Crew Scheduling

Housos Efthymios, Professor

Computer Systems Laboratory (CSL)

Electrical & Computer Engineering

University of Patras

Overview

- Crew Scheduling Problem Definition
- CSL Prototypes & Experience
- Airline Crew Scheduling
- Bus & Driver Shift Scheduling
- OR Modeling & Solution Approach
- Column Generation Approach
- Discussion

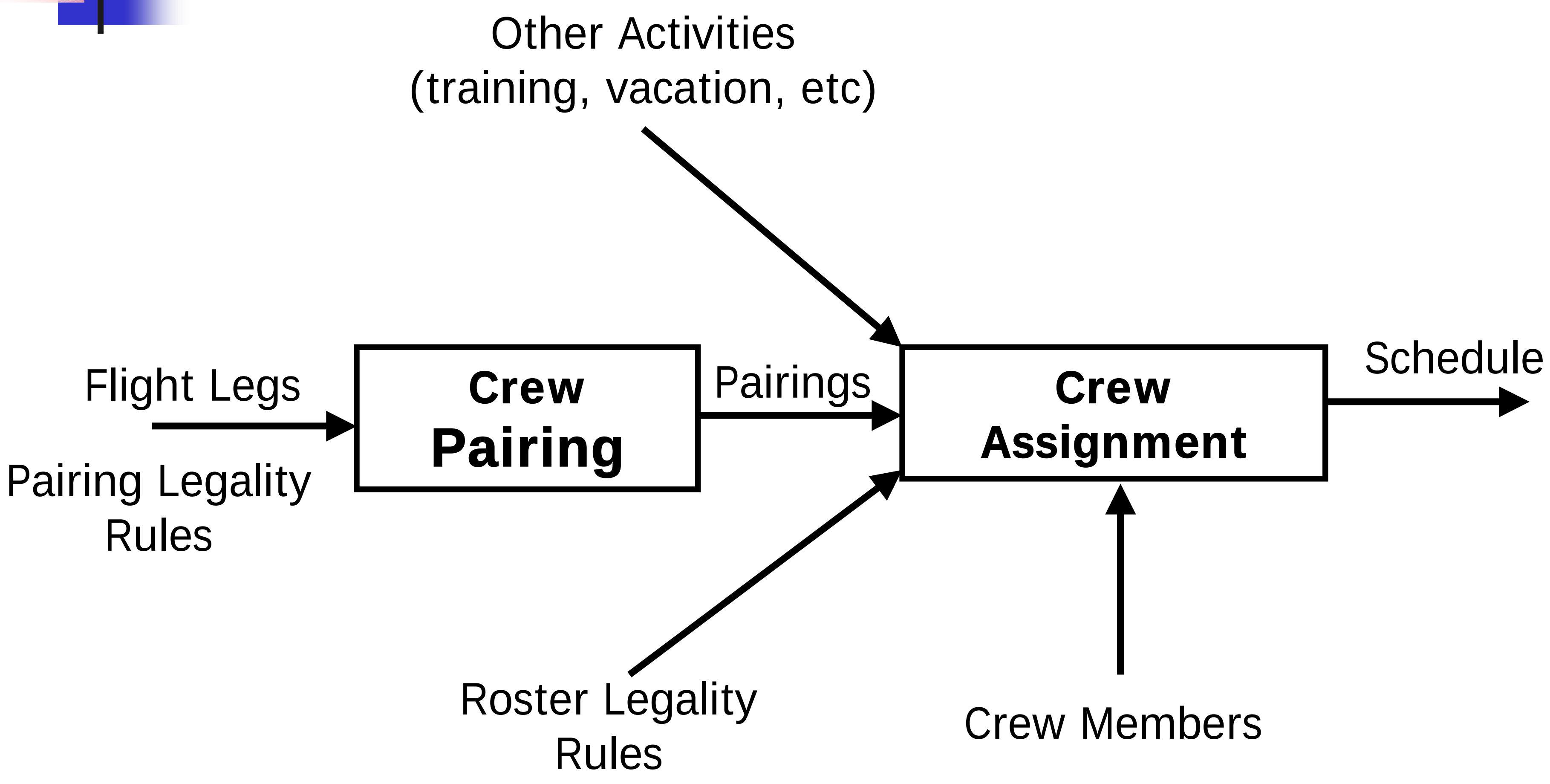
Crew Scheduling Problem Definition

- Assignment of well-defined tasks (pairing & shift construction) to a group of people while respecting a set of complicated legality rules and resource constraints.
- Most of the legality rules are non-linear and evolving through time

