Proceedings of the World Congress on Engineering 2010 Vol I
WCE 2010, June 30 - July 2, 2010, London, U.K.

Optimized Channel Assignment in Distributed Environment

Sunita Choudhary, G. N. Purohit

***Abstract*— Channel assignment problem in wireless network is closely related to vertex coloring problem. The problem is to assign channels to hosts in such a way that interference among hosts is eliminated and the total number of channels is minimum, which is equivalent to vertex coloring of a graph with minimum number of colors. A proper vertex coloring is a fundamental problem in distributed systems. In this paper, we are describing channel assignment problem as an optimization graph coloring problem with the objective function to minimize the total number of colors used and present an experimental analysis of simple and elegant distributed algorithm for optimized vertex coloring problem. We have proved our results with the help of simulations.**

Index Terms— Channel Assignment, Distributed Algorithm, Graphs, Experimentation

I. INTRODUCTION

The integration of mobile and wireless communication and computing in fixed network introduces new issues and problem in distributed computing. One of the basic problems in wireless networks is that the spectrum is a scarce resource. The tremendous growth of the mobile users population coupled with the bandwidth requirements of the new cellular services is in contrast to the limited spectrum resources that have been allocated for mobile communication. The radio spectrum for a given services is generally divided into set of disjoint radio channels.

A radio network consists of group of transmitters sharing a common radio channel (frequency) and communicating with each other using this channel. If two nearby transmitters are operating on the same frequency then they have the potential to interfere with each other. In the simplest model, the frequencies assigned to such pairs of transmitters are required to de distinct and the object is to minimize the total number of frequencies used.

Channel assignment can be defined as the problem of assigning a set of compatible channels to each cell in a spectrum in an efficient manner. So an efficient channel assignment is very important in cellular mobile network planning because it gives an efficient use of the available spectrum. However, the channel assignment problem has been shown to belong to the class of NP-complete combinatorial optimization problem.

Sunita Choudhary is with the Department of Computer Science, Banasthali University, Rajasthan, India. (phone: 91- 9413330333; fax: 0141-2375322 ; e-mail: sunitaburdak@yahoo.co.in).

G.N. Purohit is with Department of Mathematical, Banasthali University, Rajasthan, India (e-mail: gn_purohitjaipur@yahoo.co.in).

Graph coloring is a natural model for channel assignment problem. The vertices of a graph represent the transmitters, and the edge is added between any pair of vertices (transmitters) which have the potential to interfere. Then the channels are assigned frequencies corresponding to the colors assigned to vertices, of course adjacent vertices receive distinct colors.

Vertex coloring is a well attended problem of graph theory. It is basically a mapping $f : C \rightarrow V$, where V is a vertex set and C = {1,2,………t} is a set of positive integers or colors, such that no two adjacent vertices maps to the same number or color. The problem of graph coloring is known to be NP-complete problem Such NP complete problems can easily be solved with the help of distributed computing and parallelism. Both approaches break the problem into individuals and fetch the result in acceptable time domain. Vice-versa, use of vertex coloring is also beneficial in some of the distributed computing problems. For example, vertex coloring is used as a means to break symmetries that is one of the main themes in distributed computing. The Vertex coloring is so important that many distributed algorithm use such a coloring as a sub-routine in higher-order communication tasks, example include:

- Channel assignment in wireless networks,
- MIS construction,
- Exams scheduling in a university,
- Register allocation in a compile, etc.

Specially for networking aspect, the graph coloring problem is important because many network coordination primitives are based on colorings of the nodes of the network. For example, the assignment of frequencies or time slots in wireless networks is a classical application of minimum graph coloring [6], [18]. In contrast to many other distributed graph-theoretic problems [3], [7], [8], not much progress has been made on distributed coloring in the last few years.

Distributed network can be modeled as a graph G = (V, E) with nodes representing the processors and the edges representing the communication link. In this paper, we are describing channel assignment problem as an optimization graph coloring problem with the objective function to minimize the total number of colors used so that no two adjacent nodes (nodes with a direct link connecting them) can have the same color However, the typical goal is to obtain a coloring with as few colors as possible.

