Calamoneri T, Caminiti S, Petreschi R. A general approach to $L(h, k)$-label interconnection networks. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 23(4): 652–659 July 2008

A General Approach to $L(h,k)$-Label Interconnection Networks

Tiziana Calamoneri, Saverio Caminiti, and Rossella Petreschi

Department of Computer Science, Sapienza University of Rome, Italy

E-mail: {calamo, caminiti, petreschi}@di.uniroma1.it

Received March 22, 2007; revised April 24, 2008.

Abstract Given two non-negative integers h and k, an $L(h, k)$-*labeling* of a graph $G = (V, E)$ is a function from the set V to a set of colors, such that adjacent nodes take colors at distance at least h, and nodes at distance 2 take colors at distance at least k. The aim of the $L(h, k)$-labeling problem is to minimize the greatest used color. Since the decisional version of this problem is NP-complete, it is important to investigate particular classes of graphs for which the problem can be efficiently solved. It is well known that the most common interconnection topologies, such as Butterfly-like, Beneš, CCC, Trivalent Cayley networks, are all characterized by a similar structure: they have nodes organized as a matrix and connections are divided into layers. So we naturally introduce a new class of graphs, called $(l \times n)$-*multistage graphs*, containing the most common interconnection topologies, on which we study the $L(h, k)$-labeling. A general algorithm for $L(h, k)$-labeling these graphs is presented, and from this method an efficient $L(2, 1)$-labeling for Butterfly and CCC networks is derived. Finally we describe a possible generalization of our approach.

Keywords multistage interconnection network, $L(h, k)$-labeling, channel assignment problem

1 Introduction

Graph coloring is one of the main topics in graph theory. Many generalizations of the notion of graph coloring are motivated by problems of channel assignment in wireless communications, traffic phasing, fleet maintenance, task assignment, and other applications. (See [1] for a survey.)

Although in classical vertex coloring of graphs[2] a condition is imposed only on colors of adjacent nodes, many generalizations require colors to respect stronger conditions, e.g., restrictions are imposed on colors both of adjacent nodes and of nodes at distance 2 in the graph.

This paper will focus on a specific graph coloring generalization that arose first from a channel assignment problem in radio networks: the $L(h, k)$-*labeling problem*. This notion was introduced by Griggs and Yeh in the special case $h = 2$ and $k = 1$ in connection with the problem of assigning frequencies in a multihop radio network[3]. Formally:

Definition 1.1. *Given two non-negative integers h and k ($h \geqslant k$), an $L(h, k)$-labeling of an undirected graph $G = (V, E)$, is a function C from the node set V to a set of nonnegative integers, called colors, such that*:

1. $|C(x) - C(y)| \geqslant h$ *if* $\{x, y\} \in E$ (*i.e.*, *dist*$(x, y) = 1$) *and*
2. $|C(x) - C(y)| \geqslant k$ *if dist*$(x, y) = 2$

where the notation $\{x, y\}$ means an (undirected) edge and dist(x, y) is the length of the shortest path between x and y.

The aim of the $L(h, k)$-labeling problem is to minimize the greatest used color.

In the following we are interested in $L(h, k)$-*labeling* introduced in [4] as a special case of the notion of $L(m_1, \ldots, m_p)$-labeling introduced in [3] and define as follows.

Definition 1.2. *Given a positive integer p, and p non-negative integers $m_1 \geqslant m_2 \geqslant \cdots \geqslant m_p$, an $L(m_1, \ldots, m_p)$-labeling of a graph $G = (V, E)$, is a function C from the node set V to a set of non-negative integers such that $|C(x) - C(y)| \geqslant m_i$ if x and y are at distance i. The aim is to minimize the greatest used color.*

Some references about the notion of $L(h, k)$-labeling are in [4–9]. The three most studied particular cases of the $L(h, k)$-labeling problem are $L(1, 0)$-, $L(1, 1)$- and $L(2, 1)$-labeling. The first one corresponds to the classical vertex coloring of a graph, the second one corresponds to the vertex coloring of the square of a graph[10], and the third one is studied in connection with the channel assignment problem[11], in which

Short Paper

Work supported in part by Sapienza University of Rome (project "Parallel and Distributed Codes").

"close" transmitters have to be assigned different channels, and "very close" transmitters have to be assigned channels at least two apart.

In general, the decisional version of the $L(h,k)$-labeling problem is NP-complete even for small values of h and k[3]. Therefore, since the seminal work of Griggs and Yeh, researchers have produced a wide literature studying the problem for special values of h and k on special classes of graphs[12−21], for a survey see [22].

It is well known that the most common interconnection topologies, such as Butterfly-like, Beneš, CCC, Trivalent Cayley networks, are all characterized by a similar structure: they have nodes organized as a matrix and connections are divided into layers. So we naturally introduce a new class of graphs, called $(l \times n)$-*multistage graphs*, which contain the most common interconnection topologies. A general approach to $L(h,k)$-labeling these graphs is presented, and from this method an efficient $L(2,1)$-labeling for Butterfly and CCC networks is derived.