CSL Prototypes & Experience

- Airline Crew Scheduling (Pairing Construction & Crew Assignment)
- Bus & Driver Shift Scheduling

Airline Crew Scheduling Problem



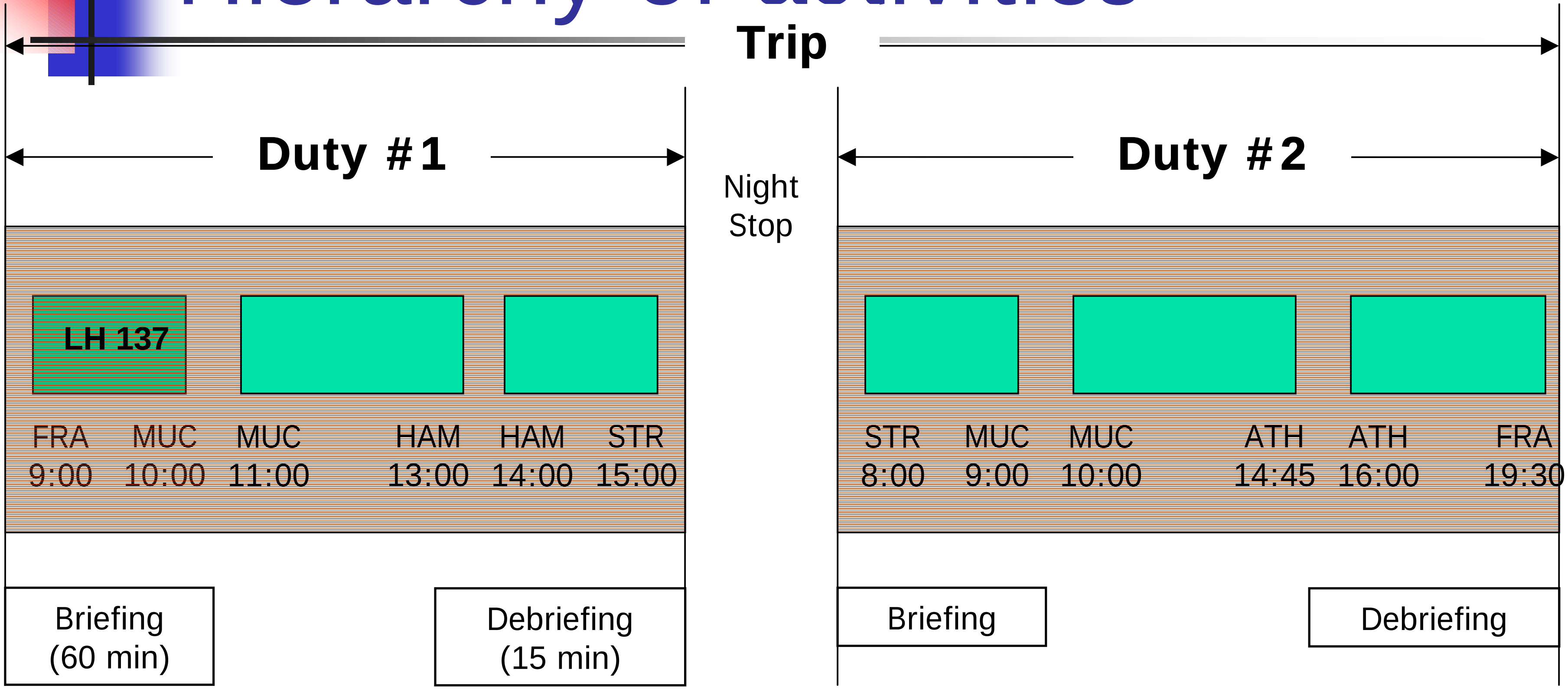
Crew Pairing Solution Methodology

- Crew Pairing and Crew assignment are too big to be solved together
- A good solution for Crew Pairing is a must for the efficient and productive use of the airline crews

