COVERING AND PACKING WITH INTERVALS

ANAMITRA ROY CHOUDHURY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY DELHI

AUGUST 2016

COVERING AND PACKING WITH INTERVALS

by

ANAMITRA ROY CHOUDHURY

Department of Computer Science and Engineering

Submitted

in Fulfillment of the Requirements of the Degree of Doctor of Philosophy

to the

Indian Institute of Technology Delhi

AUGUST 2016

Certificate

This is to certify that the thesis titled **"Covering and Packing with Intervals"** being submitted by **Mr. Anamitra Roy Choudhury** to the **Indian Institute of Technology Delhi**, for the award of the degree of **Doctor of Philosophy**, is a record of bonafide research carried out by him under our supervision. The results obtained in this thesis have not been submitted to any other university or institute for the award of any other degree or diploma.

Naveen Garg
Professor
Department of Computer Science and Engineering
Indian Institute of Technology Delhi
Delhi 110016

Amit Kumar
Professor
Department of Computer Science and Engineering
Indian Institute of Technology Delhi
Delhi 110016

Acknowledgements

I take this opportunity to express my sincere gratitude to my supervisors, Prof. Naveen Garg and Prof. Amit Kumar for their invaluable guidance throughout this work. I had always looked up to Prof. Naveen Garg, even before I started my doctoral thesis, from his lecture videos on algorithms and his eminence as a researcher. It was amazing to see how he could extract out at ease the underlying simple combinatorial argument from a mathematically involved analysis. During my course of Ph.D., I had the maximum interaction with Prof. Amit Kumar; he taught me how to conduct research, to work out all mathematical details and to communicate research ideas in writing papers. It is hard for me to imagine doing my Ph.D. without Prof. Amit Kumar's help. I sincerely hope I have gained some fraction of the prowess from my advisors over these few years.

My sincere thanks to Prof. Sandeep Sen, Dr. Ragesh Jaiswal, Dr. Amitabha Bagchi, Dr. Yogish Sabharwal and Dr. Sambuddha Roy for agreeing to be in my doctoral research committee and providing me with valuable feedback in the course of my Ph.D. I gratefully acknowledge Dr. Kavitha Telikepalli and Dr. Tobias Mömke, the members of my Ph.D. committee, for their time and valuable feedback on a preliminary version of this thesis. I also want to thank my MTech. thesis supervisor, Prof. Somenath Biswas, IIT Kanpur, for enlightening me the first glance of research.

I thank the co-authors the work with whom has been included in this thesis. The names include Archita Agarwal, Sivaramakrishnan Natarajan, Dr. Sambuddha Roy, Dr. Venkatesan Chakravarthy and Dr. Yogish Sabharwal. In particular, I learnt a lot from the technical discussions I had with Venkat and Sambuddha, two great minds who are also respectively my colleague and ex-colleague at IBM Research, India. Yogish has been been my mentor and later my manager at IBM Research India, and I take this opportunity to thank him for standing as a pillar of support, guidance and motivation over the years.

I extend my gratitude to one of my oldest friends since school days Syamantak Das. He was instrumental in developing some of the work that has been included into this thesis. He was the person with whom I have shared moments of deep anxiety as well as of big excitement during these years; his presence was very important in an activity that is often felt as tremendously solitaire.

I met some of the most interesting and warm individuals in these six years of Ph.D. Thanks to my Ph.D. mates Shibashis, Anup, Suvam, Nisha, Chinmay, Jatin, Manoj, Neha and Sayak for sharing some of the nice moments that I would cherish for long. It was really fun having those endless discussions on research and life over coffee, or dinner. I extend my thanks to my colleagues and friends from IBM Research - Vaibhav, Rashmi, Rakesh, Thomas, Gurbinder, Swati, Suman, Chandra, Sudhir, Ritwik, Pramod are only few of the names that I can put here. It is a pleasure to thank my personal friends Arkaprava, Anjana, Samik, Avik, Souvik, Ashique and Sayan for their support over the years. Some special words of gratitude go to my table-tennis buddies at office who gave me the necessary distractions from my research and made these years memorable.