In particular, defining the chromatic number of a graph G, denoted by $\chi(G)$, as the minimum number of colors that can legally color the vertices of G. Whereas it is easy to see that a graph with maximum degree Δ can be colored using at most Δ+1 colors, computing the chromatic number of a graph is

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

NP-hard. Thus, efficient algorithm that uses close to $\chi(G)$ colors is of our interest.

The rest of the paper is organized as follows. In Section II we describe different models of distributed algorithm design and reason of carrying simulation. In Section III, we summarize some important results related to distributed coloring, whereas Section IV and V present the experimentation and analyze our algorithm for different graphs. The paper ends with conclusion in Section VI.

II. ALGORITHM MODEL

Two designing models of distributed algorithms are synchronous model and asynchronous model. In a synchronous algorithm, nodes operate in synchronous rounds. Synchronous model is based on the following:

1. The a priori knowledge of every vertex consists of its own index and of parameters n and Δ.
2. Computations proceed in synchronous rounds controlled by a global clock.
3. In every round every vertex can perform the following actions:
   (a) receives messages from all of its neighbors,
   (b) executes an arbitrary amount of local computations,
   (c) sends messages to all of its neighbors,
4. All actions of vertices are deterministic.

In the asynchronous model, algorithms are event driven. Processors cannot access a global clock. A message sent from one processor to another will arrive in finite but unbounded time. Practical implementation of this model is quite difficult.

The analysis simulates the algorithm in the *synchronous, message-passing distributed network*, a theoretical model that approximates the behavior of real distributed architecture where typically the cost of routing messages is order of magnitude greater than that of performing local computation.

We carried out our experimental analysis by simulating this theoretical model for the following reasons:

- This simulation is relatively inexpensive in terms of resources and programming effort, an implementation on a real parallel machine would be much more demanding.
- A truly distributed implementation would make experimental results more informative but unsuitable for drawing general conclusions about the behavior of the algorithm.
- One obvious problem with experimental studies of algorithms in general is that they are based on a specific implementation on a specific machine. This makes it very hard to compare different experimental results. It would be quite useful to simulate in order to clearly compare with different experimental results.

III. PREVIOUS RESULTS

There are just a few experimental studies of parallel or distributed algorithms, in spite of a quite extensive literature is available on vertex coloring

Randomized and deterministic distributed algorithms have been studied in [1], [5], [17], [18]. For vertex coloring, there are randomized and deterministic algorithms that use Δ+1 or Δ colors. For the randomized case, the fastest algorithm known so far is polylogarithmic[13], [17], and for the deterministic case, the fastest one has time complexity $O(n^{O(1/\sqrt{\log n})})$[18].

Finocchi, Panconesi, and Silvestri [4] have studied the empirical efficiency of several existing distributed vertex coloring algorithms. One of the outcomes of their study is that every node can indeed try in every round unlike the original paper of Luby[12] where nodes tried to get a color in a round with a certain probability w, called the wake-up probability.

Marathe, Panconesi, and Risnger Jr. [15] have performed an empirical study of a distributed edge coloring algorithm. They studied several classes of graphs viz. trees, bipartite graphs, cliques, hypercubes, and random graphs. One of the aims of their study was to see how using a few extra colors affects the performance and report that if 5%-15% extra colors are used then the performance is significantly better.

Kuhn and Wattenhofer[10] proved new strong lower bounds for two special kinds of coloring algorithms. They show that the number of colors of the computed coloring has to be at least $\Omega(\Delta^2 / \log^2\Delta + \log\log m)$. And for iterative algorithm lower bound is $\Omega(\Delta / \log^2\Delta + \log\log m)$. Leith and Clifford [11], [14] have recently proposed a fully distributed stiachastic coloring algorithm inspired by frequency allocation within the constraints of wireless networks employing the IEEE 802.11 standard. Duffy et al. [2] analyzed the complexity of a Communication Free Learning (CFL) algorithm that is distributed algorithm and proposed for channel assignment in wireless computer network by Leith and Clifford [11] .