Airline Crew Scheduling

- Entities of the problem
  - Flight Leg: A non-stop flight with its crew complement and fleet requirements
  - Duty: A legal sequence of legs for one day
  - Pairing (Trip): A legal sequence of duties
    - Pairings start and end at the same crew base
  - Roster: A set of pairings and other activities assigned to a specific crew member

Hierarchy of activities



Bus & Driver Shift Scheduling

- Solved every afternoon for the work load of the next day
- Shift: a set of routes that will be performed by a bus and its associated driver in a day
- Shifts must be legal according to a complex set of rules while respecting previous bus-ending points
- A good solution for the problem is a legal set of shifts that efficiently covers the work load
    - (more later)

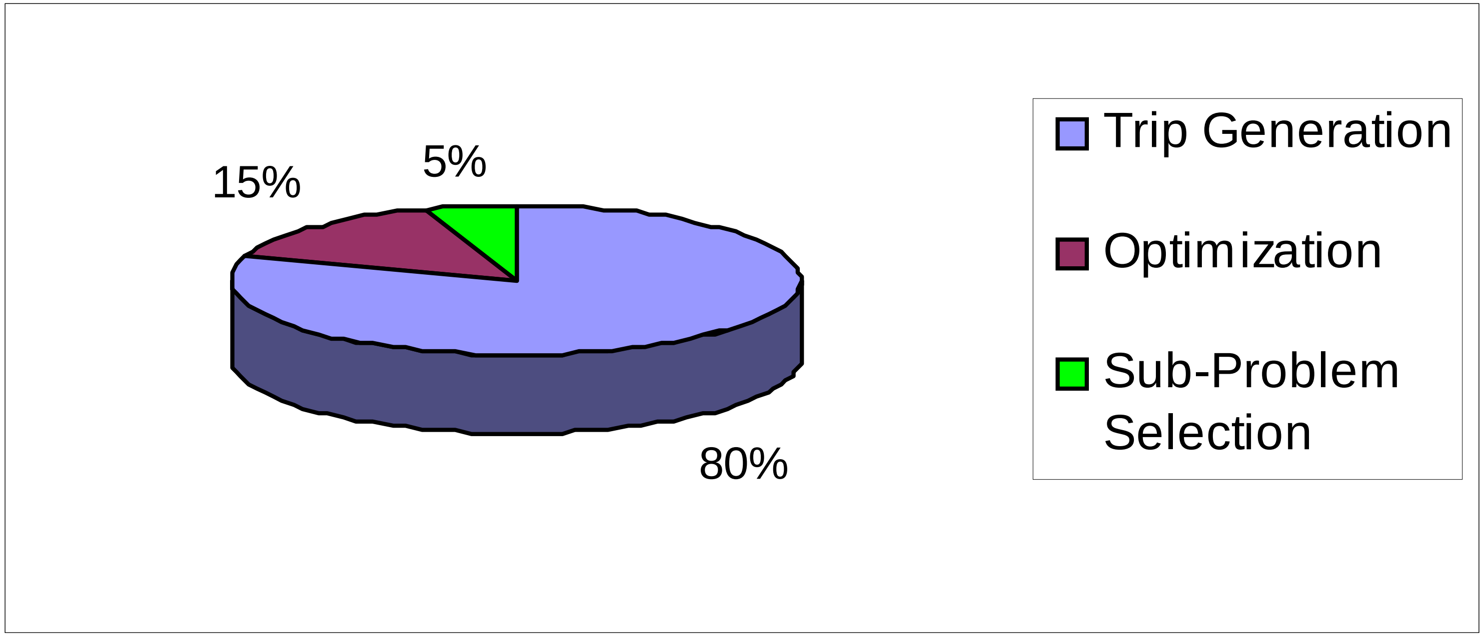
Solution Approaches for the Crew Pairing Problem

- Generate and Optimize
  - Select sub-problems (Heuristic filtering)
  - Phase 1. Generate a large set of legal pairings (**Generate**)
  - Phase 2. Select the best pairings (**Optimize**)
  - Iterate
- Column Generation