There are few persons in my life whose contribution to this thesis is more than I can express here. Thanks to my wife Sukanya for being a great companion and showing how special I am for her; I look forward to our lifelong journey. Last but not the least, I thank my parents and elder brother for their enormous support, love and encouragement towards my education, without which this work would not have been possible. I dedicate this thesis to them.

Anamitra Roy Choudhury

Abstract

In this thesis, we consider different variants of the classical covering and packing problems. The most general covering problem is the set cover problem. There are two well-known classes of approximation algorithms for the set cover problem [61], one with approximation ratio of $O(\log \Delta)$ *and another have an approximation ratio of* f*. Here* Δ *is the maximum cardinality of the sets, and* f *is the maximum number of sets that any element belongs to. However, for the problems of line cover, tree cover and several other special cases of the set cover problem, we get constant factor approximation algorithms even though* Δ *and* f *are arbitrary for the problems. The first problem we study in this thesis is to identify the property that this special case of set cover problems satisfies which makes these problems easier to approximately solve. We study a class of set cover problems that satisfy a special property which we call the small neighborhood cover property. This class encompasses several well-studied problems including vertex cover, interval cover, bag interval cover and tree cover. We design unified distributed and parallel algorithms that can handle any set cover problem falling under the above framework and yield constant factor approximations. These algorithms run in polylogarithmic communication rounds in the distributed setting and are in NC, in the parallel setting. The above results were obtained*

jointly with Archita Agarwal, Venkatesan T. Chakaravarthy, Sambuddha Roy and Yogish Sabharwal [1].

Next we study the special case of Knapsack Cover problem (this is the case of interval cover where the line consists of a single edge) under matroid constraints. Our result proves a PTAS for this problem. The problem described above falls in the realm of mixed packing covering problems. We also consider packing extensions of certain other covering problems and prove that in such cases it is not possible to derive any constant factor approximations. These results were obtained jointly with Venkatesan T. Chakaravarthy, Sivaramakrishnan R. Natarajan and Sambuddha Roy [23].

We next study some packing problems. In particular, we consider the problem of maximizing the throughput of jobs wherein each job consists of multiple tasks. Consider a system offering a uniform capacity of a resource (say unit bandwidth). We are given a set of jobs, each consisting of a sequence of at most r tasks. Each task is associated with a window (specified by a release time and a deadline) within which it can be scheduled; each task also has a processing time and a bandwidth requirement. Each job has a profit associated with it. A feasible solution must choose a subset of jobs and schedule all the tasks for these jobs such that at any point of time, the total bandwidth requirement does not exceed the capacity of the resource; furthermore, the schedule must obey the precedence constraints (tasks of a job must be scheduled in order of the input sequence). The goal is to compute the feasible solution having maximum profit. Our main result presents an $O(r)$-approximation algorithm for the general case wherein tasks can have windows and bandwidths of tasks within the same job may be non-uniform. The above results were obtained jointly with Venkatesan T. Chakaravarthy, Sambuddha Roy and Yogish Sabharwal [24].

We finally consider the problem of online coloring of intervals on a tree. Here the input consists of an undirected tree defined over a set of vertices. Each demand or path arrives online and is specified as a pair of vertices. The objective is to color the demands so that no two demands sharing an edge are given the same color. We also consider a more general case where all the edges of the tree offer uniform capacity of B, say; for this case, B paths of the same color may be routed through an edge. Bartal and Leonardi [9] show that any deterministic algorithm for the unit capacity case can not have competitive ratio better than $\Omega(\frac{\log n}{\log \log n})$. In this thesis, we present a constant competitive algorithm for the online interval coloring problem on a tree under recourse *model; also our algorithm uses constant amortized number of reassignments/ recoloring. This means that if the total number of demands requested at a time is $\mathcal{P}$, our algorithm employs $O(\mathcal{P})$ number of reassignments of colors. Our results also extend to the case where every edge of the tree offers a uniform capacity value.*