This paper is organized as follows. Section 2 is devoted to the description of a general approach for $L(h,k)$-labeling *collision free* $(l \times n)$-*multistage rows-bipartite graphs*. Section 3 presents an application to classical interconnection topologies. Two special examples for which our method gives results at most 1 far from the optimal are provided in Section 4: namely, we show an approximate $L(2,1)$-labeling of CCC and Butterfly networks. In Section 5 we describe how to generalize our approach to *collision free* $(l \times n)$-*multistage rows-p-partite graphs*, so extending the class of interconnection topologies for which we are able to give an $L(h,k)$-labeling; for this class we also provide an approach for the $L(m_1,\ldots,m_p)$-labeling problem. Possible future work and open problems are sketched in Section 6.

2 $L(h,k)$-Labeling Multistage Rows-Bipartite Graphs

In this section we define the $(l \times n)$-multistage graph and its quotient graph, then we focus our attention on the subclass of bipartite multistage graphs. So we build a class of graphs containing the most common interconnection topologies. Working on the partition of nodes into rows, we present a method of $L(h,k)$-labeling this class of graphs. Our approach reduces the problem of labeling a given multistage rows-bipartite graph of l rows to the problem of labeling a multistage rows-bipartite graph of only two rows, achieving a strong reduction of complexity.

Let us introduce some definitions as follows.

Definition 2.1. *Let V be a set of $l \times n$ nodes organized as a matrix; let $\mathcal{R} = \{R_1, R_2, \ldots, R_l\}$ be the set of the matrix rows and let $r_{i,j}$ be the j-th node of row R_i. An $(l \times n)$-*multistage graph *G is a simple loopless graph whose node set is V and such that the subgraph induced by each R_i $(i = 1, \ldots, l)$ is either a simple ordered path $r_{i,1}r_{i,2}\cdots r_{i,n}$ or a simple ordered cycle $r_{i,1}$ $r_{i,2}\cdots r_{i,n}$ $r_{i,1}$.*

Given an $(l \times n)$-multistage graph G, let us consider the *quotient graph* $G/\mathcal{R}$ resulting from the following contracting operation:

• each row R_i of G is a node i of $G/\mathcal{R}$;

• an edge connects nodes i and j of $G/\mathcal{R}$ if and only if there exists an edge (u, v) in G such that $u \in R_i$ and $v \in R_j$.

Definition 2.2. *An $(l \times n)$-multistage graph G is* rows-bipartite *if and only if its quotient graph $G/\mathcal{R}$ is bipartite. We call* Upper(G) *and* Lower(G) *the sets of rows of G corresponding to the two classes into which nodes of $G/\mathcal{R}$ are partitioned.*

Definition 2.3. *The* `reduced graph` *$R(G)$ of an $(l \times n)$-multistage rows-bipartite graph G is a simple graph with $2n$ nodes partitioned into two rows U and L containing nodes $u_1, u_2, \ldots, u_n$ and $l_1, l_2, \ldots, l_n$, respectively. The edges of $R(G)$ are:*

1. *all edges (u_i, u_{i+1}) and (l_i, l_{i+1}), $i = 1, \ldots, n-1$ (straight edges);*

2. *edge (u_n, u_1) if and only if at least one row in Upper(G) induces a cycle in G and edge (l_n, l_1) if and only if at least one row in Lower(G) induces a cycle in G (back edges);*

3. *edge (u_i, l_j) if and only if there exist two rows $R_a \in$ Upper(G) and $R_b \in$ Lower(G) such that $(r_{a,i}, r_{b,j})$ is an edge in G (cross edges).*

An example of a multistage graph is depicted in Fig.1(a). Figs.1(b) and 1(c) represent its quotient graph and its reduced graph respectively.

Remark 2.1. Definition 2.3 becomes much shorter if we define the mapping μ from nodes of G to nodes of $R(G)$ as follows:

$$\mu(r_{i,j}) = \begin{cases} u_j, & \text{if } R_i \in \mathit{Upper}(G); \\ l_j, & \text{if } R_i \in \mathit{Lower}(G). \end{cases}$$

In this way the edge set of $R(G)$ is $\{(\mu(x), \mu(y))$ s.t., (x, y) is an edge of $G\}$.

Observe that if two nodes x and y are adjacent, then $\mu(x) \neq \mu(y)$.

