


Greedy Algorithm --- Task Scheduling

Fractional Knapsack

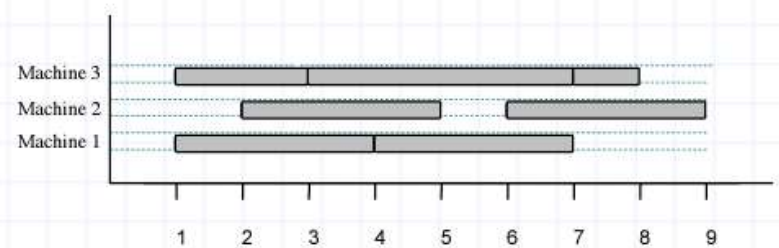
Task Scheduling

What is it?

Task Scheduling



- ◆ Given: a set T of n tasks, each having:
 - A start time, s_i
 - A finish time, f_i (where $s_i < f_i$)
- ◆ Goal: Perform all the tasks using a minimum number of “machines.”



How to solve/give algorithm, explain it, justify runtime, prove correctness by lower bound argument

Algorithm

Algorithm TaskSchedule(T):

Input: A set T of tasks, such that each task has a start time s_i and a finish time f_i

Output: A nonconflicting schedule of the tasks in T using a minimum number of machines

$m \leftarrow 0$ // optimal number of machines

while $T \neq \emptyset$ **do**

 remove from T the task i with smallest start time s_i

if there is a machine j with no task conflicting with task i **then**
 schedule task i on machine j

else

$m \leftarrow m + 1$ // add a new machine

 schedule task i on machine m

Algorithm 10.6: A greedy algorithm for the task scheduling problem.

Justify runtime

This Algo uses heap-based PQ to store task in T . Thus, remove Task i from T with smallest

start time s_i once, will take $O(\log n)$ time.

While-loop takes $O(n)$ in worst case.

Therefore, runtime: $O(\log n) * O(n) = O(n \log n)$

Prove correctness by lower bound argument

Proof:

TaskSchedule Algo use k Machines to schedule k tasks.

Let i be the first task scheduled on Machine k .

Assume we can only use $k - 1$ Machines.

When we schedule task i on $1 \sim k-1$ Machines, we find that task i conflicts with all other tasks on these Machines.

Therefore, it is impossible for use to schedule all tasks in T using only $k-1$ machines. K is the minimal number of machines for all tasks in T .

This proof uses lower-bound argument.

Difference between Task Scheduling and Telescope Scheduling problems

1: Greedy Algo vs Dynamic programming

2: Greedy 算法特点 vs Simple Subproblem, Subproblem Optimality, Subproblem Overlap

3: Greedy deals with conflict. When a task conflict with other task on all machines, Task Scheduling problem is allowed to add a new machine to solve tasks.

However, for Telescope Schedule, we only have one telescope or Machine to solve items. So we must use a Algo which can solve problem on a global view.