Scheduling on parallel machines

Friday, 3 February 2023 10:04 AM

Setup:

n jobs
m identical machines
j^{th} job needs p_j units of time

Each job is available at time 0.

C_j : If job j completes at time

Our Objective:-

minimize $C_{max} = \max_{j=1,\dots,n} C_j$ (called makespan or length of schedule)

Local Search Algorithms:-

Set of local changes/moves that change on FEASIBLE sol^{n} to the other.

Algo 1:-

- Start with any schedule
- Consider job l that finishes last
- Is there a free machine that finishes l earlier? Check.

If 'YES' Transfer 'l' to the other machine.

equivalent to checking if $\exists$ machine that finishes its currently assigned job earlier than $C_l - p_l$.

- Repeat till last job to complete cannot be transferred.

Analysis:-

Lower Bound on optimal C^*_{max}

$$\boxed{C^*_{max} \geq \max_{j=1,\dots,n} p_j}$$

Also let $P = \sum_{j=1}^{n} p_j$.

Total m machines

On average P/m units of work on 1 machine $\Rightarrow \exists$ a machine with atleast P/m work.

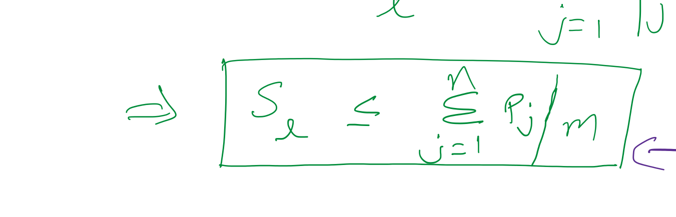
$$\Rightarrow \boxed{C^*_{max} \geq \sum_{j=1}^{n} p_j/m}$$

Let l be a job that completes last in final schedule

$\Rightarrow C_l$ is value of sol^{n}

$S_l = C_l - p_l$ is start time of l

so every machine (other than on which l was processed) busy from time 0 to S_l



Total work done here

$$m S_l \leq \sum_{j=1}^{n} p_j$$

$$\Rightarrow \boxed{S_l \leq \sum_{j=1}^{n} p_j/m}$$

Therefore from previous L.B. and

$$S_l \leq C^*_{max}$$

Also S_l to $C_l \leq C^*_{max}$

$\Rightarrow$ Makespan $\leq 2C^*_{max}$

Does it Converge Fast Enough?? Poly Time??

$\rightarrow$ C_{max} for sequence of schedules produced never increases

(If remain same $\Rightarrow$ no. of machines achieving the value decreases)

$\rightarrow$ Transfer a job to machine currently finishing earliest

Let C_{min} – completion time of machine that completes all its processing earliest

C_{min} never decreases

Claim:- We Never Transfer a Job twice

$\rightarrow$ Suppose NoT

say job j is transferred

$i \rightarrow i' \rightarrow i^*$

When j is transferred from $i \rightarrow i'$, starts at C_{min} for current schedule

Similarly, for $i' \rightarrow i^*$, it starts at C'_{min}

No change of machine i' between moves of j

$\Rightarrow C'_{min} < C_{min}$ (if transfer is better move)

Contradicts

$\Rightarrow$ Alg. ~~te~~ terminates in n iterations

Improved analysis:-

$$S_l \leq \sum_{j \neq l} p_j/m$$

Total length of schedule

$$\leq p_l + \sum_{j \neq l} p_j/m$$

$$= \left(1-\frac{1}{m}\right)p_l + \underbrace{\sum_{j=1}^{n} p_j/m}_{\leq C^*_{max}}$$

$$\leq \left(1-\frac{1}{m}\right)C^*_{max}$$

$$\boxed{\leq \left(2-\frac{1}{m}\right)C^*_{max}}$$

A Greedy Better Approximation:-

Longest Processing Time Rule

Sort the jobs in non-increasing order

Assign the jobs as soon as machine is available

Claim: A $4/3$ Approximation

Proof: By Contradiction!!

Suppose algo does not give $4/3$ approximation.

Consider the input which is a counterexample

Assume $p_1 \geq p_2 \geq \dots p_n$

Last job to complete is n?? (If not what??)

If $p_n \leq C^*_{max}/3$

Schedule length at most $(4/3)\,C^*_{max}$

Why??

$\Rightarrow$ Not a counter example

$\therefore p_n > C^*_{max}/3$

$\Rightarrow$ In optimal schedule, each machine may process at most 2 jobs

$\Rightarrow$ Such counterexample cannot exist.