Crew Scheduling

Outline

- Crew Scheduling
- Work Rules and Policies
- Manual Scheduling Process
- Model Formulation
- Automated Scheduling Experience

Crew Scheduling Problem

Input

- A set of vehicle blocks each starting with a pull-out and ending with a pull-in at the depot
- Crew work rule constraints and pay provisions

Objective:

- Define crew duties (aka runs, days, or shifts) covering all vehicle block time so as to:
  - minimize crew costs

Crew Scheduling Problem

Constraints:

- Work rules: hard constraints
- Policies: preferences or soft constraints
- Crews available: in short run the # of crews available are known

Variations:

- different crew types: full-time, part-time
- mix restrictions: constraints on max # of part-timers

Typical Crew Scheduling Approach

Three-stage sequential approach:

1. Cutting long vehicle blocks into pieces of work
2. Combining pieces to form runs
3. Selection of minimum cost set of runs

Manual process includes only steps 1 and 2; optimization process also involves step 3

Typical Crew Scheduling Approach

Cutting Blocks:

- each block consists of a sequence of vehicle revenue trips and non-revenue activities
- blocks can be cut only at relief points where one crew can replace another.
- relief points are typically at terminals which are accessible
- avoid cuts within peak period
- resulting pieces typically:
  - have minimum and maximum lengths
  - should be combinable to form legal runs

Vehicle Block Partitions

Definition: a <u>partition</u> of a block is the selection of a set of cuts each representing a relief

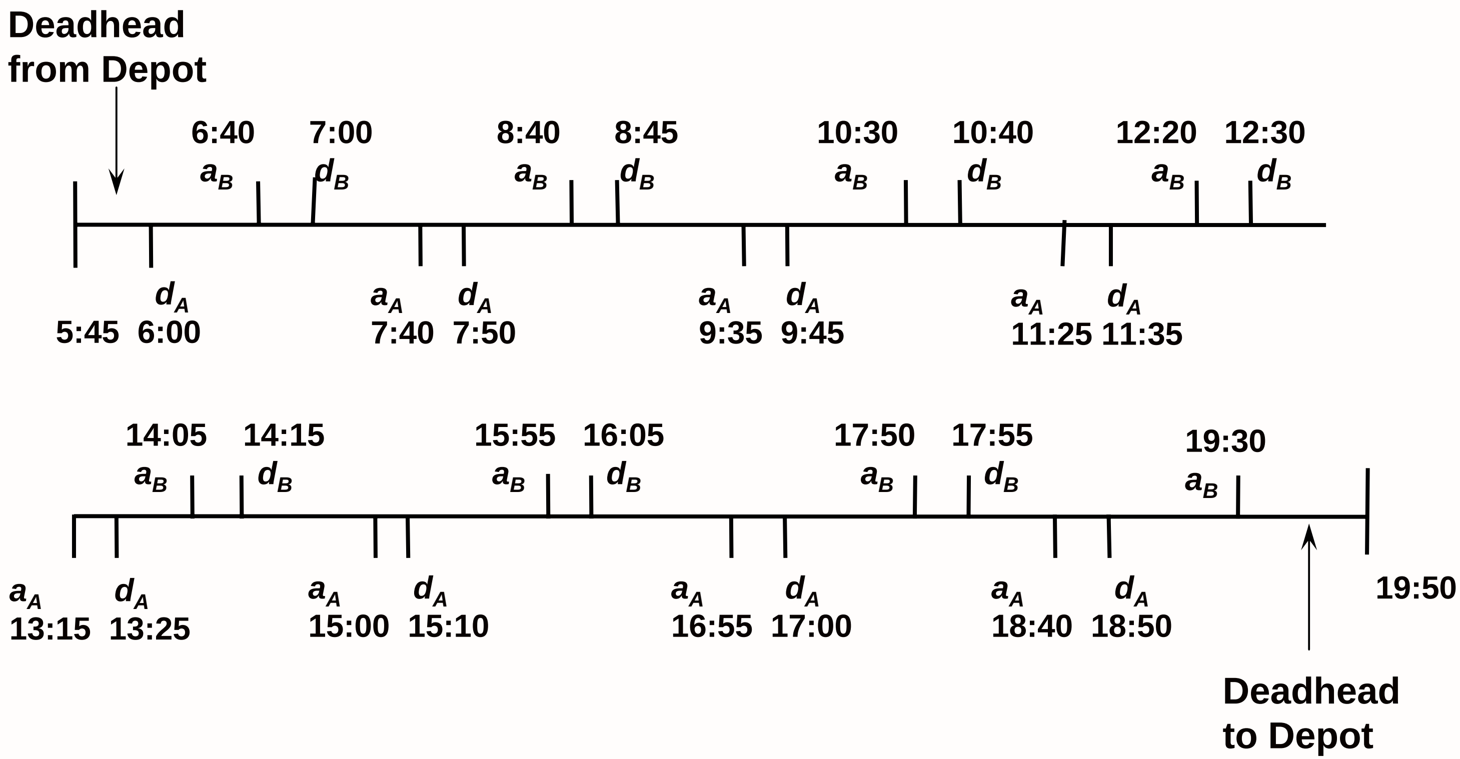
Key problems:

- very hard to evaluate a partition before forming runs
- many partitions are possible for any vehicle block

Possible Approaches:

- generate only one partition for each vehicle block
- generate multiple partitions for each vehicle block
- generate all possible partitions for each vehicle block

A Vehicle Block on Route AB



d_i = departure time from terminal i

a_i = arrival time at terminal i

Combining Pieces of Work to Form Runs

- **Large number of feasible runs by combining pieces of work**
- **Work rules are complex and constraining:**
  - **maximum work hours: e.g. 8 hrs 15 min**
  - **minimum paid hours - guarantee time: e.g. 8 hrs**
  - **overtime constraints and pay premiums: e.g. 50% pay premium**
  - **spread constraints and pay premiums: time between first report and last release for duty, e.g.**

	P_1			P_2	
6:00	•———•	**10:00**	**14:00**	•———•	**18:00**

has a spread of 12 hours

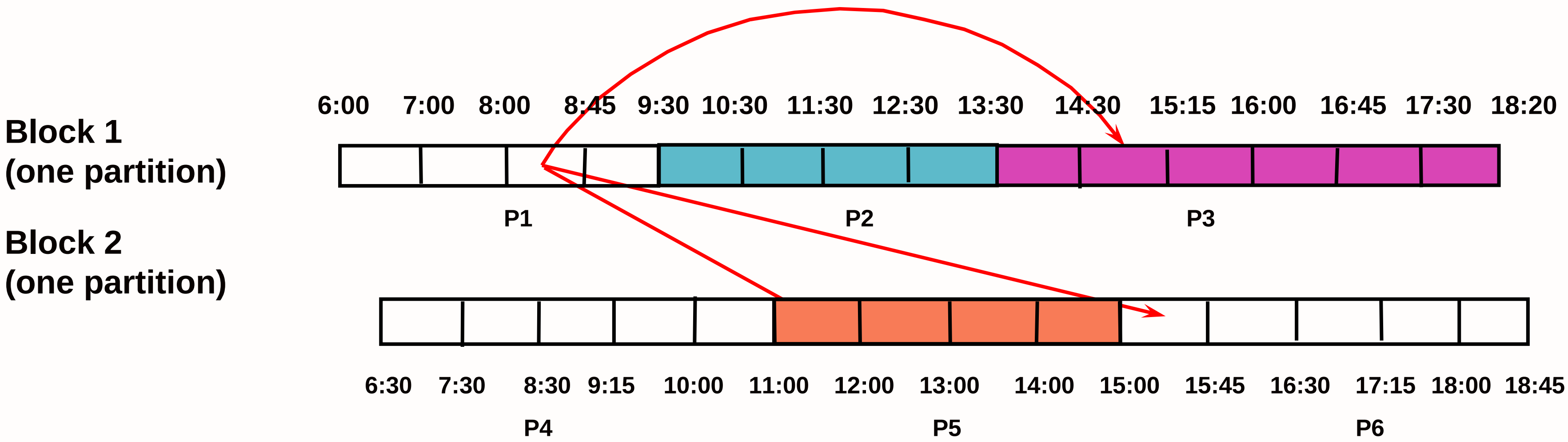
Combining Pieces of Work to Form Runs

- swing pay premiums associated with runs with pieces which start and end at different locations, e.g.

- different types of duties
  - split: a two-piece run
  - straight: a continuous run
  - trippers: a short run, usually worked on overtime

