Lecture 9

02 February 2018 16:15

We shall see a PTAS for $\underbrace{P \| C_{max}}$

Graham's 3-field notation
P : Parallel machines
C_{max} : Max completion time

Warmup: $2 \| C_{max}$

Given 2 machines and a set of jobs $J = \{j_1, j_2, \ldots, j_n\}$, find a partition of J into $J_1 \uplus J_2$ such that
$$\max\left\{\sum_{j \in J_1} p_j, \sum_{j \in J_2} p_j\right\}$$ is minimized.

Define $L = \max\left\{\frac{\sum_{j \in J} p_j}{m}, \max_j p_j\right\}$. Note that L is a lower bound on OPT.

Now, let all jobs of size $\leq \epsilon \cdot L$ be small jobs and the rest be big jobs.

Claim: Given an OPTIMAL schedule for big jobs, a greedy assignment of the small jobs can be done with the increase in makespan at the most $\epsilon L \leq \epsilon T^*$.

Proof: Assume for the sake of contradiction that after the assignment of a small job j, the makespan of a m/c i is bigger than $T^* + \epsilon L$.

$\Rightarrow$ Load on i before assignment of $j > T^*$

$\Rightarrow$ " " all m/c's before assignment of $j > T^*$

$\Rightarrow$ Total volume of jobs $> mT^*$

$\longrightarrow$ contradiction $\boxtimes$

Now we find a good schedule for big jobs. Since all big jobs

Now we find a good schedule for big jobs. Since all big jobs are of size larger than ϵL, total number of them is at most $(2/\epsilon)$.

Hence, total # possible 2-partitions of these $= (2)^{2/\epsilon}$
We enumerate over all of them and output the one with the minimum makespan.

Thm: There is a PTAS for $2||C_{max}$

Now on to the general case: $P||C_{max}$

First of all, let us guess the optimal solution $= T^*$

You should always be careful about guessing. Try to convince yourself that in this case, guessing is fine if you are willing to lose a factor $(1+\epsilon)$.

Small jobs: Like in the previous case, let all jobs with $p_j \le \epsilon T^*$ be small and the rest be Big jobs.
We want to find a schedule of the big jobs.

Let us define a rounded up instance $\tilde{I}$, where $\tilde{p}_j = (1+\epsilon)^k$ and
$$(1+\epsilon)^{k-1} < p_j \le (1+\epsilon)^k$$

Claim: There exists an optimal solution to $\tilde{I}$ with makespan $\le (1+\epsilon)T^*$

Fact: There are only $\log_{1+\epsilon}(1/\epsilon) = \mathcal{G}$ many sizes of jobs in $\tilde{I}$.
Call them classes.

Let us look at any machine in $\tilde{S}$. Since the makespan is at the most $\tilde{T} = (1+\epsilon)T^*$, the # jobs on this m/c $\le \frac{1+\epsilon}{\epsilon} = N$ (say)

Any m/c can be described using a "profile". For all this is

Any m/c can be described using a "profile". Formally, it is a vector

$$\vec{V} = \langle V_1, V_2, V_3, \ldots \quad \text{where} \quad V_i = \text{class of job in the } i^{th}$$

$$|\vec{V}| = Q^N = \left(\log_{1+\epsilon}\left(1/\epsilon\right)\right)^{(1/\epsilon + 1)} \text{[constant !!]} = C \text{ (say)}$$

Now, each machine will have a profile from the above defined vector space. Note that, a schedule can be completely described by specifying how many machines take a particular profile. Here, we are using the fact that m/c's are identical.

In other words, any schedule S is a vector of dimension C.

$$\vec{S} = \langle S_1, S_2, S_3, \ldots \ldots S_C \rangle$$

where S_j = #machines with profile j. [$j \in \vec{V}$], $0 \leq S_j \leq m$

Hence, total number of possible schedules $= m^C = m^{O_\epsilon(1)}$

$O_\epsilon(1)$: Fancy way of saying "constant dependent on ϵ"

We enumerate over all possible schedules and check for following consistency conditions:

i) $\sum_{j=1}^{C} S_j \leq m$ ii) $\sum_{j=1}^{C} S_j \cdot \vec{V_j}(i) = $ #jobs of class i

Return a consistent schedule whose makespan is at the most $(1+\epsilon)\tilde{T}$.

Thm: $P \| C_{max}$ admits a PTAS with runtime $(m)^{(\log_{1+\epsilon} 1/\epsilon)^{O(\frac{1}{\epsilon})}}$