Definition 2.4. *We say that an $(l \times n)$-multistage rows-bipartite graph G is* `Collision Free` *if for any two nodes x and y at distance 2, it holds $\mu(x) \neq \mu(y)$.*

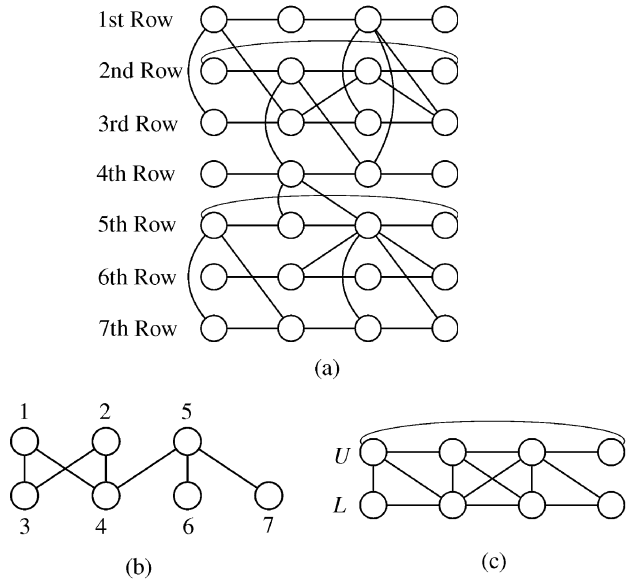

Fig.1. (a) (7×4)-multistage graph G. (b) Quotient graph G/R. (c) Reduced graph $R(G)$.

In the following we will refer to Collision Free $(l \times n)$-multistage rows-bipartite graphs as $(l \times n)$-CFMRB. It is also worth noticing that the graph depicted in Fig.1(a) is not collision free.

Theorem 2.1. *Let G be an $(l \times n)$-CFMRB and $R(G)$ be its reduced graph. If $R(G)$ admits an $L(h, k)$-labeling using colors from* 0 *to c then G admits an $L(h, k)$-labeling using colors from* 0 *to c.*

Proof. Let C_R be a feasible $L(h, k)$-labeling for $R(G)$ using colors from 0 to c. A labeling C_G for G can be derived from C_R using the mapping μ: $C_G(r_{i,j}) = C_R(\mu(r_{i,j}))$. By definition of $R(G)$, if x, y are adjacent nodes in G then $\mu(x), \mu(y)$ are adjacent nodes in $R(G)$, and if x, y are at distance 2 in G then $\mu(x), \mu(y)$ are either adjacent or at distance 2 in $R(G)$; indeed, G is collision free and then it cannot be $\mu(x) = \mu(y)$.

As a consequence, since $h \geqslant k$, C_G is a feasible $L(h, k)$-labeling for G. □

The previous theorem provides a simple algorithmic technique for computing an $L(h, k)$-labeling of any $(l \times n)$-CFMRB. The computational complexity of this technique depends on the time necessary to compute the $L(h, k)$-labeling of $R(G)$: the worst case happens when we use an exhaustive approach, and this requires time exponent in n. It is to notice that if l is exponential in n, this time complexity is polynomial (and often linear) in the number of nodes of G.

Remark 2.2. For any graph G, with maximum degree $\Delta(G)$, the number of necessary colors to $L(h, k)$-label G is never less than $h + (\Delta(G) - 1)k + 1$. Since graph $R(G)$ may be dense even if graph G is sparse, this implies that $\Delta(R(G))$ may be much greater than $\Delta(G)$ and then the greatest used color is not necessarily minimized.

3 Interconnection Topologies

An *interconnection network* consists of a collection of processors or routing-switches with direct connections between them. An *interconnection topology* is the undirected graph underlying an interconnection network: it has a node for each processor/switch and an edge between each pair of connected processors/switches. A large variety of interconnection networks have been investigated; the interested reader can refer to [23] for a wide survey.

In this section we show how to exploit Theorem 2.1 to $L(h, k)$-label interconnection topologies. Indeed, looking at the classical representation of the most common interconnection topologies, it is quite simple to see that most of them belong to the class of multistage graphs. See for example the classical two-dimensional layout of the Cube-Connected-Cycles (CCC) network[24] in Fig.2(a) and the classical representation of the Butterfly network[23] in Fig.2(b).

Remark 3.1. In Definition 2.1 the nodes of the graph are organized as a matrix; in the following we choose the matrix naturally induced by the classical representation of an interconnection topology.

This representation is not always suitable for obtaining a multistage graph. Indeed some interconnection topologies — such as Flip, Omega, Baseline and Reverse Baseline networks (see [23], pp.731–735) — are not multistage graphs if the matrix is chosen with the criterion in Remark 3.1, because their rows are neither simple paths nor cycles. Nevertheless, we are able to produce a matrix organization suitable for obtaining a multistage graph applying the same mapping used for showing that these graphs are all topological equivalent to a Butterfly network[25].

Moreover, most of the interconnection topologies are also rows-bipartite. Only few of them — such as the 3-ary Butterfly (see [23], pp. 740) and Trivalent Cayley networks[26,27] — have not this property. Indeed, their reduced graphs are 3- and 4-partite, respectively. We will deal with these interconnection topologies in Section 5. Finally, it is straightforward to see that all these interconnection topologies are collision free.