An independent problem of interest is the online coloring problem of an inductive graph. In the online coloring problem of a d-inductive graph, vertices arrive over time (input sequence not known in advance): when a vertex arrives, only edges to existing vertices are revealed. The objective is to color the vertices of the graph such that no two adjacent vertices have the same color. Irani [42] shows that there exist d-inductive graphs such that any online algorithm needs $O(d \log n)$ colors while the offline algorithm requires only $d + 1$ colors, where n is the number of vertices in the graph. Here again, we show that we can color the d-inductive graph G using $3d$ colors with the number of color reassignments bounded to $O(n)$ after arrival of n vertices.

Contents

Acknowledgements **vii**

Abstract **xi**

List of Figures **xix**

1 Introduction **1**

1.1 Preliminaries . . . 1

1.2 Organization of the thesis . . . 19

2 Set Cover Problems with Small Neighborhood Covers **21**

2.1 Overview . . . 21

2.1.1 SNC Property . . . 24

2.1.2 Example Problems . . . 24

2.1.3 Layer Decomposition . . . 30

2.1.4 Our Results . . . 32

2.1.5 Prior Work on Example Problems and Comparison . . . 34

2.1.6 Proof Techniques 37
2.2 Preliminaries 39
2.3 Computing Layer Decompositions 41
2.4 Sequential Algorithm 44
2.5 Parallel Algorithm 50
2.5.1 Forward Phase 54
2.5.2 Reverse Delete Phase 57
2.6 Distributed Algorithm 65
2.7 Decomposition Length for the Tree Cover Problem 69

3 Knapsack Cover with Matroid Constraints **75**

3.1 Overview 75
3.1.1 Problem Definition 75
3.1.2 Proof Techniques and Our Contribution 78
3.2 Preliminaries 83
3.3 KCM with Cardinality Matroids 85
3.4 Partition Matroids 88
3.5 KCM with Arbitrary Matroids 89
3.6 Multiple Matroids 94

4 Scheduling Jobs with Multiple Non Uniform Tasks **97**

4.1 Overview 97
4.1.1 Problem Definition 97
4.1.2 Our Contribution 102

4.2 LP Formulation and Solving the LP 103

4.3 Rounding the LP Solution 110

4.4 Other Results 116

4.4.1 Variants of the Main Result 116

4.4.2 Integrality Gap 119

5 Online Coloring of Intervals on a Tree **123**

5.1 Overview 123

5.1.1 Problem Definition 124

5.1.2 Our Contribution 128

5.2 Coloring on Trees 129

5.2.1 Lower Bounds 130

5.2.2 Coloring Algorithm on a Tree 136

5.3 Coloring on d-Inductive Graphs 140

5.4 Interval Coloring on Trees 144

5.5 Interval Coloring Algorithm on Uniform Capacitated Trees 147

6 Conclusion and Open Problems **149**

List of Figures

2.1 Illustration for the interval cover problem 25

2.2 Illustration for the tree cover problem 27

2.3 Illustration for the bag interval cover problem 28

2.4 Illustration for the bag priority interval cover problem 29

2.5 Sequential τ-approximation algorithm 45

2.6 Parallel Algorithm 53

2.7 Distributed Algorithm 67

2.8 Chain decomposition 72

3.1 Main Algorithm 90

4.1 Illustrations 100

4.2 Illustration for Proof of Lemma 4.1 113

4.3 Illustration for integrality gap 120

5.1 Illustration: Interval Coloring on a Tree 125

5.2 Counter example 1: large number of colors needed in online algorithm . . 131

5.3 Counter example 2: Many reassignments needed for online 2 coloring . . . 133

5.4 Tree Coloring Algorithm: Reassignment of Colors 135

5.5 Routine for randomized assignment 138

5.6 Algorithm for coloring a vertex round wise 142