PARALLEL ALGORITHM FOR RADIOCOLORING A GRAPH

N. Deo
University of Central Florida

PARALLEL ALGORITHM FOR RADIOCOLORING A GRAPH

Hemant Balakrishnan and Narsingh Deo
School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 32816-2362

{hemant, deo}@cs.ucf.edu

Abstract

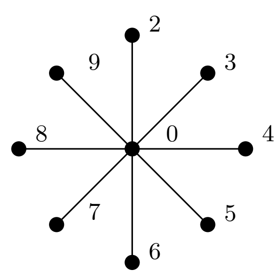
Given an undirected graph $G(V, E)$ with vertex set V and edge set E, the *Radiocoloring* of G is defined as a function $f : V \to N$ such that for all pairs of vertices x, y in G $|f(x) - f(y)| \geq 2$ when $d(x, y) = 1$ and $|f(x) - f(y)| \geq 1$ when $d(x, y) = 2$, where $d(x, y)$ is the distance between the vertices x and y and N is the set of non-negative integers. The range of numbers used is called a *span*. The radiocoloring problem consists of determining the minimum span for a given graph G. This minimum span of G is called the *radiochromatic number*, λ of G. In this paper, we discuss Nordhaus-Gaddum type result for the sum and product of the radiochromatic number of a graph and that of its complement. We also propose an approximate parallel algorithm for radiocoloring. Our algorithm is based on the largest-degree-first coloring heuristic.

Keywords: Radiocoloring, $L(2, 1)$ labelling, Nordhaus-Gaddum, Parallel algorithm.

1 INTRODUCTION

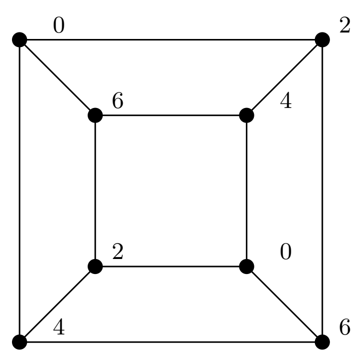
There has been a considerable growth of wireless networks in the last two decades. Wireless networks must make use of the limited radio spectrum through frequency reuse. The *frequency assignment problem* (FAP) deals with assigning channels (*i.e.* frequencies) to radio stations while keeping the interference to a minimum. According to [13], F. S. Roberts (in a private communication with Griggs and Yeh) first suggested assigning radio channels to transmitters, represented by nonnegative integers such that channels assigned to adjacent transmitters be at least two units apart, and all pairs of transmitters at a distance of two have distinct channels assigned to them. The radiocoloring problem is a graph-theoretic approach to solving the FAP.

DEFINITION 1. *A radiocoloring of a graph G is an assignment to each node, one of the numbers $0, 1, 2, \dots, \lambda$ called colors such that all pairs of adjacent vertices, get colors which differ by at least two and no pair of vertices at distance two, get the same color. The radiocoloring problem (RCP) consists of determining the minimum λ for a given graph.*



$\lambda = 9$

Example 1: Radiocoloring of a star $K_{1,8}$



$\lambda = 6$

Example 2: Radiocoloring of a 3-dimensional cube

The first condition (every adjacent pair of nodes have colors that are at least two units apart) makes sure that there is a *guard band* between channels assigned to adjacent radio stations.

DEFINITION 2. *A guard band is an unused frequency band between two adjacent channels in wireless communication to minimize the interference between the channels.*

The second condition (pair of nodes at a distance of two have different colors) makes sure that the radio stations with a common neighbor, say u, do not communicate with u using the same channel.

The RCP was first introduced by Griggs and Yeh [13] as the $L(2,1)$ labelling problem. The RCP in general has been proved to be NP-complete [13]. However exact results have been obtained for certain special class of graphs such as paths [21], cycles, trees [13], cacti, unicycles, bicycles [14], cliques, stars [4]; and approximate bounds have been obtained for certain others such as bipartite graphs, outerplanar graphs, split graphs [1], chordal graphs [18], hypercubes [20], planar graphs [14], unigraphs [3].