Theorem 2.1 allows us to $L(h, k)$-label each interconnection topology G that is $(l \times n)$-CFMRB, in the following simple way.

1) Compute the reduced graph $R(G)$ associated with G;

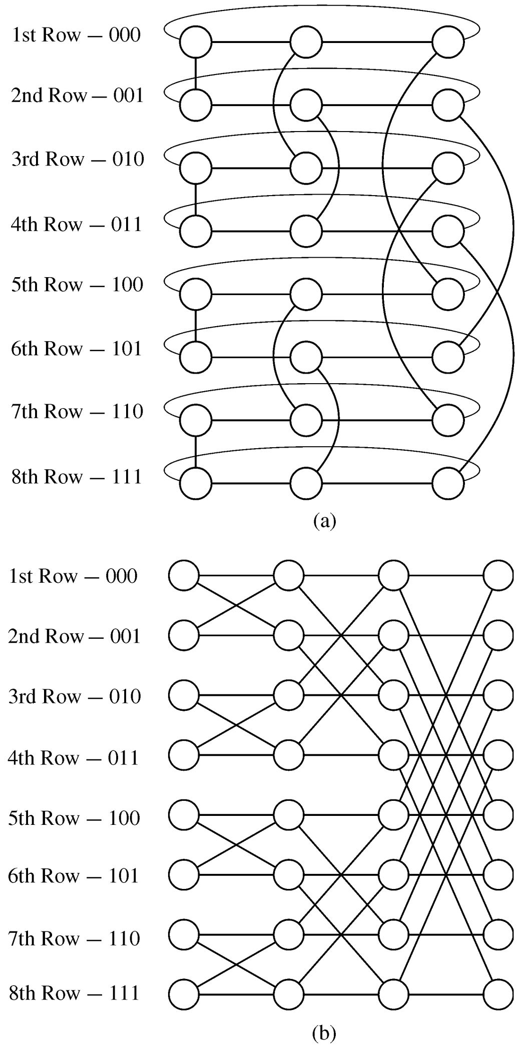

Fig.2. (a) Two-dimensional layout of a 3-dimensional CCC network. (b) Classical representation of an 8-input Butterfly network.

2) Compute an $L(h,k)$-labeling of $R(G)$;

3) Assign to each node $r_{i,j}$ of G the color of its corresponding node $\mu(r_{i,j})$ of $R(G)$.

It is not difficult to see that Steps 1) and 3) work in linear time with respect to the size of G. We remark that we are dealing with interconnection topologies whose maximum degree is bounded by a constant, then the size of G is linear with respect to the number of its nodes.

The complexity of Step 2) is related to how the algorithm for $L(h,k)$-labeling $R(G)$ works; however, even if the algorithm searches the solution in an exhaustive way, this step requires time exponent in n, where n is the number of columns of G. Since we are considering interconnection topologies whose number l of rows is exponential with respect to the number n of the columns, this step is polynomial with respect to the size of G.

The worth of our approach consists in coloring a graph of only $2n$ nodes instead of a graph of $O(n2^n)$ nodes.

We have to highlight that the considered interconnection topologies are often characterized by a high symmetry. This implies that the density of their reduced graphs is of the same order of their density and then, despite the Remark 2.2, the size of our solution is nearly optimal. The results presented in the next section will confirm this statement.

4 Direct Labeling of Two Particular Networks

In this section, applying the algorithm stated in Section 3, we provide a labeling scheme in the special case $h = 2$ and $k = 1$ for two well-known interconnection topologies: Cube-Connected-Cycles and Butterfly networks.

The special structure of the reduced graph of both these topologies allows us to avoid the exhaustive approach and to give a direct labeling scheme. Finally, we prove that these two $L(2,1)$-labeling schemes use a number of colors at most 1 far from optimal.

4.1 $L(2,1)$-Labeling CCC Networks

The *n-dimensional Cube-Connected-Cycles* network, CCC_n, is constructed from the n-dimensional hypercube by replacing each node of the hypercube with a cycle of n nodes. The i-th dimension edge incident to a node of the hypercube is connected to the i-th node of the corresponding cycle of the CCC. In Fig.2(a) the classical representation of a 3-dimensional CCC network is depicted. The resulting graph has $n2^n$ nodes each with degree 3.

This degree gives us a trivial lower bound on the number of colors needed to $L(2,1)$-label a CCC network, that is 6. Indeed, let v be a node colored with a color $C(v)$ different from both 0 and the maximum used color. The 3 nodes adjacent to v must have colors different from each other and at distance 2 from $C(v)$, i.e., different from $C(v)-1$, $C(v)$ and $C(v)+1$. It follows that at least 6 colors are necessary.

In this subsection we show how to $L(2,1)$-label a CCC_n with at most 7 colors.

