Conquering Edge Faults in a Butterfly with Automorphisms

Meghanad D. Wagh and Khadidja Bendjilali
Department of Electrical and Computer Engineering
Lehigh University, Bethlehem, PA 18015, USA
mdw0@Lehigh.Edu, khb5@Lehigh.Edu

Abstract

Mapping an algorithm to an architecture with faults is an important problem in parallel processing. This paper deals with wrapped butterfly architectures with edge faults. We investigate the effect of automorphisms of a wrapped butterfly on its edges. Given a fault set, one can then choose an appropriate automorphism to map the algorithm to use only fault-free edges. By using an algebraic model of the butterfly, we obtain simple expressions for all its automorphisms. Use of powerful algebraic techniques then quickly yield the edge transformations due of these automorphisms. This strategy of avoiding edge faults using automorphisms is quite novel because previously automorphisms have been employed only to avoid the node faults. We illustrate our methods by mapping Hamilton cycle on the butterfly under various edge fault scenarios.

1. Introduction

Over the last few decades, the semiconductor technology has delivered increasingly faster and and yet smaller integrated circuits. Unfortunately, this ability to create chips of shrinking sizes and higher complexities has now hit the technological barriers. It is therefore an accepted premise that parallel processing will be the future of computing.

In a distributed memory parallel machine, a large number of processors work on the same problem concurrently. These processors exchange information using interconnection networks. Unfortunately, the communication speeds have not kept up with the computational speeds. As a result, the performance of a parallel machine is often dictated by the underlying interconnection network. Because of this, the parallel architecture can be modeled as a graph in which the nodes represent the processors and edges, the communication paths between them. Hypercubes, butterflies and meshes are some of the popular graphs on which many of the existing parallel machines are based [1]. To run a computation on a parallel machine, one partitions the task into a set of sub-tasks and develops a task graph. One then maps the task graph on the interconnection graph of the architecture such that the communicating sub-tasks are mapped (as far as possible) on processors that have a direct link between them. However, since most parallel machines have a large number of processors and interconnection links, the probability of some processors and/or links developing faults cannot be ignored. Therefore there has been a substantial research effort to develop strategies to map algorithms (or rather, their task graphs, on the interconnection graphs that have faulty nodes or edges.

This paper deals with the *wrap-around butterfly network* denoted here by B_n, where n denotes the dimension of the butterfly. B_n represents a good trade-off between the cost and the performance of a parallel machine. It has a large number of processors ($n2^n$), fixed node degree (4), low diameter ($\lfloor 3n/2 \rfloor$), symmetry, and ability to support a variety of parallel algorithms [1–5]. Cube Connected Cycles is a sub-graph of B_n[6]. Other extensions of B_n are also available [7, 8].

Let Z_n denote the group of integers $\{0, 1, \ldots, n-1\}$ under the operation of addition modulo n and Z_2^n, the group of binary vectors of length n under the operation of modulo 2 addition. Then the wrapped butterfly graph B_n, $n \geq 3$, is defined to have $n2^n$ nodes, each labeled with a pair (m, V) where $m \in Z_n$ and $V \in Z_2^n$. A node (m, V) is connected to four distinct nodes: $(m+1, V)$, $(m+1, V \oplus 2^m)$, $(m-1, V)$ and $(m-1, V \oplus 2^{m-1})$. Note that the third and the fourth edges are inverses of the first and the second edges respectively. Thus the edges of a wrapped butterfly are bidirectional. The first index m of the node (m, V) is often called its column and the second index, V, its row.

With the advances in the VLSI technology, it is now possible to build parallel machines with a large number of processors. However, larger the machine, higher is the probability that one or more of its processors or links will develop a fault. Thus, for the underlying networks of these large machines, mappings of algorithms on faulty graphs becomes an important design issue.

Previous results about mappings on faulty butterflies include one by Vadapalli and Srimani who have shown that in B_n, there exists a cycle of length at least $n2^n - 2$ with one faulty node and $n2^n - 4$ with two faulty nodes [9]. Later, Tsai et al., improved this to show that for odd n, cycle length $n2^n - 2$ is possible with two faulty nodes [10]. They also proved that in the presence of one faulty node and one faulty edge, there exists a cycle of length $n2^n - 2$ when n is even, and $n2^n - 1$, when n is odd. Hwang and Chen have shown that the maximal cycle of length $n2^n$ can be embedded in a faulty butterfly even with two edge faults [11]. However, these studies have used the binary representation of the butterfly resulting in rather complex mappings.

This paper proposes a new approach to mappings on faulty butterflies using an algebraic model first given in [5]. We show that with this model, it is rather simple to obtain all the automorphisms of the butterfly. Automorphisms can be used to translate an algorithm mapping to one that avoids node faults. For example, an algorithm mapping can avoid a faulty node N_{faulty} by using a free node N_{free} (assuming one exists) and an automorphism $\phi(\cdot)$ of the interconnection graph such that $\phi(N_{free}) = N_{faulty}$. By remapping tasks on each node N to node $\phi(N)$, one can run the algorithm entirely on fault free nodes. Automorphisms have also been used to obtain better VLSI layouts of butterfly networks [12, 13].