Schneider and Wattenhofer[19] presented deterministic algorithm which finishes in O(log* n) time, n being the number of nodes, for growth bounded graphs. They also compute a Δ +1 coloring in O(log* n) time, where Δ is the maximum degree of the graph. Pindiproli and Kothapalli [16] compared the empirical performance of redistributed vertex coloring algorithm proposed by [9] with that of Luby's algorithm [12].

Optimized graph coloring is yet not much explored. In this paper, we are trying to find an optimized graph coloring algorithm and experimenting the algorithm that uses very few colors than Δ, where Δ is the maximum degree of any graph.

IV. SUMMARY OF RESULTS

- The algorithm is independent of the number of nodes. With graphs having same degree and different n, algorithm produces almost similar results.
- Color complexity and time complexity mainly depends on maximum degree Δ of the graphs.
- The algorithm produces the optimized graph coloring. For graphs having hundreds to thousands vertices with maximum degree Δ with $10 \le \Delta \le 100$, it requires only 5-30 colors, that is much less than Δ. We are saving almost 50-70% of colors, because our palette size is Δ+1.
- In terms of time complexity, the algorithm requires 10-30 rounds only for graphs having hundreds and thousands of nodes and Δ varying from 10 to 100.
- The smaller the number of high degree vertices, the higher is the probability of saving colors.
- Time complexity is $O(\Delta/\log\Delta)$ and the coloring complexity is O(Δ), but as degree increases number

ISBN: 978-988-17012-9-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

of colors used by our algorithm is much less than Δ.

V. EXPERIMENTAL RESULTS

The implementation of the algorithms is done in JAVA on a standard x86 architecture machine. Our experiments were carried on a Intel® Pentium® 4 CPU 3.00 GHz processor, with 512 Mbytes of memory, running a standard Microsoft Windows XP Professional Version 2002 Service Pack 2. The Java version used was 1.5.0_06.

In this Section we analyze the obtained experimental results of distributed algorithm for computing Δ coloring. We consider the cases (i) two dimensional regular meshes and (ii) random graphs.

(i) Two dimensional, regular meshes

We have considered two dimensional, regular mesh with Δ=6 for our experiment and number of nodes varying from 12 to 500. Since each vertex has exactly 6 neighbors except boundary vertices, this structure is very similar to 7 cell i.e. hexagonal cell structure of a cellular network which is the base of all cellular networks. We carried out 20-30 simulation for different graph sizes and present the average case behavior in terms of colors and rounds; Fig 1 shows the graphical representation of the percentage of nodes colored in different rounds. Fig 2 shows average number of nodes colored in different rounds.

(*ii*) *Results for Random Graphs*

To verify our results for higher degree graphs, we have generated random graphs G(n,Δ) for particular number of nodes and for particular maximum degree Δ. In Random Graphs number of nodes varying from 200 to 7000 and maximum degree varying from 10-100. Fig 3 represents the number of colors used for proper vertex coloring in graphs, whereas in Fig 4, we present communication rounds required for proper vertex coloring of random graphs. Average numbers of colors used by graphs of different degrees are represented in Fig 5.

VI. CONCLUSION

Channel assignment problem is one of the important issues in mobile and wireless communication. We have presented a distributed algorithm for channel assignment, which provides optimal channel assignment. The algorithm uses fewer than Δ colors as Δ increases. It gives optimized vertex coloring. Radio spectrum has high cost and communication cost is very low and we have to save our spectrum for its best utilization, our algorithm provides the best solution.