The quotient graph of a CCC_n is the n-dimensional hypercube, that is bipartite; hence the 2^n cycles of G can be easily partitioned into sets *Upper*(G) and *Lower*(G). The reduced graph is then constituted by two n length cycles U and L in which each u_i is connected with the corresponding l_i (see Fig.3).

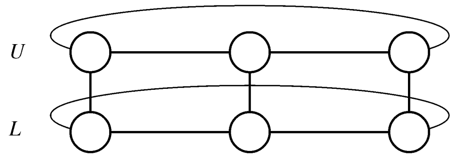

Fig.3. Reduced graph of a 3-dimensional CCC network.

The following direct $L(2,1)$-labeling scheme is expressed as a pair of n length strings S_U and S_L, each representing the sequence of colors assigned respectively to $u_1, u_2, \ldots, u_n$ and $l_1, l_2, \ldots, l_n$.

Lemma 4.1. *Given an n-dimensional Cube-Connected-Cycles network G, there exists an $L(2,1)$-labeling of the reduced graph of G that uses 6 colors if n is multiple of 3, and 7 colors otherwise.*

Proof. We give a constructive proof. If $n \leqslant 2$ the labeling is trivial, then assume $n \geqslant 3$. First, let us consider the case n multiple of 3, i.e., $n = 3q$ $(q \geqslant 1)$. $S_U = (024)^q$ and $S_L = (351)^q$, where the power means that the string is repeated q times.

Now, let $n = 3q + r$ $(1 \leqslant r \leqslant 2)$; then $S_U = 026x4(025)^{q-1}$ and $S_L = 640y2(641)^{q-1}$, where

$$x = \begin{cases} \epsilon, & \text{if } r = 1; \\ 1, & \text{if } r = 2; \end{cases} \qquad y = \begin{cases} \epsilon, & \text{if } r = 1; \\ 5, & \text{if } r = 2; \end{cases}$$

and ϵ is the empty string. Notice that if $q = 1$ the repeated substrings do not appear, i.e., $S_U = 026x4$ and $S_L = 640y2$.

It is easy to verify that the provided $L(2,1)$-labelings are feasible and that they use either 6 or 7 colors, depending on the value of n. □

Exploiting Theorem 2.1, it is possible to immediately transfer this $L(2,1)$-labeling to CCC_n. Observe that, as the proof of Lemma 4.1 gives a direct labeling of each node depending only on its own position in $R(G)$ and on n, the algorithm can be run on CCC_n in distributed constant time, provided that each node knows its own position.

The trivial lower bound proved for CCC_n holds also for its reduced graph, as it is regular of degree 3. Nevertheless, supported by experimental results, we conjecture that 7 colors are also necessary for $R(G)$, if n is not multiple of 3. On the other hand, for what concerns CCC_n, we experimentally verified that there exist some values of n (e.g., $n = 5$) admitting a 6-colors $L(2,1)$-labeling, even if its reduced graph needs 7 colors.

4.2 $L(2,1)$-Labeling Butterfly Networks

The *N-input Butterfly network* B_N (with N power of 2) has $N(\log_2 N + 1)$ nodes. The nodes correspond to pairs (i, j), where i $(0 \leqslant i < N)$ is a binary number and denotes the row of the node, and j $(0 \leqslant j \leqslant \log_2 N)$ denotes its column. Two nodes (i, j) and (i', j') are connected by an edge if and only if $j' = j + 1$ and either i and i' are identical (*straight edge*) or i and i' differ in precisely the j'-th bit (*cross edge*).

A Butterfly network, depicted in Fig.2(b), is an $(N \times (\log_2 N + 1))$-multistage graph whose quotient graph is the $(\log_2 N)$-dimensional hypercube, and hence is rows-bipartite. Its reduced graph $R(B_N)$ is constituted by two n length paths U and L in which each u_i (or l_i) is connected with both l_{i-1} (or u_{i-1}) and l_{i+1} (or u_{i+1}), if they exist (see Fig.4).

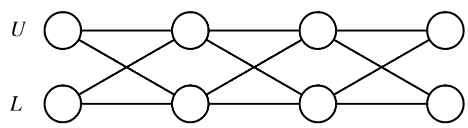

Fig.4. Reduced graph of an 8-input Butterfly network.

A trivial lower bound on the number of colors needed to $L(2,1)$-label a Butterfly network is 7. This value can be obtained with considerations similar to the one used for CCC network, since all internal nodes of B_N have degree 4.

Also for B_N, as we will show in the following, it is possible to give a direct $L(2,1)$-labeling using a number of colors at most 1 far from optimal. More in detail, if N is either 2^2 or 2^3, we provide a labeling using 7 colors, that is optimal; for all greater values of N our method requires 8 colors.

