in section 2.3.1. Note that the choices of the transform element $\mathbf{T}$ and the class representing elements $a_i$ in this example satisfy the conditions described in Corollary 3, which means Equation 6 is true.

At each node, we store the size $20 \times 4$ routing table shown in Figure 2. The numbers 3, 4, 18, and 11 in Figure 1 correspond to the values of $i'$ for the different links in Step 3 of Table 2. Suppose we need to route a message from node 0 to 16. There are three shortest paths between the two nodes, namely:

$$\text{path 1}: \quad 0 \xrightarrow{\mathbf{B}} 4 \xrightarrow{\mathbf{A}} 10 \xrightarrow{\mathbf{A}} 16$$
$$\text{path 2}: \quad 0 \xrightarrow{\mathbf{A}^{-1}} 18 \xrightarrow{\mathbf{B}} 1 \xrightarrow{\mathbf{A}^{-1}} 16$$
$$\text{path 3}: \quad 0 \xrightarrow{\mathbf{B}^{-1}} 11 \xrightarrow{\mathbf{A}} 2 \xrightarrow{\mathbf{B}^{-1}} 16$$

We begin with iteration 0, $i = 0$ and $j = 16$. According to Step 1 of Table 2, we identify $j'$ as 16. From row 16 of the routing table, there are three choices corresponding to the three shortest paths. Arbitrarily we choose link $\mathbf{B}$. According to step 3, the new source $i' = 4$. Now we enter iteration 1 with $i = 4$ and $j = 16$. Step 1 identifies $j'$ to be 6. From row 6 of the routing table, we pick link $\mathbf{A}$, which determines the new source $i' = 3$. Then at iteration 2, $i$ is 3 and $j$ is 6. For this source and destination pair we have $j' = 3$, which means link $\mathbf{A}$ should be taken according to row 3 of the routing table. The equations in step 3 then determine $i'$ to be 3. Finally at iteration 3, both source and destination are 3 and the algorithm terminates. We have thus successfully found path 1 ($\mathbf{BAA}$) between 0 and 16. The iterations of this example are summarized in Table 3.

### Conclusions

Because of their vertex-transitive property and density, Cayley graphs and Borel Cayley graphs in particular are interesting interconnection models for multicomputers. However, Borel Cayley graphs are defined in the matrix domain, which does not have simple ordering and it is

| | A | B | $A^{-1}$ | $B^{-1}$ |
|---|---|---|---|---|
| 1 | | | 1 | |
| 2 | | | | 1 |
| 3 | 1 | | | |
| 4 | | 1 | | |
| 5 | 1 | 1 | 1 | |
| 6 | 1 | | | |
| 7 | 1 | | | |
| 8 | | | 1 | |
| 9 | 1 | | | 1 |
| 10 | | 1 | | |
| 11 | | | | 1 |
| 12 | | | 1 | 1 |
| 13 | 1 | 1 | | 1 |
| 14 | 1 | | | |
| 15 | | | 1 | |
| 16 | | 1 | 1 | 1 |
| 17 | 1 | 1 | 1 | |
| 18 | | | 1 | |
| 19 | | 1 | | |
| 20 | | | | 1 |

Fig. 2. *A Routing Table for* $\mathbf{BL}_2(\mathbf{Z}_7)$

difficult to design routing algorithm in this domain.

GCR provided an integer-domain representations of Cayley graphs. Such a representation is isomorphic to the original Cayley graph, and thus retains all the properties, including density and symmetry. GCR graphs are obviously partially symmetric. Nodes within the same class have the same connection rules. More specifically, if $i$ connects to $j$ through a sequence of generators, then $i + m'\ q \pmod n$ connects to $j + m'q \pmod n$ through the same sequence of generators, where $m'$ is any integer.

Since Cayley graphs are symmetric, it should be possible to establish a more general relationship: if $i$ connects to $j$ through a sequence of generators, then $i'$ connects to $j'$ through the same sequence of generators, where $i'$ is any integer vertex-labels and $j'$ is a function of $i$, $j$, and $i'$. In this paper, we provided a framework for expressing this relationship. Basically, we need to identify three tables or functions that specify the arbitrary choices of the transform element $\mathbf{T}$ and class representing elements in the transformation from Cayley graphs to GCR. In the case of Borel Cayley graphs, these tables can be simplified into equations, and $j'$ is a non-linear function of $i$, $j$ and $i'$ (Equation 6).