The rest of the paper is organised as follows: Section 2 deals with Nordhaus-Gaddum type result for the sum and product of the radiochromatic number of a graph and that of its complement. In Section 3 we propose an algorithm based on the largest-degree-first coloring heuristic to radiocolor a given graph and then parallelize it to obtain a cost-optimal, approximate parallel algorithm. Finally in Section 4 we give our conclusions and discuss further work.

Throughout this paper, G will denote a simple undirected graph and $\overline{G}$, its complement; n, the number of vertices; K_n, a complete graph on n vertices; χ, the chromatic number; and λ, the radiochromatic number.

2 NORDHAUS-GADDAM TYPE RESULTS

Nordhaus and Gaddum [17] obtained the upper and lower bounds on the sum and product of the chromatic number of a graph and that of its complement. We reproduced their results here for the sake of completeness.

THEOREM 1. (Nordhaus and Gaddum [17]) *If G and $\overline{G}$ are complementary graphs on n nodes, their chromatic numbers, $\chi(G)$ and $\chi(\overline{G})$, satisfy the following:*

$$\begin{aligned} 2\sqrt{n} \;&\leq\; \chi(G)+\chi(\overline{G}) \;\leq\; n+1, \;\; \textit{and} \\ n \;&\leq\; \chi(G)\cdot\chi(\overline{G}) \;\leq\; \left(\frac{n+1}{2}\right)^2. \end{aligned}$$

We derive similar bounds on the radiochromatic number of a graph and that of its complement.

THEOREM 2. *If G and $\overline{G}$ are complementary graphs on n nodes, their radiochromatic numbers, $\lambda(G)$ and $\lambda(\overline{G})$, satisfy the following:*

$$\begin{aligned} 2\sqrt{n}-2 \;&\leq\; \lambda(G)+\lambda(\overline{G}) \;\leq\; 3n-3, \;\; \textit{and} \\ 0 \;&\leq\; \lambda(G)\cdot\lambda(\overline{G}) \;\leq\; \left(\frac{3n-3}{2}\right)^2. \end{aligned}$$

PROOF: Griggs and Yeh [13] proved that for a graph G with n nodes

$$\lambda(G) \quad \leq \quad n + \chi(G) - 2,$$

which can also be written for the complement $\overline{G}$,

$$\lambda(\overline{G}) \quad \leq \quad n + \chi(\overline{G}) - 2.$$

Adding the preceding two inequalities, we get

$$\lambda(G) + \lambda(\overline{G}) \quad \leq \quad 2n - 4 + \chi(G) + \chi(\overline{G}). \tag{1}$$

In [17] Nordhaus and Gaddum have shown that $\chi(G) + \chi(\overline{G}) \ \leq \ n + 1$. Substituting this inequality into inequality (1), we get

$$\lambda(G) + \lambda(\overline{G}) \quad \leq \quad 3n - 3. \tag{2}$$

Which gives us an upper bound on $\lambda(G) + \lambda(\overline{G})$.

To compute the lower bound on the sum, $\lambda(G) + \lambda(\overline{G})$, we use the relationship between radiochromatic number of a graph G and its chromatic number obtained by Chang and Kuo [6]. They show that

$$\chi(G) - 1 \quad \leq \quad \lambda(G),$$

which can also be written for the complement $\overline{G}$

$$\chi(\overline{G}) - 1 \quad \leq \quad \lambda(\overline{G}).$$

Adding the preceding two inequalities, we get

$$\chi(G) + \chi(\overline{G}) - 2 \quad \leq \quad \lambda(G) + \lambda(\overline{G}). \tag{3}$$