Approach: generate and cost out each feasible run

Combining Pieces of Work to Form Runs

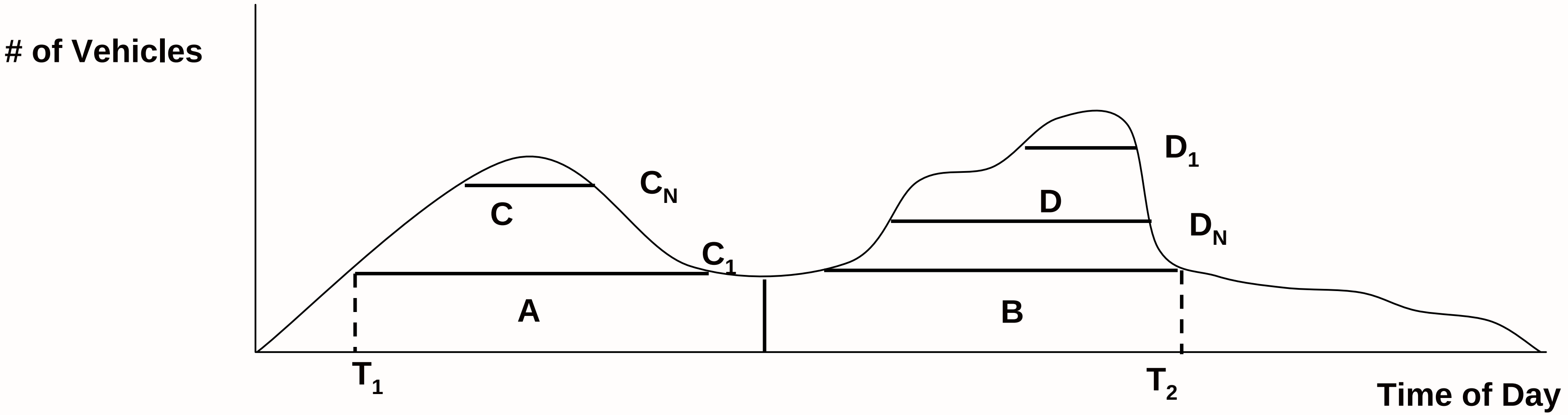


Possible Runs from defined pieces P1-P6:

Run #	1st piece	2nd piece	Spread Time	Work Time	Cost
1	P1	P2	7:30	7:30	C1
2	P1	P3	12:20	8:20	C2
3	P1	P5	9:00	7:30	C3
4	P1	P6	12:45	7:15	C4
5	P2	P3	8:50	8:50	C5
6	P2	P6	9:15	7:45	C6
7	P4	P3	11:50	9:20*	---
8	P4	P5	8:30	8:30	C8
9	P4	P6	12:15	8:15	C9
10	P5	P6	7:45	7:45	C10

* illegal run: max work time violation

Crew Scheduling: Manual Techniques



T_1 is earliest AM pullout which can still serve PM peak

T_2 is latest PM pullback which can still serve AM peak

A are AM straights (or short split runs)

B are PM straights (or short split runs)

C and D are long split runs

Typical Sequence

1. Based on total vehicle hours estimate total operators required
2. Determine # operators required in AM and PM peaks
3. Determine B based on # of pull-ins after time T_2.
4. Determine # split runs (# of PM Peak Vehicles - B)
5. Determine A based on # of AM Peak Vehicles - split runs
6. Combine earliest pullouts in C with earliest pull-ins in D to produce minimum spread splut runs C_1D_1. Iterate until all split runs are matched C_ND_N.

Example

Time Period	# Vehicles	Period Length	# Vehicle Hours
AM Peak	8	3	24 →AM duties = 4
Base	4	6	24
PM Peak	8	3	24 →split duties = 4
Evening	4	6	24 →PM duties = 4
			96, or 12 FTOs

Selection of Minimum Cost Set of Runs

- Usually built around mathematical programming formulation

Problem Statement:

Given a set of *m* trips and a set of *n* feasible driver runs, find a sub-set of the *n* runs which cover all trips at minimum cost

Mathematical Model for Crew Scheduling Problem

A. Basic Model: Set Partitioning Problem

Notation:

- P = set of trips to be covered
- R = set of feasible runs
- c_j = cost of run j
- δ_i^j = binary parameter, if 1 means that trip i is included in run j, 0 o.w.
- x_j = binary decision variable, if 1 means run j is selected, 0 o.w.

$$\text{Min} \quad \sum_{j \in R} c_j x_j$$

$$\text{Subject to:} \quad \sum_{j \in R} x_j \delta_i^j = 1 \qquad \forall i \in P$$

$$x_j \in \{0,1\}, \qquad \forall j \in R$$

Mathematical Model for Crew Scheduling Problem

Problem size:

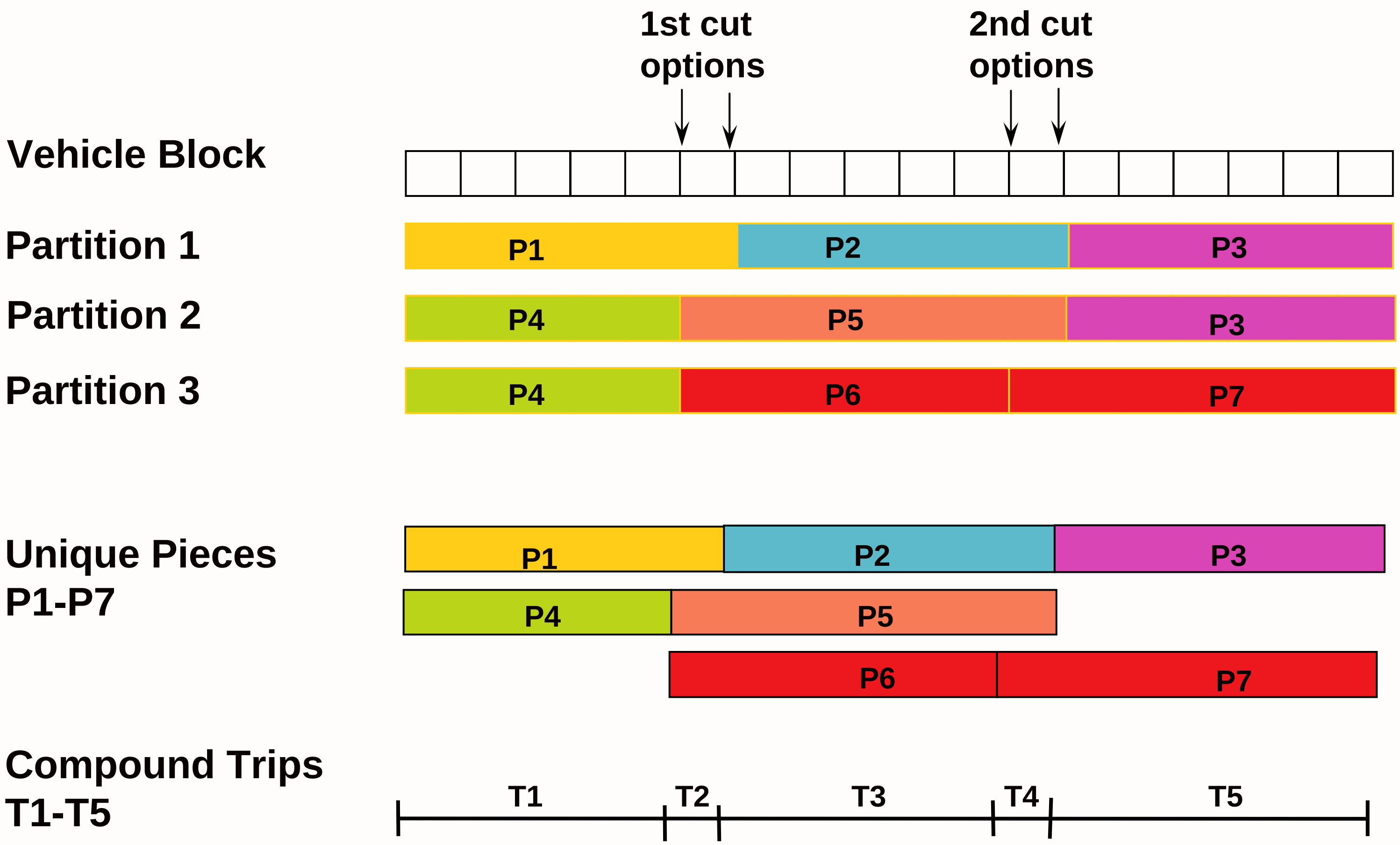
R decision variables (likely to be in millions)

P constraints (likely to be in thousands)

Problem size reduction strategy:

- replace individual trips with compound trips consisting of a sequence of vehicle trips which will always be served by a single crew.

Partitions of Vehicle Block, Pieces of Work and Compound Trip



May reduce the # of constraints but by less than one order of magnitude

Variations of Set Partitioning Problem

1. Set R consists of all feasible runs given all feasible partitions for all vehicle blocks
   - size of model, specifically # of columns, explodes with problem size
   - only possible for small problems
2. Set R consists of a subset of all feasible runs
   - not guaranteed to find an optimal solution
   - effectiveness will depend on quantity and quality of runs included
3. Column generation based on starting with a subset of runs and generating additional runs which will improve the solution as part of the model solution process.

Model with Side Constraints

Often the number (or mix) of crew types is constrained in various ways which can be formulated as side constraints

Example: Suppose total tripper hours are constrained to be less than 25% of timetable time.

Let: WT = total timetable time
R^T = set of tripper runs
t_j = work time for tripper run j

Then the additional constraint is:

$$\sum_{j \in R^T} t_i x_i \leq 0.25\, WT$$

Experience with Automated Crew Scheduling Systems

- Virtually universally used in medium and large operators world-wide
- Two most widely used commercial packages are HASTUS (by GIRO Inc in Montreal) and Trapeze (by Trapeze Inc in Toronto), each with over 200 customers world-wide
- Typical cost ranges from $100K to $2 mill for the software
- Key benefits of automated scheduling are:
  - scheduling process time reductions
  - improved accuracy
  - modest improvements in efficiency (typically 0-2%)
  - provides a key database for many other IT applications

Experience with Automated Crew Scheduling Systems

- Evolution of software has been from "black box" optimization/heuristics to highly interactive and graphical tools
- Current systems allow much greater ability to "shape" the solution to the needs of specific agencies
- One implication however is a profusion of these "soft" parameters which means greater complexity and it is very hard to get full value out of systems.