As in the previous subsection, the following $L(2,1)$-labeling of $R(B_N)$ is expressed as a pair of $\log_2 N + 1$ length strings S_U and S_L.

Lemma 4.2. *Given an N-input Butterfly network G, there exists an $L(2,1)$-labeling of the reduced graph of G that uses 7 colors if N is either 2^2 or 2^3, and 8 colors otherwise.*

Proof. Let us initially consider $N = 2^2$ and $N = 2^3$; in the first case $S_U = 250$ and $S_L = 361$ and in the second case $S_U = 2503$ and $S_L = 3614$.

For all greater values of N, the strings S_U and S_L are built by repeating the pattern 036 and 147, respectively, until the paths are completely labeled; of course, the last occurrences of the patterns could not be complete. More formally, let $\log_2 N + 1 = 3q + r$ $(0 \leqslant r \leqslant 2)$; then $S_U = (036)^q x$ and $S_L = (147)^q y$, where

$$x = \begin{cases} \epsilon, & \text{if } r = 0; \\ 0, & \text{if } r = 1; \\ 03, & \text{if } r = 2; \end{cases} \qquad y = \begin{cases} \epsilon, & \text{if } r = 0; \\ 1, & \text{if } r = 1; \\ 14, & \text{if } r = 2. \end{cases}$$

It is easy to verify that the provided $L(2,1)$-labelings are feasible and that they use 8 colors, unless $N = 2^2$ or $N = 2^3$, in which cases 7 colors are sufficient. □

This direct labeling can be computed on B_N in distributed constant time, provided that each node knows its own position. The provided $L(2,1)$-labeling of $R(B_N)$ is optimal, as proven by the following reasoning. If $N < 2^4$ the claim can be directly proven. Otherwise, by contradiction suppose that the optimal number of colors is 7 and, for any $2 \leqslant j \leqslant \log_2 N - 2$, consider the six nodes u_{j-1}, u_j, u_{j+1}, l_{j-1}, l_j and l_{j+1}. As these nodes are at mutual distance $\leqslant 2$, they receive different colors and at least one of them is colored with color c, where $2 \leqslant c \leqslant 4$. Without loss of generality, let u_j be such a node. Of course, nodes adjacent to u_j, i.e., u_{j-1}, u_{j+1}, l_{j-1} and l_{j+1} receive the four colors different from $c-1$, c, $c+1$. It follows that node l_j must be colored with either $c-1$ or $c+1$. With reasonings identical to the previous ones, nodes adjacent to l_j, i.e., u_{j-1}, u_{j+1}, l_{j-1} and l_{j+1} receive the four colors different either from $c-2$, $c-1$, c, or from c, $c+1$, $c+2$. In both cases we get a contradiction.

5 $L(m_1, \ldots, m_p)$-Labeling Multistage Rows-p-Partite Graphs

As we have already noted (cf. Section 3), some interconnection topologies are not multistage rows-bipartite, since their reduced graphs are not bipartite. In this section we will introduce a generalization of our technique in order to $L(m_1, \ldots, m_{p'})$-label, $2 \leqslant p' \leqslant p$, also those topologies whose reduced graph is p-partite. This leads to an $L(h,k)$-labeling of these graphs in the special case $p' = 2$.

Definition 5.1. *An $(l \times n)$-multistage graph G is rows-p-partite if and only if its quotient graph $G/\mathcal{R}$ is p-partite. We call $S_1, S_2, \ldots, S_p$ the sets of rows of G corresponding to the classes in which nodes of $G/\mathcal{R}$ are partitioned.*

Definition 5.2. *The reduced graph $R(G)$ of an $(l \times n)$-multistage rows-p-partite graph G is a graph with $p \times n$ nodes partitioned into p rows containing nodes $v_1^i, v_2^i, \ldots, v_n^i$ $(1 \leqslant i \leqslant p)$. The set of edges of $R(G)$ is $\{(\mu(x), \mu(y))$ s.t. (x,y) is an edge of $G\}$ where:*

$$\mu(r_{a,b}) = v_b^i \quad s.t.\ R_a \in S_i.$$

Definition 5.3. *An $(l \times n)$-multistage rows-p-partite graph G is p'-`Collision Free` $(2 \leqslant p' \leqslant p)$ if for any two nodes x and y at distance less than or equal to p', it holds $\mu(x) \neq \mu(y)$.*

Notice that, if G is p'-Collision Free, nodes at distance $d \leqslant p'$ in G are mapped into nodes at distance no longer than d in $R(G)$ by function μ. This ensures that a feasible $L(m_1, \ldots, m_{p'})$-labeling $(p' \leqslant p)$ of $R(G)$ can be transposed on G by way of μ, justifying the following theorem.