This paper obtains all the automorphisms of the butterfly (Theorems 1, 2, 4). We explore the edge transformations in butterfly networks due to automorphisms. In particular, we show that automorphisms can change the type (to be defined later) of edges. Exactly 2^n automorphisms of B_n affect all the edges in a column similarly (Theorem 5). The remaining 2^n automorphisms change the type of exactly half the edges in a column while the other edges retain their type (Theorems 7, 8). Further, one can design automorphisms to achieve the desired edge transformation. This allows one to map algorithms onto butterfly machines with edge faults. As examples, we show that a butterfly B_n supports a Hamilton cycle even when it has faulty edges in all but two of its rows as long as the faults in a given set of rows are constrained to one type and those outside to one type as well (Theorems 10, 11). Further, the requirement of two fault-free rows can be lifted when n is odd (Theorems 12, 13). We also show that B_n with up to 2^n faulty edges of the same type in each column except one is still Hamiltonian (Theorem 9). Our procedure allows one to map the Hamilton cycle on to the faulty butterfly easily and directly. The simplicity of the automorphism and the resultant edge mappings show promise of wide applicability of this technique to a variety of applications.

2 An algebraic model of the butterfly

Binary representation has been widely used to model many common interconnection networks including the butterfly. However, binary models are difficult to analyze and complex to use. In this paper we will use an algebraic model using direct product of finite fields and cyclic groups, first given in [5]. The simplicity of the model and access to powerful algebraic techniques allows us to explore the automorphisms of the butterfly with relative ease.

In the butterfly model of [5], nodes of B_n are labeled with pairs (m, X), $m \in C_n$, $X \in GF(2^n)$, where C_n is the cyclic group of integers 0 through $n-1$ under the operation of addition modulo n and $GF(2^n)$ is the finite field of 2^n elements. We will often refer to m as the column and X, the row, of node (m, X). Let α denote the primitive element of $GF(2^n)$ and $\langle \beta_{n-1}, \beta_{n-2}, \ldots, \beta_0 \rangle$, its dual basis. The node connectivity of graph B_n can then be described through an algebraic relationship. In particular, a vertex (m, X) of B_n is connected to the vertices $(m+1, \alpha X)$, $(m+1, \alpha X + \beta_{n-1})$, $(m-1, \alpha^{-1} X)$ and $(m-1, \alpha^{-1} X + \beta_0)$. For convenience, We refer to these four edges as f, g, f^{-1} and g^{-1} respectively. It is easy to verify that if edge f goes from node N_1 to N_2, then the edge that goes from N_2 to N_1 is f^{-1}. The same observation is also true for g and g^{-1}. The simplicity of this model should be apparent from the fact that the two components of the destination of (m, X) are independent. On the other hand, in binary representation, the destination of (m, V) is $(m+1, V \oplus 2^m)$, where, as one can see, the second coordinate is a function of both m and V, the two coordinates of the source. For the proof and examples of the algebraic model, reader is referred to [5]. Tables 1 and 2 show the relationships between the elements of $GF(2^3)$ and $GF(2^4)$ used in the definition of B_3 and B_4. .

Table 1: Structure of $GF(2^3)$.

Primitive Polynomial: $x^3 + x + 1$ Elements and their Relationships:	
0	$\alpha^3 = \alpha + 1$
1	$\alpha^4 = \alpha^2 + \alpha$
α	$\alpha^5 = \alpha^2 + \alpha + 1$
α^2	$\alpha^6 = \alpha^2 + 1$
Dual Base $\langle \beta_2, \beta_1, \beta_0 \rangle$ = $\langle \alpha, \alpha^2, 1 \rangle$.	

For the purpose of this paper, one need not worry about the dual basis elements, except that they are constants satisfying the properties given in the following Lemma.

Lemma 1 *Let $\langle \beta_{n-1}, \beta_{n-2}, \ldots, \beta_0 \rangle$ denote the dual base*

Table 2: Structure of $GF(2^4)$.

Primitive Polynomial: x^4+x+1 Elements and their Relationships:	
0	$\alpha^7=\alpha^3+\alpha+1$
1	$\alpha^8=\alpha^2+1$
α	$\alpha^9=\alpha^3+\alpha$
α^2	$\alpha^{10}=\alpha^2+\alpha+1$
α^3	$\alpha^{11}=\alpha^3+\alpha^2+\alpha$
$\alpha^4=\alpha+1$	$\alpha^{12}=\alpha^3+\alpha^2+\alpha+1$
$\alpha^5=\alpha^2+\alpha$	$\alpha^{13}=\alpha^3+\alpha^2+1$
$\alpha^6=\alpha^3+\alpha^2$	$\alpha^{14}=\alpha^3+1$
Dual Base $\langle\beta_3,\beta_2,\beta_1,\beta_0\rangle$ = $\langle 1,\alpha,\alpha^2,\alpha^{14}\rangle$.	

of GF(2^n). Then

$$\beta_i = \begin{cases} \alpha\beta_0 & \textit{if} \quad i = n-1 \\ \alpha\beta_{i+1} + p_{i+1}\beta_{n-1} & \textit{if} \quad i = 0, 1, \ldots, n-2, \end{cases}$$

where α is the primitive element of the field and p_i is the coefficient of x^i in the primitive polynomial used to generate the field.

