(1) Introduction

3 Motivating Examples:

1) Scheduling in CPU.

We want to be able to schedule the various processes which are running concurrently on a computer (i.e. Allocate chunks of CPU time towards each process)

We can have multiple objectives:

- Minimize average waiting time, or
- Maximize throughput.

Our tasks (i.e. The processes) have multiple properties:

- Processing time - Time needed to finish the proc
- Priorities - Weights indicating importance

Further,

- Tasks need not be known beforehand; in fact, neither does the processing time (These are known as real-time problems, we will mainly focus on problems with known quantities)
- Tasks need not run in contiguous chunks. They can be interrupted, and resumed at a later time. (This is known as preemption)

→ Ex. 2, 3

2) Production Planning.

The Carts-for-U factory produces 3 types of carts:

- Shopping Carts - Airport Trolleys - Cargo Carts

Production of each type of cart goes through the same stages:

- Producing metal skeleton
- Coating Metal
- Attaching parts to the skeleton
- Packaging of carts

Each stage consists of the same machines, not necessarily identical (i.e. could differ in speed or other capabilities)

We could have multiple objectives:

- Minimize # of late demands
  (The "demand" for each cart is the # required and due date)
- Minimize the "total penalty" of late demands
  (i.e. some demands may not be met, but we want to minimize lateness)
- Minimize maximum lateness of demands.

3) Gate Assignment at an airport

An airport has m gates where planes land and take off. We assume all gates are identical and there is a hanger where planes can [redacted] wait. Suppose plane j lands at time r_j, departs at time d_j and takes time $P_{j,i,i'}$ to go from gate i to i'. How should planes be assigned to gates to minimize total airline delay?

1.1

Def'n (Job, Machine)

Every scheduling problem consists of:

- Jobs - Entities that require resources. We will commonly use j, k, l to denote jobs.
- Machines (M/C) - Resources that jobs require. We will use i, i', i'', etc. to denote M/Cs.

- Jobs:

A job j typically is associated with:

- Processing time (P_{ij})

  This is the time taken by machine i to process job j.

  Note:

  1) Always ≥ 0
  2) Can be infinity (i.e. This machine cannot process job j)
  3) If machines are identical, then j's processing time is independent of i, and we will use P_j to denote j's processing time.

- Release Date (r_j)

  Time when j enters the system.

  Note: j can only be scheduled on a machine at or after its release date

- Due Date (d_j)

  Time by which j must be completed

  Note: Could be a soft deadline that can incur penalty in our objective fn. or hard deadline

- Weight (w_j)

  Indicates the importance / priority of job j.

  $w_j \geq 0$ unless otherwise stated.

→ Machines

- Machines:
We can have multiple settings/environments:
  - Single M/C
  - Multiple M/C
    - Identical machines
    - Non-identical machines
  - Shop Scheduling: Multiple machines and a job consists of various operations that need to be performed on various machines (Could be a sequence - See Ex. 2)

- Objective
1) Sum of completion times:
Let C_j denote the completion time of job j. We write:

$$\min \sum_j C_j$$

2) Weighted Completion time:

$$\min \sum_j w_j C_j$$

3) Total Waiting time:

$$\min \sum_j (C_j - r_j)$$

4) Maximum lateness:
Let $L_j := C_j - d_j$ be the lateness of j.

$$\min . \max_j L_j$$

5) No. of late jobs.
Define $u_j := \begin{cases} 1 & C_j > d_j \\ 0 & \text{otherwise} \end{cases}$

$$\min \sum u_j$$

Scanned with CamScanner