In [17] Nordhaus and Gaddum have shown that $2\sqrt{n} \ \leq \ \chi(G) + \chi(\overline{G})$. Substituting this inequality into inequality (3), we get

$$2\sqrt{n} - 2 \quad \leq \quad \lambda(G) + \lambda(\overline{G}). \tag{4}$$

Combining inequalities (2) and (4) we get,

$$2\sqrt{n} - 2 \quad \leq \quad \lambda(G) + \lambda(\overline{G}) \quad \leq \quad 3n - 3.$$

Now, in order to determine the upper bound on the product, $\lambda(G) \cdot \lambda(\overline{G})$, we observe that

$$0 \quad \leq \quad (\lambda(G) - \lambda(\overline{G}))^2 \quad = \quad (\lambda(G) + \lambda(\overline{G}))^2 - 4\lambda(G) \cdot \lambda(\overline{G}),$$

which implies

$$4\lambda(G) \cdot \lambda(\overline{G}) \quad \leq \quad (\lambda(G) + \lambda(\overline{G}))^2.$$

Substituting the upper bound on $\lambda(G) + \lambda(\overline{G})$ from inequality (2) we get

$$\lambda(G) \cdot \lambda(\overline{G}) \quad \leq \quad \left(\frac{3n-3}{2}\right)^2. \tag{5}$$

For the lower bound on the product $\lambda(G) \cdot \lambda(\overline{G})$, we look for the smallest value $\lambda(G)$ can have. Unlike in traditional vertex coloring, 0 is a valid color in radiocoloring, and the null graph on n nodes (consisting of n isolated nodes with no edges) can be radiocolored with $\lambda = 0$.

It is obvious that the lowest value the product, $\lambda(G) \cdot \lambda(\overline{G})$, can attain is 0, which is attained only when either G or $\overline{G}$ is a null graph (the other being the complete graph). Hence

$$\lambda(G) \cdot \lambda(\overline{G}) \quad \geq \quad 0. \tag{6}$$

Combining inequalities (5) and (6) we get

$$0 \quad \leq \quad \lambda(G) \cdot \lambda(\overline{G}) \quad \leq \quad \left(\frac{3n-3}{2}\right)^2.$$

3 A PARALLEL ALGORITHM

In practice a number of different approaches to solving the FAP have been developed , all being NP-hard[5, 8, 5, 12, 15, 16, 10]. The graph-theoretic approach via radiocoloring has also been shown to be NP-complete [13]. A few approximate algorithms have been proposed for the RCP, and some exact algorithms have been proposed for special classes of graphs namely, outerplanar graphs, graphs with treewidth k, permutation and split graphs by Bolaender, Kloks, Tan, and Leeuwen [1], planar graphs by Fotakis, Nikoletseas, Papadopoulou, and Spirakis [11]. Little effort has been made in speeding up the approximate algorithms for radiocoloring using parallel computers.

We propose an approximate *parallel algorithm for radiocoloring* (PARC) a general graph, which is based on the parallel graph coloring algorithm by Das and Deo[7]. It makes use of a *concurrent-read and exclusive-write* (CREW) *parallel random access machine* (PRAM) consisting of n processors, each of which is equipped with a small amount of local memory and a processing unit. Processors are identified by a unique number and they share a global memory. They can simultaneously read from the global shared memory. However, they cannot simultaneously write into the shared memory. The processors communicate among themselves using shared variables, and each can perform a scalar, arithmetic, or boolean operation in unit time.

We represent parallel operation by using the the *parallel forloop* construct. For example, the code segment

```
for all P_i such that 0 < i < q do
    parbegin
    statements to be executed in parallel
    parend
end for
```

when run on a processor P_k, indicates that when P_k executes the forloop it would fork into q parallel processes in q processors (corresponding to processors P_i, $0 < i < q$). The processors share a common environment and are distinguished by their unique processor *id* P_i. Any statement in between the **parbegin** and **parend** are executed simultaneously by all the q processors. On reaching the **parend** statement the q processes again join into a single process which could again run on P_k.