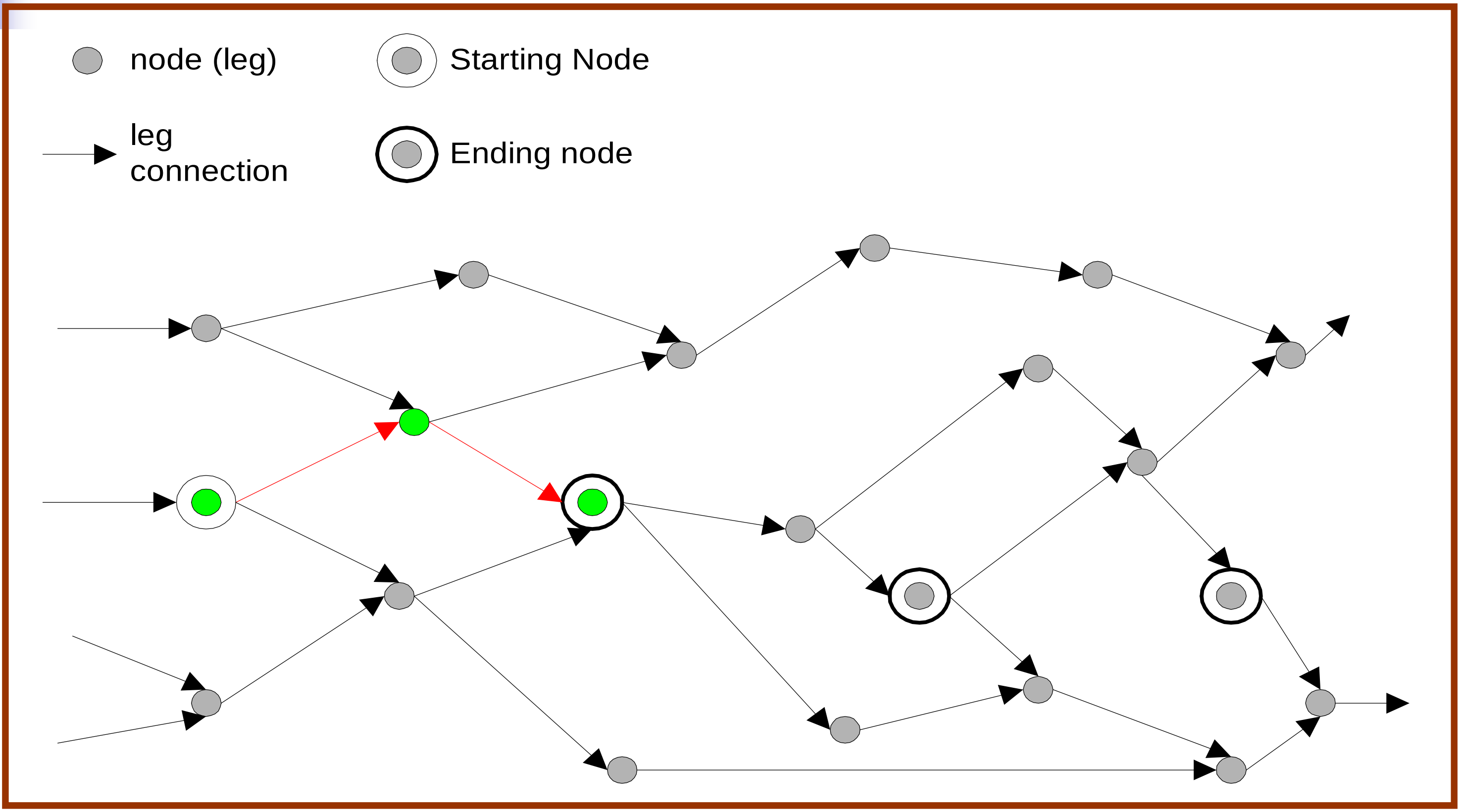
Generate and Optimize in Production (CARMEN)

- Initially used in CARMEN’s Pairing Construction System (PAC)
- In use since 1995 by most European Airlines
- Clever sub-problem selection filters and tools
- Day by Day (DbD) iteration process
- Efficient modeling of complex legality rules via a separate rule system