Proof. Omitted for brevity. ∎

3 Automorphisms of the butterfly network

Wagh and Guzide have previously shown that the algebraic model allows efficient mappings of cycles of all (possible) lengths and trees of largest sizes on the butterfly [5]. We extend their work by exploring the automorphisms of butterfly in the same setting.

We show that there are two kinds of automorphisms of B_n. There are $n2^n$ automorphisms which map nodes in column m to nodes in column $m+t$ for some t. We denote these automorphisms by $\phi(\cdot)$. There is also another independent automorphism which maps nodes in column m to nodes in column $-m \bmod \mathrm{n}$. We denote this automorphism by $\psi(\cdot)$. A product of $\psi(\cdot)$ with the set of $\phi(\cdot)$ automorphisms provides all the $n2^{n+1}$ automorphisms of B_n. In a previous paper we have discussed the set of automorphismas $\phi(\cdot)$ [14]. This paper is focused mostly on automorphism $\psi(\cdot)$, its effect on edges of B_n and its application (along with $\phi(\cdot)$) to mappings on faulty butterflies.

We begin by defining the automorphism $\phi(\cdot)$ in Theorem 1 whose proof is given in [14].

Theorem 1 *Let constants $K_0, K_1, \ldots, K_{n-1} \in GF(2^n)$ satisfy*

$$K_i = \alpha K_{i-1} \;\; \textit{or} \;\; \alpha K_{i-1} + \beta_{n-1},$$

where the indices of K are computed modulo n. Then the function $\phi(\cdot) : B_n \to B_n$ defined as

$$\phi((m,\ X)) = (m+t,\ X+K_m) \tag{1}$$

for any $t \in Z_n$, is an automorphism of B_n, i.e., it maps nodes of B_n to nodes and edges to edges.

Note that constant t merely translates edges in one column to a column t away. This t and constant elements $K_i \in GF(2^n)$, $0 \le i < n$ fully define the automorphism $\phi(\cdot)$. We will henceforth refer to t as the *column offset* and K_is as the *automorphism offsets*,

One can see the simplicity of the automorphism $\phi(\cdot)$ defined in (1). Every node in the network is applied the same column offset and every node in the same column is applied the same automorphism offset. Further, the offsets of the two coordinates of a node label are *independent*. This makes use of such an automorphism especially attractive.

Theorem 1 allows one to design such an automorphism under various conditions. For example, suppose one wants an automorphism such that for a given pair of nodes $N_1 = (a,\ U), N_2 = (b,\ V) \in B_n$, the automorphism maps N_1 to N_2, i.e.,

$$\phi(N_1) = N_2. \tag{2}$$

(If we can do this for an arbitrary pair of nodes, it would imply that B_n is a symmetric network.) Such a mapping can be obtained by choosing a column offset t and automorphism offsets $K_0, K_1, \ldots, K_{n-1} \in GF(2^n)$ satisfying condition in Theorem 1 and then defining ϕ as in (1). Note that the relations between K_is provide certain flexibility in the choice of the constants. We exploit this flexibility to ensure that (2) is satisfied.

Let us rewrite the relations between K_is as

$$K_i = \alpha K_{(i-1) \bmod \mathrm{n}} + c_i\beta_{n-1}, \qquad 0 \le i \le n-1, \tag{3}$$

where each c_i is either 0 or 1. One can use (3) repeatedly to get

$$K_a = (1+\alpha^n)^{-1}\Big(\sum_{j=0}^{n-1} c_{(a-j) \bmod \mathrm{n}}\ \alpha^j\Big)\beta_{n-1}. \tag{4}$$

Further, if $\phi((m,\ X)) = (m+t,\ X+K_m)$, then to satisfy (2) requires that

$$\begin{aligned} t &= (b-a) \bmod \mathrm{n} \qquad \text{and} \\ K_a &= U+V. \end{aligned} \tag{5}$$

By combining (4) and (5), one gets

$$(U+V)(\alpha^n+1)\beta_{n-1}^{-1} = \sum_{j=0}^{n-1} c_{(a-j) \bmod \mathrm{n}}\alpha^j, \tag{6}$$

One can see that the left hand side of (6) is an element of $GF(2^n)$ and can therefore be uniquely expressed in the normal basis $\langle \alpha^{n-1}, \alpha^{n-2}, \ldots, 1 \rangle$. This gives the unique set of values for c_is. One can then use these values in (3) to obtain the automorphism offsets $K_{(a+1)\,\mathrm{mod}\,\mathrm{n}}$, $K_{(a+2)\,\mathrm{mod}\,\mathrm{n}}$, ..., $K_{(a-1)\,\mathrm{mod}\,\mathrm{n}}$.

As is evident from this discussion, all the automorphism offsets for any $\phi(\cdot)$ are related such that choosing any one of them, say, K_0, fixes all the others. On the other hand, distinct K_0 and t values give rise to distinct automorphisms. Thus there are exactly $n2^n$ automorphisms of butterfly B_n when the first index of all the nodes is translated by the same amount.

We now specify automorphism $\psi(\cdot)$ of B_n that reflects the column index of each node.

Theorem 2 *For every $X \in GF(2^n)$, $X = \sum_{i=0}^{n-1} x_i\beta_i$, let $X' = \sum_{i=0}^{n-1} x_i\beta_{n-1-i}$. Then the mapping*

$$\psi(m, X) = (n - m, X')$$

is an automorphism of B_n.