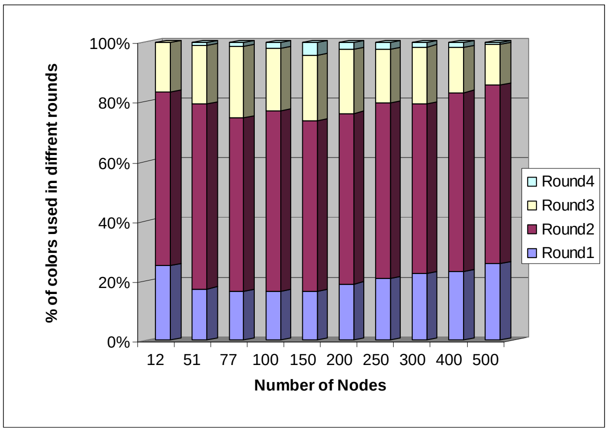

Fig 1 Percentage of nodes colored in different rounds

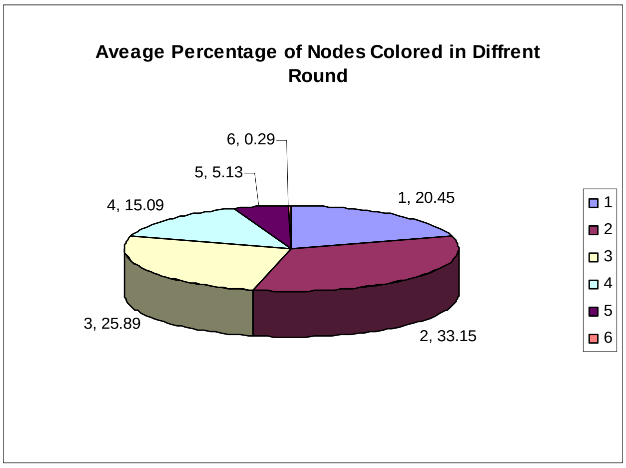

Fig 2 Average Percentage of Nodes Colored in Different Rounds

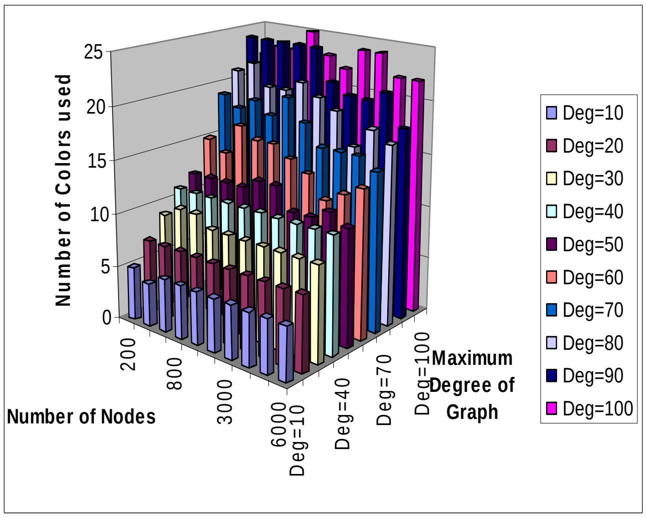

Fig.3 Number of Colors Used in Random Graphs

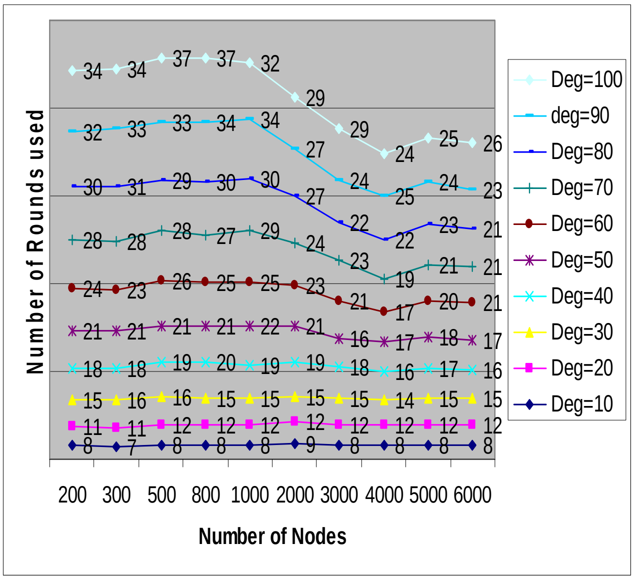

Fig 4 Communication Rounds in Random Graphs

REFERENCES

[1] Awerbuch B., Goldberg A., Luby M. and Plotkin S..1989. "Network decomposition and locality in distributed computation,", in Proc. 30th IEEE Symp. on Foundations of Computer Science,. 364-369. Tavel, P. 2007 Modeling and Simulation Design. AK Peters Ltd.