Theorem 5.1. *Let G be a p'-Collision Free $(l \times n)$-multistage rows-p-partite graph and $R(G)$ its reduced graph. If $R(G)$ admits an $L(m_1, \ldots, m_{p''})$-labeling $(2 \leqslant p'' \leqslant p')$ using colors from 0 to c then G admits an $L(m_1, \ldots, m_{p''})$-labeling using colors from 0 to c.*

The time complexity for labeling $R(G)$ grows up with p, namely, when an exhaustive approach is used, it requires time exponent in pn.

Let us now consider the application of Theorem 5.1 to interconnection topologies. Since in general p is limited by a constant, the complexity for labeling these topologies remains linear in the number of nodes of graph G, then we are able to efficiently label networks such as N-input s-ary Butterfly and n-dimensional Trivalent Cayley networks. In Fig.5 the 3-ary Butterfly network and its reduced graph are shown.

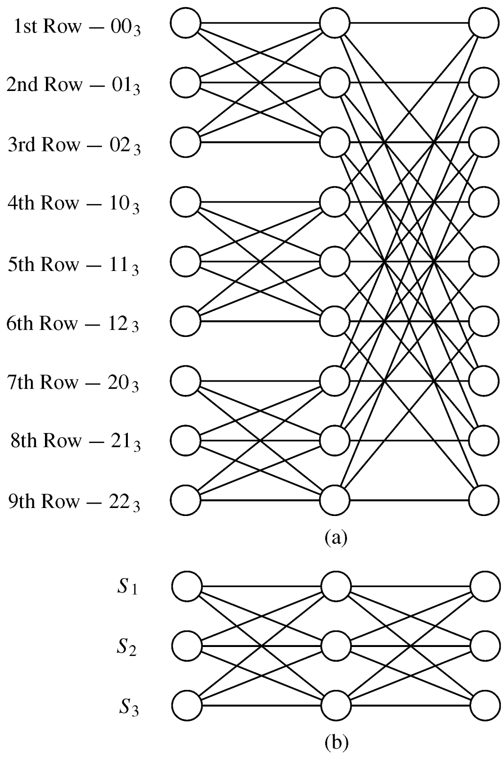

Fig.5. (a) Classical representation of a 9-input 3-ary Butterfly network G. (b) Reduced graph $R(G)$.

6 Conclusions and Open Problems

In this paper we have dealt with the $L(h,k)$-labeling problem. The decisional version of this problem has been shown to be NP-complete, then it is important to investigate particular classes of graphs for which this problem can be efficiently solved. In this context we have introduced the new class of p'-Collision Free $(l \times n)$-multistage rows-p-partite graphs, including most of the well-known interconnection topologies, and we have studied the $L(m_1, \ldots, m_{p''})$-labeling problem, $2 \leqslant p'' \leqslant p'$, on this class providing a general approach to efficiently label these multistage graphs.

Our method allows us to label the most common interconnection topologies in linear time, although a good approximation ratio is not guaranteed. In practice this approach gives good results as shown in Section 4. As an example, we have presented how to $L(2,1)$-label CCC and Butterfly networks in distributed constant time using a number of colors at most 1 far from the optimal. We are trying to apply this method to other classes of graphs.

It remains an open problem to show if a constant approximation ratio can be guaranteed (even for subclasses), exploiting some special properties studied for interconnection topologies; for example, the Buddy Property[23] could guarantee the sparsity of the reduced graph.

Finally, it would be interesting to investigate if the decisional version of the $L(m_1, \ldots, m_p)$-labeling problems remains NP-complete for the classical interconnection topologies.

Acknowledgments We are grateful to all the anonymous referees for their advices to help improve the readability of the paper, and we would like to thank in particular the referee who suggested the proof of optimality of the algorithm for labeling $R(B_N)$.

References

[1] Roberts F S. From Garbage to Rainbows: Generalizations of Graph Coloring and Their Applications. Graph Theory, Combinatorics, and Applications, 2, Alavi Y, Chartrand G, Oellermann O R, Schwenk A J (eds.), 2, New York: Wiley, 1991, pp.1031–1052.

[2] Jensen T R, Toft B. Graph Coloring Problems. New York: John Wiley & Sons, 1995.

[3] Griggs J R, Yeh R K. Labeling graphs with a condition at distance 2. *SIAM Journal on Discrete Mathematics* 1992, 5(4): 586–595.

[4] Georges J P, Mauro D W. Generalized vertex labelings with a condition at distance two. *Congressus Numerantium*, 1995, 109: 141–159.

[5] Calamoneri T. Exact solution of a class of frequency assignment problems in cellular networks. *Discrete Mathematics & Theoretical Computer Science*, 2006, 8: 141–158.

[6] Calamoneri T, Pelc A, Petreschi R: Labeling trees with a condition at distance two. *Discrete Mathematics*, 2006, 306(14): 1534–1539.

[7] Georges J P, Mauro D W. Labeling trees with a condition at distance two. *Discrete Mathematics*, 2003, 269(1-3): 127–148.