Proof. It is simple to see that $\psi(\cdot)$ is one-to-one and onto. We only need to prove that it preserves the edge connectivity of B_n. In particular, we demonstrate that since vertex (m, X) is connected to the vertices $(m + 1, \alpha X + c\beta_{n-1})$, $c \in \{0, 1\}$, $\psi(m, X)$ is also connected to vertices $\psi(m + 1, \alpha X + c\beta_{n-1})$. Let $X = \sum_{i=0}^{n-1} x_i\beta_i$. Then using the relationships between the consecutive β_is given in Lemma 1, one gets

$$\begin{aligned} \alpha X + c\beta_{n-1} &= \sum_{i=1}^{n-1}(x_i\beta_{i-1} + x_i p_i\beta_{n-1}) + (c + x_0)\beta_0 \\ &= \sum_{i=0}^{n-2} x_{i+1}\beta_i + (c + \sum_{i=0}^{n-1} p_i x_i)\beta_{n-1}. \quad (7) \end{aligned}$$

Thus

$$\psi(m + 1, \alpha X + c\beta_{n-1}) = (n - m - 1, Y) \qquad (8)$$

where,

$$\begin{aligned} Y &= \sum_{i=0}^{n-2} x_{i+1}\beta_{n-1-i} + (c + \sum_{i=0}^{n-1} p_i x_i)\beta_0 \\ &= \sum_{i=1}^{n-1} x_i\beta_{n-i} + (c + \sum_{i=0}^{n-1} p_i x_i)\beta_0) \qquad (9) \end{aligned}$$

Now,

$$\begin{aligned} \alpha Y &= \sum_{i=1}^{n-1} x_i\alpha\beta_{n-i} + (c + \sum_{i=0}^{n-1} p_i x_i)\beta_{n-1}) \\ &= \sum_{i=1}^{n-1}(x_i\beta_{n-1-i} + x_i p_{n-i}\beta_{n-1}) \\ &\quad + (c + \sum_{i=0}^{n-1} p_i x_i)\beta_{n-1}) \\ &= \sum_{i=0}^{n-1} x_i\beta_{n-1-i} + c'\beta_{n-1}, \qquad (10) \end{aligned}$$

where $c' \in \{0, 1\}$ denotes

$$c' = c + \sum_{i=0}^{n-1}(p_i + p_{n-i})x_i. \qquad (11)$$

Note that

$$\begin{aligned} \psi(m, X) &= (n - m, \sum_{i=0}^{n-1} x_i\beta_{n-1-i}) \\ &= (n - m, \alpha Y + c'\beta_{n-1}). \qquad (12) \end{aligned}$$

From (8) and (12) it is obvious that vertex $\psi(m, X)$ is connected to vertex $\psi(m + 1, \alpha X + c\beta_{n-1})$, $c \in \{0, 1\}$. ∎

When the context is clear, we sometimes write $\psi(X)$ in place of $\psi((m, X))$. Theorem 3 lists Some basic properties of $\psi(\cdot)$.

Theorem 3 *1. $\psi(\cdot)$ is an order 2 automorphism.*

2. $\psi(X_1 + X_2) = \psi(X_1) + \psi(X_2)$.

3. $\psi((m, X)) = (n - m, X)$ for exactly $2^{\lceil n/2 \rceil}$ values of $X \in GF(2^n)$.

Proof. The first two properties of $\psi(\cdot)$ are obvious from its definition. For any $X = \sum_{i=0}^{n-1} x_i\beta_i$, $\psi((m, X)) = (n - m, X)$ if and only if $x_i = x_{n-1-i}$, $0 \le i < \lfloor n/2 \rfloor$. From this the third property follows. ∎

We end this section with the following theorem enumerating all the automorphisms of B_n.

Theorem 4 *B_n has a total of $n2^{n+1}$ automorphisms.*

Proof. Note that the product of two automorphisms is also an automorphism. Thus in addition to the $n2^n$ automorphisms defined by Theorem 1, another set of automorphisms can be defined by multiplying each of these $\phi(\cdot)$s by the automorphism $\psi(\cdot)$ in Theorem 2. Since the order of automorphism $\psi(\cdot)$ is 2, these are all the automorphisms of B_n. ∎

4 Edge Transformations by automorphisms

This section investigates the effect of an automorphism on the butterfly edges. We call edges $(i-1,X) \rightarrow (i,\alpha X)$ and $(i-1,X) \rightarrow (i,\alpha X+\beta_{n-1})$ for all $X \in GF(2^n)$ as the edges in the ith column of B_n.

The automorphism $\phi(\cdot)$ of Theorem 1 affects all the edges in the same column similarly.

Theorem 5 *Let the automorphism offsets be related as:*

$$K_i = \alpha K_{(i-1)\,\mathrm{mod}\,\mathrm{n}} + c_i\beta_{n-1}, \qquad 0 \leq i \leq n-1,$$

(a) If $c_i = 1$, then the automorphism $\phi(\cdot)$ maps all f edges of B_n in column i to g edges and all g edges to f edges.
(b) If $c_i = 0$, then the automorphism $\phi(\cdot)$ maps all f edges of B_n in column i to f edges and all g edges to g edges.