[2] Duffy K.R. , O'Connell N. , Sapozhnikov A..2008. "Complexity analysis of a decentralised graph colouring algorithm", Information Processing Letters, v.107 n.2, p.60-63

[3] Elkin M..2004.." Distributed approximation: A survey". ACM SIGACT News, 35(4):40–57

[4] Finocchi, I., Panconesi, A., and Silvestri, R.2002.: "Experimental analysis of simple, distributed vertex colouring algorithms". In Proc. of the ACM Symposium on Discrete Algorithms (SODA).

[5] Grable D.A. and Panconesi A.1997.." Nearly optimal distributed edge colouring inO(log log n) rounds,", in Random Structures & Algorithms, 385-405.

[6] Hermann T. and Tixeuil S. 2004.." A distributed TDMA slot assignment algorithm for wireless sensor networks". In Proc. of 1st Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS), volume 3121 of Lecture Notes in Computer Science, pages 45–58

[7] Jianping Jiang, Ten-Hwang Lai, Neelam Soundarajan 2002. "On Distributed Dynamic Channel Allocation in Mobile Cellular Networks", IEEE Transactions on Parallel and Distributed Systems ,Volume 13 , Issue 10 ,1024 – 1037

[8] Kuhn F., Moscibroda T., and Wattenhofer R..2006." The price of being near-sighted". In Proc. of the 17th ACM-SIAM Symposium on Discrete Algorithms (SODA).

[9] Kothapalli, K., Onus, M., Scheideler, C., and Schindelhauer .2006. " Distributed coloring in ˜O(√logn)–bits". In Proc. of the International Parallel and Distirbuted Processing Symposium (IPDPS).

[10] Kuhn F. and Wattenhofer R.2006." On the complexity of distributed graph coloring", 25th Ann ACM Symp Princ Distrib Comput. (PODC), ACM Press, 7–15

[11] Leith, D. J, and Clifford P.2006." Convergence of distributed learing algorithms for optimal wireless channel allocation", In IEE CDC. Pages 2980-2985

[12] Luby M..1985. " A simple parallel algorithm for the maximal independent set problem". Proc. of ACM Symposium on Theory of Computing , 1–10.

[13] Luby M.1988. " Removing randomness in parallel computation without a processor penalty", in Proc. 29 th IEEE Symposium on Foundations of Computer Science, 162-173

[14] Malone, D., Clifford P.,Reid D. and Leith D.2007. "Experimental implementation of optimal WLAN channel selection without communication". In IEEE DySPAN, 316-319

[15] Madhav V. Marathe, Alessandro Panconesi, Larry D. Risinger Jr..2004.. " An experimental study of a simple, distributed edge-coloring algorithm". ACM Journal of Experimental Algorithms 9.

[16] Pindiproli S.K. and Kothapalli K..2009." Experimental Analysis of Distributed Coloring Algorithms", Submitted for publication, Jan 2009.

[17] Panconesi A. and Srinivasan A..1992. " Improved distributed algorithms for coloring and network decomposition problems" , in Proc.ACM Symposium on Theory of Computing, ACM, NewYork, . 581-592.

[18] Ramanathan S..1999. "A unified framework and algorithm for channel assignment in wireless networks". Wireless Networks, 5:81–94

[19] Schneider J., Wattenhofer R..2008." log-star distributed maximal independent set algorithm for growth-bounded graphs**,** Annual ACM Symposium on Principles of Distributed Computing ", Proceedings of the twenty-seventh ACM symposium on Principles of distributed computing , 35-44