3.1 ALGORITHM DESCRIPTION

The *largest-degree-first* (LF) algorithm for graph coloring, proposed by Walsh and Powell [19] can be extended to radiocolor general graphs. The adjacency matrix of the given graph G and the number of vertices, n, are provided as the input to the algorithm. The algorithm consists of three steps: in the first step, the given vertices are sorted in a non-increasing order on the basis of their degrees; in the second step, a distance-two binary matrix of graph G is constructed; and finally, the vertices are assigned colors in the sorted order one by one. The algorithm makes use of an $n \times n$ adjacency matrix, ADJMATRIX, which is provided as input, a linear array, DEGREE, where the degree of each vertex is computed and stored, a linaer array, SORT, to store the vertices sorted by their degree, a $n \times n$ distance-two matrix, D2MATRIX, to show the list of vertices that are at a distance of two from each vertex, a $n \times 2n - 1$ (It is obvious that any graph can be radiocolored with a maximum λ of $2n - 2$ which is the radiochromatic number of a complete graph on n vertices [4].) matrix, FORBIDDEN, to record the colors that cannot be assigned to a vertex, and a linear array, COLOR, to store the colors assigned to each vertex.

First, the degree of each vertex is computed, degree of a vertex v, DEGREE$[v] = \Sigma_{0<i<n}$ ADJMATRIX$[v, i]$. Now the vertices are sorted based by their degrees in a non-increasing order and stored in the array SORT. Next, the distance-two matrix is computed, D2MATRIX$[x, y] \leftarrow 1$, if and only if ADJMATRIX$[x, k] = 1$ and ADJMATRIX$[k, y] = 1$ for some k, where $0 < k < n$, and $x \neq y$. Then, a color c is assigned to vertex v, if and only if c is the minimum number such that FORBIDDEN$[v, c] = 0$. Now the matrix FORBIDDEN is updated, for any vertex u, if ADJMATRIX$[v, u] = 1$ then FORBIDDEN$[u, c-1]$, FORBIDDEN$[u, c]$, and FORBIDDEN$[u, c+1]$ are assigned 1 (*i.e.* any vertex u which is adjacent to v cannot receive colors that are not at least two apart from c). Similarly, for any vertex u, if D2MATRIX$[v, u] = 1$ then FORBIDDEN$[u, c] = 1$ (*i.e.* color c cannot be assigned to any vertex u which is at a distance of two from v). In each successive iteration the next uncolored vertex with the largest degree is colored and the matrix FORBIDDEN updated to incorporate the change.

Like the sequential algorithm, the parallel version of the algorithm has three steps and makes use of all the data structures described above.

