DYNAMIC PROGRAMMING FOR SCHEDULING PROBLEMS

JATIN BATRA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY DELHI
FEBRUARY 2019

DYNAMIC PROGRAMMING FOR SCHEDULING PROBLEMS

by

JATIN BATRA

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

FEBRUARY 2019

Certificate

This is to certify that the thesis titled **"Dynamic programming for scheduling problems"** being submitted by **Jatin Batra** 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

Acknowledgments

The author is extremely grateful to his supervisors Prof. Naveen Garg and Prof. Amit Kumar for their invaluable and enduring inspiration, guidance and support. The author would like to profusely thank Andreas Wiese, Tobias Moemke, Deeparnab Chakrabarty, Debmalya Panigrahy, Rong We and Carola Winzen for their wonderful hospitality. The author is also grateful to Nikhil Kumar, Aounon Kumar, Syamantak Das, Anamitra Roy Choudhary and Anup Kumar Bhattacharya for being great colleagues. The author thanks TCS for their fellowship and travel support.

Jatin Batra

Abstract

In this thesis, we consider the classical problems of scheduling jobs on a single machine to minimize weighted flowtime and the unsplittable flow problem on a path. All the problems we study are in the offline setting. For these results, we also rely on discovering novel connections to covering problems and exploiting these through ideas in dynamic programming.

In the weighted flow-time problem on a single machine, we are given a set of n jobs, where each job has a processing requirement p_j, release date r_j and weight w_j. The goal is to find a preemptive schedule which minimizes the sum of weighted flow-time of jobs, where the flow-time of a job is the difference between its completion time and its released date. We give the first pseudo-polynomial time constant approximation algorithm for this problem. The algorithm also extends directly to the problem of minimizing the ℓ_p norm of weighted flow-times. The running time of our algorithm is polynomial in n, the number of jobs, and P, which is the ratio of the largest to the smallest processing requirement of a job. Our algorithm relies on a novel reduction of this problem to a generalization of the multi-cut problem on trees, which we call `Demand MultiCut` problem. Even though we do not give a constant factor approximation algorithm for the `Demand MultiCut`

problem on trees, we show that the specific instances of `Demand MultiCut` obtained by reduction from weighted flow-time problem instances have more structure in them, and we are able to employ techniques based on dynamic programming. Our dynamic programming algorithm relies on showing that there are near optimal solutions which have nice smoothness properties, and we exploit these properties to reduce the size of DP table.

In the unsplittable flow problem on a path, we are given a path with capacities on its edges and a set of tasks where each task is characterized by a source and a sink vertex, a demand, and a profit. The goal is to find a subset of the tasks of maximum total profit such that all task demands from this subset can be routed simultaneously without violating the capacity constraints. [Bansal et al., STOC 2006] gave a quasi-polynomial time-approximation scheme if the task demands are in a quasi-polynomial range.

Finding a PTAS for it has remained an important open question. In this thesis, we make progress towards this goal. When the task densities—defined as the ratio of a tasks profit and demand—lie in a constant range we obtain a PTAS. We also improve the QPTAS of Bansal et al. by removing the assumption that the demands need to lie in a quasi-polynomial range. Our third result is a PTAS for the case where we are allowed to shorten the paths of the tasks by at most an ϵ-fraction. Each of these results critically uses a sparsification lemma which we believe could be of independent interest. The lemma shows that in any (optimal) solution there exists an $O(\epsilon)$-fraction (measured by weight) of its tasks whose removal creates, on each edge, a slack which is at least as large as the $(1/\epsilon)$-th largest demand using that edge. This slack can then be used to allow slight errors when estimating or rounding quantities arising in the computation.

सार

इस थीसिस में, हम एक मशीन पर नौकरियों के समय निर्धारण की शास्त्रीय समस्याओं पर विचार करते हैं

एक पथ पर भारित प्रवाह और अस्थिर प्रवाह समस्या को कम करें। सभी समस्याओं

हम अध्ययन ऑफ़लाइन सेटिंग में हैं। इन परिणामों के लिए, हम उपन्यास की खोज पर भी भरोसा करते हैं

समस्याओं को कवर करने और गतिशील प्रोग्रामिंग में विचारों के माध्यम से शोषण करने के लिए कनेक्शन।

एक मशीन पर भारित प्रवाह-समय की समस्या में, हमें n नौकरियों का एक सेट दिया जाता है,

जहां प्रत्येक कार्य में प्रसंस्करण आवश्यकता pj, रिलीज़ दिनांक rj और भार wj है। लक्ष्य

