APPROXIMATION ALGORITHMS FOR SCHEDULING

by

SANKARAN ANAND

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
April 2013

Certificate

This is to certify that the thesis titled **"Approximation Algorithms for Scheduling"** being submitted by **Mr. Sankaran Anand** 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.

Dr. Naveen Garg
Department of Computer Science and Engineering
Indian Institute of Technology
Delhi 11001

Dr. Amit Kumar
Department of Computer Science and Engineering
Indian Institute of Technology
Delhi 11001

Date:
Place: New Delhi

Acknowledgements

I am indebted to my thesis advisors Naveen Garg and Amit Kumar for the initiation, continuation and completion of this thesis. I began work with Naveen starting at the undergraduate level and I got a taste of research in working with him on my B.Tech project. Afterwards, his support was operative in my pursuit of a PhD at IIT Delhi. Both Naveen and Amit generously gave me a lot of time, provided direction to my research, proved difficult theorems, helped me in writing up and presenting my work and above all were extremely patient with me during my stay here at IIT Delhi.

My two trips to Max Planck Institute (MPI) in Saarbrucken, Germany in the summer of my second and third year at IIT Delhi were invaluable to me in providing opportunities in meeting with other researchers in the field. I wish to thank IMPECS (Indo-German Max Planck Center for Computer Science) for sponsoring my trips there and Kurt Melhorn at MPI for hosting me. I also wish to thank Nicole Megow, Tobias Friedrich, Karl Bringmann and Chien-Chun-Huang at MPI for working with me on various problems.

I also wish to thank the inhabitants of the research scholars room for creating a friendly and pleasant atmosphere at IIT Delhi. This includes, but is not limited to, Manoj Gupta, Yamuna Prasad, Syamantak Das, Chinmay Narayan, Anuj Gupta, Arindam Pal, Anamitra Roy Choudhary, Nisha Jain, Shibashis Guha and Swati Sharma. Many of them also accompanied me on outings, research or otherwise, which will forever be a source of good memories.

Last, but not the least, I wish to thank my parents, especially my mother, for their immense support and for keeping me on track during my time here.

Sankaran Anand

Abstract

We consider the problem of online scheduling of jobs on multiple machines (jobs arrive over time and the online algorithm does not have knowledge of the future). We assume that the jobs can be freely pre-empted and resumed later. Within this setting, there are various objectives we can pursue. In this thesis, we look at two such objectives. Both of these objectives are hard to achieve in an online setting (in the worst case). Therefore, we weaken the adversary using *speed-augmentation*. In other words, the machines available to the online scheduler are a factor s faster than the (offline) adversary.

The problems considered are the following:

- *Minimizing flow time on related machines*

  In this setting, we allow machines to have different speeds. A job can be processed on any machine and it takes time inversely proportional to the speed of the machine. In this setting, we give an $O(\log P)$-competitive online algorithm for minimizing flow time. This result is tight.

- *Using dual-fitting to minimize flow time*:

  The flow time of a job is the total time spent by a job in the system, i.e. the completion time minus the release time. In the basic setting, we are interested in minimizing the (weighted) flow time of jobs in the system. Again, the objective is hard to achieve for an online algorithm, so we allow the machines of the on-line algorithm to have $(1 + \varepsilon)$-extra speed than the offline optimum (the adversary).

  Typically, such algorithms are analyzed using non-trivial potential functions which yield little insight into the proof technique. We propose that one can often analyze such algorithms by looking at the dual (or Lagrangian dual) of the linear (or convex) program for the corresponding scheduling problem, and finding a

feasible dual solution as the on-line algorithm proceeds. As representative cases, we get the following results :

1. For the problem of minimizing weighted flow-time, we show that a slight modification of the greedy algorithm of Chadha-Garg-Kumar-Muralidhara is $O\left(\frac{1}{\varepsilon}\right)$-competitive. This is an improvement by a factor of $\frac{1}{\varepsilon}$ on the competitive ratio of the older algorithm.
2. For the problem of minimizing weighted ℓ_k-norm of flow-time, we show that a greedy algorithm gives an $O\left(\frac{k}{\varepsilon^{2+\frac{1}{k}}}\right)$-competitive ratio. This matches the result of Im and Moseley.
3. For the problem of minimizing weighted flow-time plus energy, and when the energy function $f(s)$ is equal to $s^\gamma, \gamma \geq 1$, we show that a natural greedy algorithm is $O(\gamma^2)$-competitive. Prior to our work, such a result was known for the related machine setting only (Gupta-Krishnaswamy-Pruhs).

- We investigate the dependence on k of the competitive ratio of an online scheduler to minimize the l_k-norm of flow time to k. For this, we consider the extreme case $k \to \infty$ and prove the following:

  1. Any on-line algorithm for minimizing maximum weighted flow-time on subset-parallel machines must have competitive ratio of $\Omega(\log m)$ even with constant speed augmentation of machines.
  2. For the setting of related machines, we show that there is an $O(\epsilon^{-3})$-competitive algorithm to minimize the maximum weighted flow-time.
  3. In case of unweighted ℓ_∞ norm of flow-time, i.e., all jobs have weight 1: we give a $2/\varepsilon$-competitive algorithm for minimizing maximum unweighted flow-time on unrelated machines, where we give $(1+\varepsilon)$-speed augmentation to the machines. Without any speed augmentation, any on-line algorithm has competitive ratio of $\Omega(m)$, where m is the number of machines. Our algorithm above is not "immediate dispatch", i.e. it does not assign jobs to machines as soon as they arrive. To justify this drawback, we show an $\Omega(\log m)$ bound on the competitive ratio of any immediate-dispatch algorithm. In both of these lower bound constructions, only use jobs whose processing times are from $\{1, \infty\}$ and thus also apply to the subset-parallel setting.

Contents