Input : ADJMATRIX, n
Output : λ, SORT

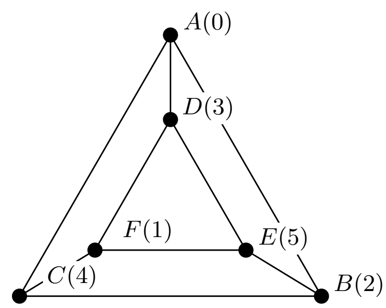
```
procedure PARC
for all P_i such that 0 ≤ i < n do {/* Step 1: Sort the vertices in
descending order based on their degree */}
  parbegin
  DEGREE[i] ← 0
  COLOR[i] ← 0
  for all l such that 0 ≤ l < n do
    D2MATRIX[i, l] ← 0
    FORBIDDEN[i, 2l] ← 0
    if l ≠ n − 1 then
      FORBIDDEN[i, 2l + 1] ← 0
    end if
    if ADJMATRIX[i, l] = 1 then
      DEGREE[i] ← DEGREE[i] + 1
    end if
  end for
  Sort the vertices in parallel to obtain the array SORT
  parend
end for
for all j such that 0 ≤ j < n do {/* Step 2: Compute the
distance two matrix */}
  for all P_i such that 0 ≤ i < n do
    parbegin
    if ADJMATRIX[i, j] = 1 then
      for all m such that 0 ≤ m < n do
        if ADJMATRIX[j, m] = 1 AND m ≠ i then {/*
        concurrent read required */}
          D2MATRIX[i, m] ← 1
        end if
      end for
    end if
    parend
  end for
end for
λ ← 0 {/* assume λ to be 0 initially */}
```

```
for all j such that 0 ≤ j < n do {/* Step 3: Assign colors to vertices */}
  for all P_i such that 0 ≤ i < n do
    parbegin
    find the smallest c such that FORBIDDEN[SORT[j], c] = 0
    parend
  end for
  COLOR[SORT[j]] ← c
  if λ < c then
    λ ← c
  end if
  for all P_i such that 0 ≤ i < n do
    parbegin
    if ADJMATRIX[SORT[j],i] = 1 AND c ≠ 0 then
      FORBIDDEN[i, c − 1] ← 1
    end if
    if ADJMATRIX[SORT[j],i] = 1 then
      FORBIDDEN[i, c] ← 1
    end if
    if ADJMATRIX[SORT[j],i] = 1 AND c ≠ 2n − 2 then
      FORBIDDEN[i, c + 1] ← 1
    end if
    if D2MATRIX[SORT[j],i] = 1 then
      FORBIDDEN[i, c] ← 1
    end if
    parend
  end for
end for
```

3.2 COMPLEXITY OF THE ALGORITHM

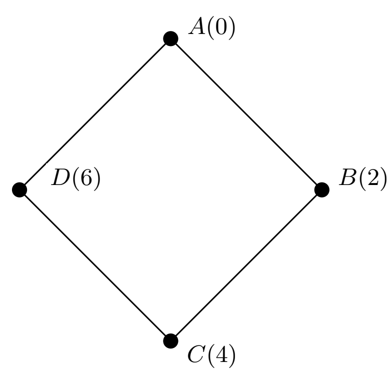
In this section, we first analyze the complexity of the sequential algorithm. The sequential algorithm consists of three steps, at first we compute the degree of each vertex, which takes $O(n^2)$ time, followed by sorting which takes $O(n \log n)$ time. In the next step we compute the distance-two matrix which takes time $O(n^3)$ and the final step involves assigning colors to the vertices; this step could be divided into two, first we search for a color to be assigned to a vertex and then update the FORBIDDEN matrix. This requires $O(n^3)$ time. Thus the overall complexity for the sequential algorithm is

$$\approx O(n^2) + O(n \log n) + O(n^3) + O(n^3),$$
$$\approx O(n^3).$$



$\lambda = 5$

Example 3: PARC employs $\lambda = 5$ to radiocolor the 3-prism shown above which is the best possible value.



$\lambda = 6$

Example 4: The cycle shown above can be radiocolored with $\lambda = 4$ but PARC uses $\lambda = 6$ which is the radiochromatic number of a complete graph on 4-vertices.

The parallel algorithm also consists of three steps. Given n processors we can compute the array DEGREE in parallel in time $O(n)$. Sorting the array DEGREE can be done in $O(\log n)$ time, if we use parallel quicksort. Then we compute the D2MATRIX which takes $O(n^2)$ time using n processors. Assigning colors to the vertices and updating the FORBIDDEN matrix requires time $O(n \log n)$ with n processors. The overall complexity for the parallel algorithm would be

$$\approx O(n) + O(\log n) + O(n^2) + O(n \log n),$$
$$\approx O(n^2).$$

The *cost* of a parallel algorithm is the product of the number of processors and time. A parallel algorithm is said to be *cost-optimal*, if the parallel cost is the same as the sequential time. The cost of the PARC is $O(n) \times O(n^2) \approx O(n^3)$, which is the same as the sequential time. Hence, our algorithm is cost-optimal. By employing Brent's scheduling [2] to PARC, one can use fewer processors and still obtain the same asymptotic time complexity.