एक पूर्वनिर्धारित अनुसूची खोजना है जो नौकरियों के भारित प्रवाह के समय को कम करता है,

जहां नौकरी के प्रवाह का समय उसके पूरा होने और उसके जारी होने के बीच का अंतर है

दिनांक। हम इसके लिए पहला छद्म-बहुपद समय निरंतर सन्निकटन एल्गोरिदम देते हैं

संकट। एल्गोरिथ्म भी सीधे ℓ_p के मानक को कम करने की समस्या तक फैलता है

भारित प्रवाह-समय। हमारे एल्गोरिथ्म का चल रहा समय n में बहुपद है, संख्या

नौकरियों की, और पी, जो सबसे छोटी प्रसंस्करण आवश्यकता के लिए सबसे बड़ा का अनुपात है

एक नौकरी। हमारा एल्गोरिथ्म इस समस्या के एक उपन्यास में कमी पर निर्भर करता है

पेड़ों पर बहु-कट समस्या, जिसे हम मांग मल्टीकूट समस्या कहते हैं। भले ही

हम डिमांड मल्टीकूट के लिए एक निरंतर कारक सन्निकटन एल्गोरिदम नहीं देते हैं

पेड़ों पर समस्या, हम दिखाते हैं कि डिमांड मल्टीकॉट के विशिष्ट उदाहरण प्राप्त हुए हैं

भारित प्रवाह-समय की समस्या के उदाहरणों में कमी से उनमें और अधिक संरचना होती है, और

हम गतिशील प्रोग्रामिंग के आधार पर तकनीकों को नियोजित करने में सक्षम हैं। हमारी गतिशील प्रोग्रामिंग

एल्गोरिथ्म यह दिखाने पर निर्भर करता है कि इष्टतम समाधान के पास हैं जो अच्छा है

चिकनाई गुण, और हम डीपी तालिका के आकार को कम करने के लिए इन गुणों का फायदा उठाते हैं।

एक पथ पर अस्थिर प्रवाह समस्या में, हमें इसकी क्षमता के साथ एक रास्ता दिया जाता है

किनारों और कार्यों का एक सेट जहां प्रत्येक कार्य को एक स्रोत और एक सिंक वर्टेक्स, एक मांग,

और एक लाभ। लक्ष्य ऐसे अधिकतम लाभ के कार्यों का एक सबसेट खोजने के लिए है

इस सबसेट से सभी कार्य मांगों को क्षमता का उल्लंघन किए बिना एक साथ रूट किया जा सकता है

बाधाओं। [बंसल एट अल।, एसटीओसी 2006] ने एक अर्ध-बहुपद समय-अनुमान दिया

योजना अगर कार्य की मांग एक अर्ध-बहुपद श्रेणी में हो।

इसके लिए पीटीएएस खोजना एक महत्वपूर्ण खुला प्रश्न बना हुआ है। इस थीसिस में, हम बनाते हैं

इस लक्ष्य की ओर प्रगति। जब कार्य घनत्व - एक कार्य लाभ के अनुपात के रूप में परिभाषित किया गया है

और मांग - एक निरंतर सीमा में झूठ हम एक पीटीएएस प्राप्त करते हैं। हम भी QPTAS में सुधार करते हैं

बंसल एट अल। इस धारणा को हटाकर कि मांगों को एक अर्ध-बहुपद में निहित होना चाहिए

रेंज।

हमारा तीसरा परिणाम उस मामले के लिए एक पीटीएएस है जहां हमें रास्ते को छोटा करने की अनुमति है

अधिकांश ϵ- भिन्न के द्वारा कार्यों का। इनमें से प्रत्येक परिणाम गंभीर रूप से एक स्पार्सिफिकेशन का उपयोग करता है

लेम्मा जिसे हम मानते हैं कि स्वतंत्र हित हो सकता है। लेम्मा से पता चलता है कि किसी भी में

(इष्टतम) समाधान मौजूद है एक O (ϵ) - (वजन से मापा जाता है) इसके कार्यों का

हटाने से प्रत्येक किनारे पर एक स्लैक बनता है, जो कम से कम (1 / ϵ) जितना बड़ा होता है

उस किनारे का उपयोग करके मांग करें। इस स्लैक का उपयोग तब किया जा सकता है जब अनुमान लगाते समय थोड़ी सी भी त्रुटि हो

या गणना में उत्पन्न होने वाली गोलाई मात्रा।

Contents

List of Figures