[8] Georges J P, Mauro D W, Stein M I. Labeling products of complete graphs with a condition at distance two. *SIAM Journal on Discrete Mathematics*, 2001, 14(1): 28–35.

[9] J van den Heuvel, R A Leese, M A Shepherd. Graph labelling and radio channel assignment. *Journal of Graph Theory*, 1998, 29(4): 263–283.

[10] Agnarsson G, Halldórsson M M. Coloring powers of planar graphs. *SIAM Journal on Discrete Mathematics*, 2003, 16(4): 651–662.

[11] Hale W K. Frequency assignment: Theory and application. *Proceedings of the IEEE*, 1980, 68(12): 1487–1514.

[12] K I Aardal, S P M van Hoesel, A M C A Koster, C Mannino, A Sassano. Models and solution techniques for frequency assignment problems. ZIB-Report 01-40, Konrad-Zuse-Zentrum fur Informationstechnik Berlin, 2001.

[13] H L Bodlaender, T Kloks, R B Tan, J van Leeuwen. Lambda coloring of graphs. In *Proc. 17th Annual Symposium on Theoretical Aspects of Computer Science* (*STACS 2000*), Lille, France, *LNCS* 1770, Feb. 2000, pp.395–406.

[14] Calamoneri T, Petreschi R. $L(2,1)$-labeling of planar graphs. *Journal on Parallel and Distributed Computing*, 2004, 64(3): 414–426.

[15] Calamoneri T, Petreschi R. λ-coloring matrogenic graphs. *Discrete Applied Mathematics*, 2006, 154: 2445–2457.

[16] Chang G J, Kuo D. The $L(2,1)$-labeling problem on graphs. *SIAM Journal on Discrete Mathematics*, 1996, 9: 309–316.

[17] Fiala J, Kloks T, Kratochvíl J. Fixed-parameter complexity of λ-colorings. In *Proc. Graph Theoretic Concepts in Computer Science* (*WG'99*), *LNCS* 1665, 1999, pp.350–363.

[18] Murphey R A, Pardalos P M, Resende M G C. Frequency Assignment Problems. Handbook of Combinatorial Optimization, Du D-Z, Pardalos P M (eds.), Kluwer Academic Publishers, 1999, pp.295–377.

[19] Sakai D. Labeling chordal graphs: Distance 2 condition. *SIAM Journal on Discrete Mathematics*, 1994, 7(1): 133–140.

[20] Shepherd M. Radio channel assignment [Dissertation]. Merton College, Oxford, 1998.

[21] Whittlesey M A, Georges J P, Mauro D W. On the λ number of Q_n and related graphs. *SIAM Journal on Discrete Mathematics*, 1995, 8(4): 499–506.

[22] Calamoneri T. The $L(h,k)$-labelling problem: A survey and annotated bibliography. *The Computer Journal*, 2006, 49(5): 585–608, http://www.dsi.uniroma1.it/~calamo/survey.html.

[23] Leighton F T. Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes. Morgan Kaufmann, 1992.

[24] Preparata F P, Vuillemin J. The cube-connected cycles: A versatile network for parallel computation. *Communications of ACM*, 1981, 24(5): 300–309.

[25] Calamoneri T, Massini A. Efficient algorithms for checking the equivalence of multistage interconnection networks. *Journal on Parallel and Distributed Computing*, 2004, 64(1): 135–150.

[26] Calamoneri T, Petreschi R. A new 3D representation of trivalent Cayley networks. *Information Processing Letters*, 1997, 61(5): 247–252.

[27] Vadapalli P, Srimani P K. Trivalent Cayley graphs for interconnection networks. *Information Processing Letters*, 1995, 54(6): 329–335.

Tiziana Calamoneri graduated in mathematics in 1992 and obtained her Ph.D. degree in computer science 1997, at Sapienza University of Rome, Italy. She was a post-doctoral fellow at the Italian National Research Council; then she became assistant professor, and since 2006 she is an associate professor at the Department of Computer Science, Sapienza University of Rome. Her research interests include graph algorithms, two- and three-dimensional graph drawing, layout of networks topologies and optimal routing schemes, channel assignment in wireless networks.

Saverio Caminiti graduated in computer science in 2003 and obtained his Ph.D. degree in computer science in 2008, at Sapienza University of Rome, Italy. He currently has a post-doctoral position. His research interests include graph algorithms, combinatorial mappings, layout of network topologies.

Rossella Petreschi graduated in mathematics in 1972. From 1973 to 1988 she was, first, with the Italian National Research Council, then with the Department of Mathematics at University of L'Aquila, Italy, and finally with the Department of Mathematics at Sapienza University of Rome. Since its constitution in 1989, she has been with the Department of Computer Science at Sapienza University of Rome, where she is currently a full professor in computer science. Her current research interests include the design of sequential and parallel algorithms to solve problems, arising from the project of computer science systems, that find natural description in terms of graph theory.