Proof. Consider an f edge between nodes $N_1 = (i-1,X)$ and $N_2 = (i,\alpha X)$ of the sub-graph of B_n. Now, $\phi(N_1) = (i-1, X+K_{i-1})$ and,

$$\begin{aligned}\phi(N_2) &= (i,\alpha X + K_i)\\ &= (i,\alpha X+\alpha K_{i-1}+c_i\beta_{n-1})\\ &= (i,\alpha(X+K_{i-1})+\beta_{n-1})\end{aligned}$$

From this, one can clearly see that the edge between $\phi(N_1)$ and $\phi(N_2)$ is a g edge. The translation of a g edge into an f edge can be similarly proved. ∎

Note that the automorphism $\phi((m,\ X)) = (m+t,\ X+K_m)$ also advances the column number m by quantity t. In this case, $c_m = 1$ has the effect of mapping the f edges of the sub-graph between columns $m-1$ and m to g edges and all g edges to f edges; but these transformed edges now appear in column $m+t$. Similarly the edges in mth column are mapped to edges of the same type in column $m+t$ if $c_m = 0$.

To describe the effect of the automorphism $\psi(\cdot)$ on the edges of B_n, we first define a set S as

$$S = \{X \in GF(2^n) | \psi(X) = \alpha\psi(\alpha X)\} \qquad (13)$$

Some of the basic properties of S are listed in the following theorem.

Theorem 6 *Let p_i denote the coefficient of x^i in the primitive polynomial used to generate $GF(2^n)$. Then*

1. *$X = \sum_{i=0}^{n-1} x_i\beta_i \in S$ if and only if $\sum_{i=1}^{n-1}\ x_i\ (p_i + p_{n-i}) = 0$.*
2. *For any $X \notin S$, $\alpha\psi(\alpha X)+\psi(X) = \beta_{n-1}$.*
3. *If $p_i = p_{n-i}$, then $\beta_i \in S$,*
4. *S is a subgroup of $GF(2^n)$ under the operation of addition.*
5. *There are exactly 2^{n-1} elements in S.*

Proof. Omitted for brevity. ∎

Set S plays an important role in edge transformations of B_n under $\psi(\cdot)$ as the following theorem shows.

Theorem 7 *When $X \in S$, ψ maps f edges from (m,X) to f edges and g edges to g edges. On the other hand, when $X \notin S$, ψ maps f edges from (m,X) to g edges and g edges to f edges.*

Proof. Consider an edge $(m,X) \rightarrow (m+1,\alpha X + c\beta_{n-1})$. If $c=0$, this represents an f edge and if $c=1$, a g edge. The automorphism maps the first node to $N_1 = (n-m,\psi(X))$ and the second to $N_2 = (n-m-1,\alpha^{-1}\psi(X)+c\beta_0)$ if $X \in S$. Clearly there is an f edge from N_2 to N_1 when $c=0$ and a g edge when $c=1$.

If $X \notin S$, then from the second part of Theorem 6, one can see that the second node maps to $N_2^{'} = (n-m-1,\alpha^{-1}\psi(X)+\alpha^{-1}\beta_{n-1}+c\beta_0) = (n-m-1,\alpha^{-1}\psi(X)+(c+1)\beta_0)$. Thus there is a g edge from $N_2^{'}$ to N_1 when $c=0$ and an f edge when $c=1$. ∎

As a consequence of Theorem 6, we have the following result.

Theorem 8 *Automorphism $\psi(\cdot)$ maps edges from exactly half the rows of the butterfly to the edges of the same type.*

Proof. Theorem 7 shows that edges starting from nodes in the same row (i.e., nodes (m,X) having the same X) behave similarly; all of them either map to edges of the same type (when $X \in S$) or map to edges of the other type (when $X \notin S$). The stated result is true because $|S| = 2^{n-1}$ (Theorem 6, Part 5). ∎

5 Application of automorphisms to tolerate edge faults

Previously automorphisms have only been used to tolerate node faults. However, Theorems 5 and 7 directly express the effect of an automorphism on the butterfly edges. Consequently, one can now use these automorphisms to tolerate edge faults for many mappings on the butterfly.

The general procedure to obtain a fault free mapping on a faulty butterfly is simple. If some edges used in the mapping are faulty but the edges to which they *can* be mapped by *some* automorphism are free, then applying that automorphism to the mapping will allow it to use only fault-free edges. Note that much of the power of this method is due to the fact that we have $n2^{n+1}$ well-defined and simple automorphisms that map edges in a deterministic fashion. We illustrate this procedure by constructing a Hamilton cycle under various edge fault scenarios.

Theorem 9 *If the edges in one of the columns of B_n are fault free and the faults in each of the other columns are limited to only one type of edges, then B_n is Hamiltonian.*

Proof. As shown in [5], it is possible to construct a Hamiltonian cycle in B_n by first constructing two cycles using only f edges; one linking all nodes (m, X), $X \neq 0$, and another linking all nodes $(m, 0)$. These cycles are merged into a Hamiltonian cycle by using a pair of g edges in column t: $(t-1, 0) \rightarrow (t, \beta_{n-1})$ and $(t-1, \beta_0)$ and $(t, 0)$. With $0 \leq t < n$, there are n such independent pairs of g edges that may be used to merge the cycles. We will use the g edges in the column of B_n that has no faults. We now show that one can design an automorphism $\phi : B_n \rightarrow B_n$ which will avoid all faults. To construct ϕ, we compute constants c_i, $0 \leq i < n$ such that

$$c_i = \begin{cases} 1 & \text{if there is a fault in } f \text{ edge in column } i \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