With the establishment of such a framework, an iterative routing algorithm that exploits the vertex-transitive property is developed. Such a routing algorithm is an improved version of the routing table scheme introduced in [3], which is capable of finding multiple, shortest path(s) between any source and destination pair. The advantage of this new algorithm is that the ability of finding multiple, shortest path(s) is retained while the entire path can be determined at the source without increasing the space complexity of the algorithm. An example from a Borel Cayley graph is used to illustrate the formulation

| iteration | $i$ | $j$ | link |
|---|---|---|---|
| 0 | 0 | 16 | $B$ |
| 1 | 4 | 16 | $A$ |
| 2 | 3 | 6 | $A$ |
| 3 | 3 | 3 | -- |

Table3: *A Summary of Iterations*

of vertex-transitivity in GCR and the exploitation of the property in routing. The asymptotic space complexity for the entire network of size $n$ of the improved algorithm is $O(n^2)$ and the time complexity is of $O(D)$, where $n$ is the number of nodes and $D$ is the diameter of the graph.

## References


[1] D.V. Chudnovsky, G.V. Chudnovsky, and M.M. Denneau. Regular Graphs with Small Diameter as Models for Interconnection Networks. Technical Report RC 13484(60281), IBM Research Division, February 1988.

[2] B.W. Arden and K.W. Tang. Representation and Routing of Cayley Graphs. *IEEE Transactions on Communications*, 39(11), November 1991.

[3] D.A. Reed and R.M. Fujimoto. *Multicomputer Networks.* MIT Press, Cambridge, Massachusetts, 1987.

[4] Larry D. Wittie. "Communication Structures for Large Networks of Microcomputers". *IEEE Transactions on Computers*, 30(4):264–273, April 1981.

[5] J.A. Bondy and U.S.R. Murty. *Graph Theory With Applications.* North-Holland, New York, 1979.

[6] J.C. Bermond, C. Delorme, and J.J. Quisquater. "Tables of Large Graphs with Given Degree and Diameter". *Information of Processing Letters*, 15(1):10–13, 1982.

[7] D. A. Reed and M.L. Patrick. "Iterative Solution of Large, Sparse Linear Systemson a Static Data Flow Architecture: Performance Studies". *IEEE Transactions on Computers*, 34(10):874–880, October 1985.

[8] S.B. Akers and B. Krishnamurthy. "A Group-Theoretic Model for Symmetric Interconnection Networks". *IEEE Transactions on Computers*, 38(4):555–565, April 1989.

[9] Mondher Ben-Ayed. *Dynamic Routing for Regular Direct Computer Networks.* PhD thesis, Electrical Engineering Department, College of Engineering and Applied Science, University of Rochester, Rochester, New York, 1990.

[10] J.P. Hayes. "Architecture of a Hypercube Supercomputer". In *Proceedings, 1986 International Conference on Parallel Processing*, pages 653–660, 1986.

[11] F.P. Preparata and J. Vuillemin. "The Cube-Connected Cycles: A Versatile Network for Parallel Computation". *Communications of the ACM*, pages 300–309, May 1981.

[12] J.C. Bermond and C. Delorme. "Strategies for Interconnection Networks: Some Methods from Graph Theory". *Journal of Parallel and Distributed Computing*, 3:433–449, 1986.

[13] N. Biggs. *Algebric Graph Theory*. Cambridge University Press, London, 1974.

[14] Mark Homewood, David May, David Shepherd, and Roger Shepherd. "The IMS T800 Transputer". *IEEE MICRO*, pages 10–26, October 1987.

[15] M. Hall. *The Theory of Groups*. Wiley, New York, 1974.

[16] K.W. Tang and B.W. Arden. Representations and Routing of Borel Cayley Graphs. Technical Report EL-89-07, Department of Electrical Engineering, University of Rochester, 1989. (To be published in the proceedings of INFO JAPAN, 1990.).

[17] B.W. Arden and K.W. Tang. "Routing for Generalized Chordal Rings". In *Proceedings of the ACM:18th Computer Science Conference*, Washington, D.C., February 20-22 1990.