4 CONCLUSION AND FUTURE WORK

In this paper, we study the radiocoloring problem on graphs and derive Nordhaus-Gaddam type result for the sum and product of the radiochromatic number of a graph and its complement. We also propose a cost-optimal, parallel largest-degree-first algorithm to radiocolor graphs for a shared memory, concurrent-read and exclusive-write (CREW) architecture. Further work on the PARC could involve implementation of the algorithm, application of Brent's scheduling, use of more efficient data structures, like the adjacency matrix of order $n \times \Delta^*$ as described by Eckstein and Alton in [9].

REFERENCES

[1] H. Bodlaender, T. Kloks, R. Tan, and J. van Leeuwen. Approximation for λ-coloring on graphs. Technical Report UU-CS-2000-25, Utrecht University, 2000.

[2] R. P. Brent. The parallel evaluation of general arithmetic expressions. *Journal of the Association for Computing Machinery*, 21:201–206, 1974.

$^*\Delta$ is the maximum degree of the graph

[3] T. Calamoneri and R. Petreschi. L(2, 1)-coloring matrogenic graphs. *Proceedings of the 5th Latin American Symposium on Theoretical Informatics*, 2002.

[4] T. Calamoneri and R. Petreschi. On the radiocoloring problem. *Proceedings of the 4th International Workshop on Distributed Computing*, 2002.

[5] D. Castelino, S. Hurley, and N. M. Stephens. A tabu search algorithm for frequency assignment. *Annals of Operations Research*, 63:301–319, 1996.

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

[7] S. K. Das and N. Deo. Parallel coloring of graphs: two approximate algorithms. *International Journal of Computer Mathematics*, 27:147–158, 1989.

[8] M. Duque-Anton, D. Kunz, and B. Rüber. Channel assignment for cellular radio using simulated annealing. *IEEE Transactions on Vehicular Technology*, 42:14–21, 1993.

[9] D. M. Eckstein and D. A. Alton. Parallel graph processing using depth-first search. *Proceedings of the Conference on Theoretical Computer Science*, 1977.

[10] P. C. Fishburn and F. S. Roberts. Full color theorems for L(2, 1)-colorings. Technical Report DIMACS TR:2000-08, Center for Discrete Mathematics and Theoretical Computer Science, 2000.

[11] D. A. Fotakis, S. E. Nikoletseas, V. G. Papadopoulou, and P. G. Spirakis. NP-completeness results and efficient approximations for radiocoloring in planar graphs. *Proceedings of the 25th International Symposium on Mathematical Foundations of Computer Science*, 2000.

[12] N. Funabiki and Y. Takefuji. A neural network parallel algorithm for channel assignment problem in cellular radio networks. *IEEE Transactions on Vehicular Technology*, 41:430–437, 1992.

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

[14] T. K. Jonas. *Graph Coloring Analogues With a Condition at Distance Two: L(2, 1)- Labelings and List λ-Labelings.* PhD thesis, Department of Mathematics, University of South Carolina, Columbia, SC, 1993.

[15] D. Kunz. Channel assignment for cellular radio using neural networks. *IEEE Transactions on Vehicular Technology*, 40:188–193, 1991.

[16] W. K. Lai and G. G. Coghill. Channel assignment through evolutionary optimization. *IEEE Transactions on Vehicular Technology*, 45:91–96, 1996.

[17] E. A. Nordhaus and J. W. Gaddum. On complemantary graphs. *American Mathematical Monthly*, 63:175–177, 1956.

[18] D. Sakai. Labeling chordal graphs: Distance two condition. *SIAM Journal on Discrete Mathematics*, 7:133–140, 1994.

[19] D. J. A. Walsh and M. B. Powell. An upper bound for the chromatic number and its application to timetabling problem. *Computer Journal*, 10:85–86, 1967.

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

[21] R. Y. Yeh. *Labeling Graphs with a Condition at Distance Two.* PhD thesis, Deptartment of Mathematics, University of South Carolina, Columbia, SC, 1990.