One can then get K_0 by (6) as

$$K_0(\alpha^n + 1)\beta_{n-1}^{-1} = \sum_{j=0}^{n-1} c_{(-j)\ mod\ n}\alpha^j.$$

The other K_i values can then be inferred from (3). Theorem 5 then shows that the Hamilton cycle will use f edges in columns where f edges are fault free and g edges where f edges have faults. Thus the transformed Hamiltonian cycle will not have any faulty edges. ∎

Theorem 9 is interesting because it implies that up to 2^{n-1} edges of the same type may be faulty in up to $n-1$ columns and the faulty butterfly is still Hamiltonian. It is easy to extend this idea to any other mapping also. A direct result of Theorem 9 is the following result.

Corollary 1 *A butterfly with $n-1$ edge faults distributed one per column is Hamiltonian.*

Theorem 10 *If the edges in rows 0 and β_0 of B_n are fault-free, the faults in other rows $X \in S$ are restricted to g edges and those in rows $X \notin S$ are restricted to only one type of edges, then B_n is Hamiltonian.*

Proof. We prove the theorem by constructing a Hamiltonian cycle in B_n using only fault-free edges.

We begin with a cycle containing all nodes $(m, X) \in B_n$, $X \neq 0$ linked by f edges and another containing all nodes $(m, 0) \in B_n$, again using only f edges. (This is a procedure similar to that in [5].) We then merge these two cycles into a Hamiltonian cycle using g edges in rows 0 and β_0: $(t, 0) \rightarrow (t+1, \beta_{n-1})$ and $(t, \beta_0) \rightarrow (t+1, 0)$ for some t. If the faults in rows other than 0 and β_0 are only in g edges, then we already have the fault-free Hamiltonian cycle. If the faults in rows $X \notin S$ are in f edges, then applying the automorphism ψ to B_n would map them to fault-free g edges as stated in Theorem 7. Note that ψ maps the f edges in rows $X \in S$ to fault-free f edges, thus giving the required Hamiltonian cycle. ∎

Application of this theorem is illustrated in Fig. 1 which assumes that the butterfly B_4 has a large number of faults in the category specified by the Theorem 10. Note that in B_4, $S = \{0, \alpha, \alpha^6, \alpha^7, \alpha^8, \alpha^{10}, \alpha^{11}, \alpha^{14}\}$.

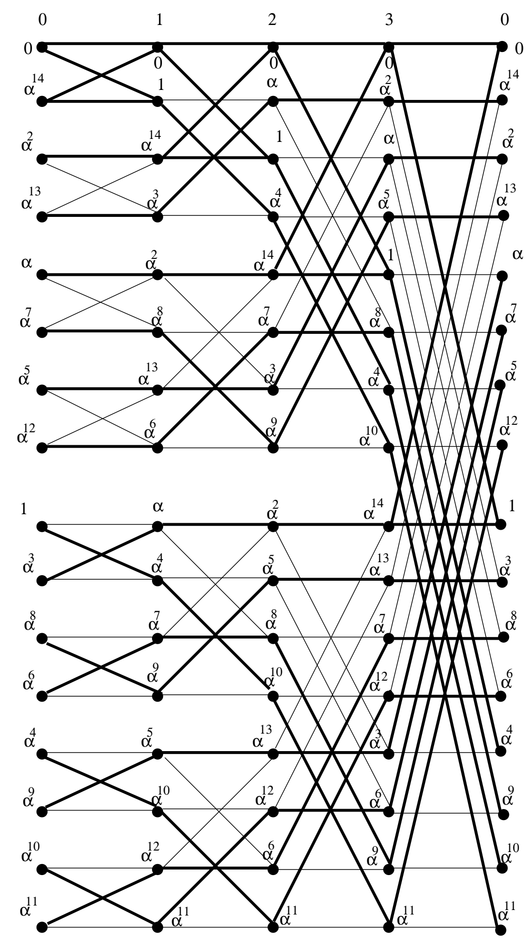

Fig. 1. Butterfly B_4 with faulty edges marked with light lines and fault-free edges with dark lines. The column numbers are at the top and the row index of each node is marked next to the node.

The fault-free Hamiltonian cycle is then obtained as:

$$
\begin{aligned}
&(0,\alpha^{14}) \to (1,1) \to (2,\alpha^4) \to (3,\alpha^{10}) \to (0,\alpha^{11}) \to \\
&(1,\alpha^{12}) \to (2,\alpha^6) \to (3,\alpha^7) \to (0,\alpha^8) \to (1,\alpha^9) \to \\
&(2,\alpha^5) \to (3,\alpha^{13}) \to (0,\alpha^3) \to (1,\alpha) \to (2,\alpha^2) \to \\
&(3,\alpha^{14}) \to (0,0) \to (1,0) \to (2,0) \to (3,0) \to \\
&(0,1) \to (1,\alpha^4) \to (2,\alpha^{10}) \to (3,\alpha^{11}) \to (0,\alpha^{12}) \to \\
&(1,\alpha^6) \to (2,\alpha^7) \to (3,\alpha^8) \to (0,\alpha^9) \to (1,\alpha^5) \to \\
&(2,\alpha^{13}) \to (3,\alpha^3) \to (0,\alpha) \to (1,\alpha^2) \to (2,\alpha^{14}) \to \\
&(3,1) \to (0,\alpha^4) \to (1,\alpha^{10}) \to (2,\alpha^{11}) \to (3,\alpha^{12}) \to \\
&(0,\alpha^6) \to (1,\alpha^7) \to (2,\alpha^8) \to (3,\alpha^9) \to (0,\alpha^5) \to \\
&(1,\alpha^{13}) \to (2,\alpha^3) \to (3,\alpha) \to (0,\alpha^2) \to (1,\alpha^{14}) \to \\
&(2,1) \to (3,\alpha^4) \to (0,\alpha^{10}) \to (1,\alpha^{11}) \to (2,\alpha^{12}) \to \\
&(3,\alpha^6) \to (0,\alpha^7) \to (1,\alpha^8) \to (2,\alpha^9) \to (3,\alpha^5) \to \\
&(0,\alpha^{13}) \to (1,\alpha^3) \to (2,\alpha) \to (3,\alpha^2) \to (0,\alpha^{14})
\end{aligned}
$$

If the edge faults are located differently, then one can use the automorphism $\psi(\cdot)$ with a different Hamiltonian cycle to obtain a fault-free mapping as the following theorem shows.

Theorem 11 *If the edges in rows $\sigma = (1+\alpha)^{-1}\beta_{n-1}$ and $\sigma + \beta_0$ of B_n are fault-free, the faults in other rows $X \in S$ are restricted to f edges and those in rows $X \notin S$, restricted to only one type of edges, then B_n is Hamiltonian.*

Proof. We use the original Hamiltonian cycle of Theorem 10. By applying a $\phi(\cdot)$ which flips edges in every column (Theorem 5), we get a Hamiltonian cycle which uses only g edges in all rows other than rows σ and $\sigma + \beta_0$. The rest of the proof runs parallel to the proof of Theorem 10. ∎

Theorems 10 and 11 require that two rows of B_n be fault-free. As shown in the next two theorems, this condition may be dropped if n is odd.

Theorem 12 *Let n be odd. If the faults in rows 0 and β_0 of B_n are restricted to the f edges, those in the other rows $X \in S$ to the g edges, and in rows $X \notin S$ to faults of only one type, then B_n is Hamiltonian.*

Proof. We first construct a Hamiltonian cycle as follows. Start from any node of the butterfly and choose the next node from a current node $(m, X) \in B_n$ using:

$$
\text{next node} = \begin{cases} (m+1, 0) & \text{if } X = \beta_0, \\ (m+1, \beta_{n-1}) & \text{if } X = 0 \text{ and} \\ (m+1, \alpha X) & \text{otherwise.} \end{cases} \qquad (15)
$$

It is easy to prove that the cycle from (15) is a Hamilton cycle. Further, nodes in rows 0 and β_0 in this cycle use only fault-free g edges. The rest of nodes use f edges. If they are fault free, we already have the fault-free Hamiltonian cycle. If faults in rows $X \notin S$ are restricted to edges of type f, then applying automorphism $\psi(\cdot)$ to this cycle will give the fault-free Hamiltonian cycle. ∎

Theorem 12 can be illustrated by mapping a Hamiltonian cycle in a faulty B_3 shown in Fig. 2. Note that in B_3, $S = \{0, 1, \alpha^4, \alpha^5\}$. The Hamiltonian cycle obtained from

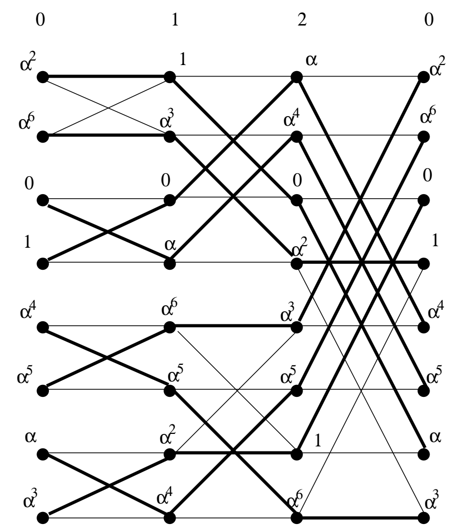

Fig. 2. Butterfly B_3 with faulty edges marked with light lines and fault-free edges with dark lines. The column numbers are at the top and the row index of each node is marked next to the node.

Theorem 12 is shown below.

$$
\begin{aligned}
&(0,\alpha^2) \to (1,1) \to (2,0) \to (0,\alpha) \to (1,\alpha^4) \to \\
&(2,\alpha^5) \to (0,\alpha^6) \to (1,\alpha^3) \to (2,\alpha^2) \to (0,1) \to \\
&(1,0) \to (2,\alpha) \to (0,\alpha^4) \to (1,\alpha^5) \to (2,\alpha^6) \to \\
&(0,\alpha^3) \to (1,\alpha^2) \to (2,1) \to (0,0) \to (1,\alpha) \to \\
&(2,\alpha^4) \to (0,\alpha^5) \to (1,\alpha^6) \to (2,\alpha^3) \to (0,\alpha^2)
\end{aligned}
$$

A similar result can also be derived by applying $\phi(\cdot)$ which flips all the edges of the cycle (15) to get the starting Hamiltonian cycle used in Theorem 12. We state the result below without proof.

Theorem 13 *Let n be odd. If the faults in rows σ and $\sigma + \beta_0$ of B_n are restricted to the g edges, those in the other rows $X \in S$ to the f edges, and in rows $X \notin S$ to faults of only one type, then B_n is Hamiltonian.*

Note that the symmetry of B_n will allow further generalization of Theorems 10 - 13.

We end this section by showing that one can also employ automorphisms $\phi(\cdot)$ and $\psi(\cdot)$ together to get even more powerful results.

Theorem 14 *If the edges in one of the columns of B_n are fault free, and the faults in each of the other columns are such that edges from $X \in S$ have one type of fault and those from $X \notin S$ have another type of fault. Then B_n is Hamiltonian.*

Proof. If the faulty edges from (m, X), are of type g when $X \in S$ and of type f if $X \notin S$, then applying ψ will map all of these faulty edges to type g in column $n - m$. On the other hand if the faulty edges from (m, X), are of type f when $X \in S$ and of type g if $X \notin S$, then applying ψ will map all of these faulty edges to type f in column $n - m$. Thus, after applying ψ, all the faulty edges in any column will be limited to only one type and there will be no faulty edges in one column. Theorem 9 can then be used to build the required Hamiltonian cycle using fault free edges. ∎

6 Conclusion

In the past, automorphisms have been used to map algorithms on architectures with (generally one) node fault. This paper shows that automorphisms can also be used to map algorithms on architectures with edge faults. To achieve this, we propose the use of an appropriate interconnection graph automorphism to map the set of faulty edges to free edges. Using an algebraic model, this paper has obtained all the $n2^{n+1}$ automorphisms of the wrapped butterfly B_n of dimension n. The resultant automorphisms are simple; they map the two coordinates of a node label independently. This simplicity allows one to determine the mapping of edges due to any automorphism. Conversely, it is also possible to design an automorphism to achieve the desired edge mapping. We have illustrated our technique by mapping a Hamilton cycle on a butterfly under various edge fault scenarios. We believe that having a large set of $n2^{n+1}$ simple automorphisms, each with a specific determined edge translation property makes this method applicable to a large number of mappings on faulty butterflies.

References

[1] F. Leighton, *Introduction to parallel algorithms and architectures: Arrays, trees, hypercubes*. M. Kaufman Pub., 1992.

[2] W. Lin, T. Sheu, C. R. Das, T. Feng, and C. Wu, "Fast data selection and broadcast on the butterfly network," in *Proc Workshop Future Trends Distrib Comput Syst 1990s*, pp. 65–72, 1990.

[3] A. Ranade, "Optimal speedup for backtrack search on a butterfly network," *Mathematical systems theory*, vol. 27, pp. 85–102, Jan 1994.

[4] E. J. Schwabe, "Constant-slowdown simulations of normal hypercube algorithms on the butterfly network," *Information processing letters*, vol. 45, p. 295, Apr 1993.

[5] M. D. Wagh and O. Guzide, "Mapping cycles and trees on wrap-around butterfly graphs," *SIAM J. on Comput.*, vol. 35, pp. 741–765, 2006.

[6] R. Feldmann and W. Unger, "The cube-connected-cycle is a subgraph of the butterfly network," *Parallel Processing Letters*, vol. 2, no. 1, pp. 13–19, 1992.

[7] O. Guzide and M. D. Wagh, "Extended butterfly networks," in *Proc. of the The 18th Int. Conf. on Parallel and Distributed Computing Systems*, (Las Vegas, NV), pp. 109–113, 2005.

[8] O. Guzide and M. D. Wagh, "Enhanced butterfly: A Cayley graph with node degree 5," in *Proc. of the The 20th Int. Conf. on Parallel and Distributed Computing Systems*, (Las Vegas, NV), pp. 224–229, 2007.

[9] P. Vadapalli and P. Srimani, "Fault tolerant ring embedding in tetravalent cayley network graphs," *J Circuits Syst Comput*, vol. 10, pp. 527–536, 1996.

[10] C.-H. Tsai, T. Liang, L.-H. Hsu, and M.-Y. Lin, "Cycle embedding in faulty wrapped butterfly graphs," *Networks*, vol. 42, no. 2, pp. 85–96, 2003.

[11] S.-C. Hwang and G.-H. Chen, "Cycles in butterfly graphs," *Networks*, vol. 35, no. 2, pp. 161–171, 2000.

[12] A. Avior, T. Calamoneri, S. Even, A. Litman, and A. L. Rosenberg, "A tight layout of the butterfly network," *Theory of Computing Systems*, vol. 31, pp. 475–488, Dec 1998.

[13] C.-H. Yeh, B. Parhami, E. A. Varvarigos, and H. Lee, "VLSI layout and packaging of butterfly networks," in *Proc. of ACM symp. on Parallel algorithms and architectures*, (New York, NY, USA), pp. 196–205, ACM, 2000.

[14] M. D. Wagh and K. Bendjilali, "Butterfly automorphisms and edge faults," in *Proc. of Int. Symp. on Paralel and Distributed Computing*, (Istanbul, Turkey), 2010.