Time distribution of the Generate & Optimize Approach



Trip Generation Process

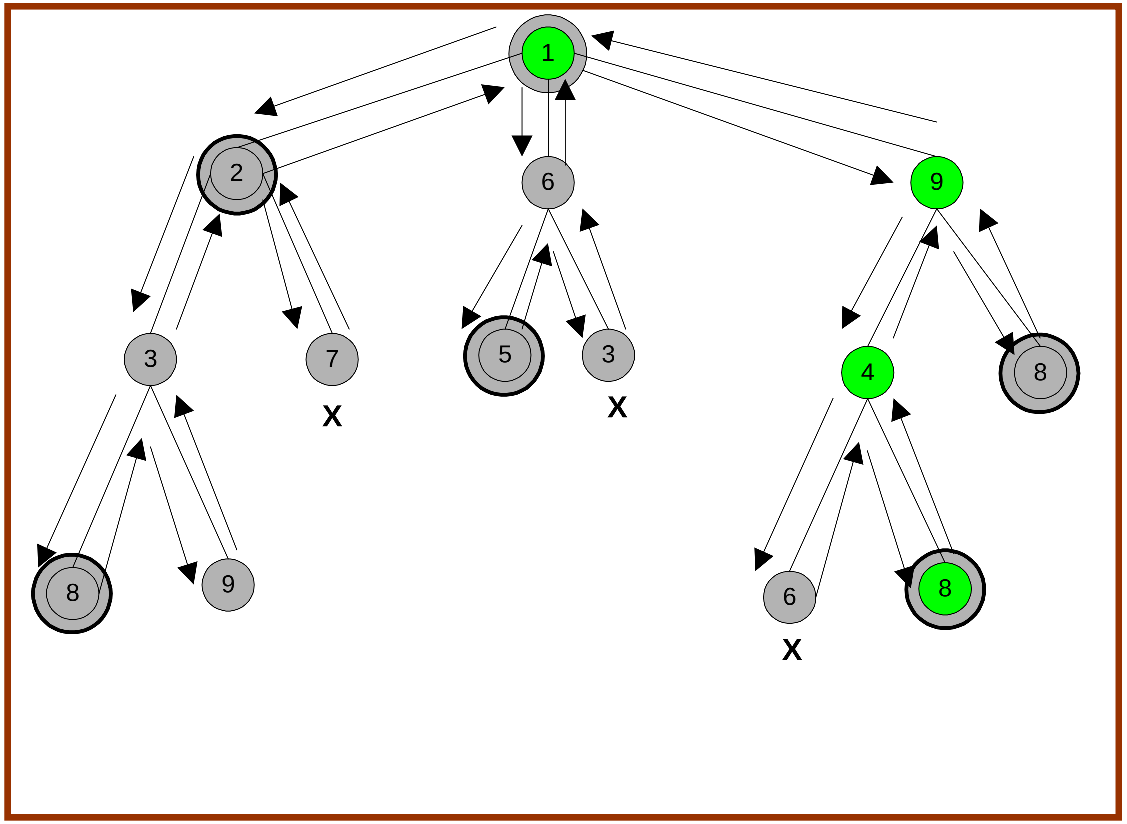


Connection matrix graph
(each leg appears only once)

Trip Generation Algorithm

Depth first search algorithm

- For each starting node a separate search tree is defined
- The DFS process is controlled by:
  - Search width
  - Maximum number of total trips
  - Maximum number of trips per starting node
  - Legality rules



Basic Procedure for Crew Scheduling Problems OR(1)

- Formulated as a Set Covering (SCP) or Set Partitioning (SPP) problems

$$\text{(SCP)} \quad \min\{cx : Ax \geq 1,\ x \in \{0,1\}^n\}$$

$$\text{(SPP)} \quad \min\{cx : Ax = 1,\ x \in \{0,1\}^n\}$$

OR Modeling Approach (2)

- A binary variable (column) represents a **legal** schedule of a person that covers a set of tasks
- Each variable (column) embeds all non-linear legality rules
- Legality rules are external to the model
- Constraints ensure the covering of all tasks

OR Modeling Approach (3)

- The airline crew pairing problem involves the finding of a set of trips that covers a set of flights with minimum cost

$$A = \begin{array}{cccccc} & \text{trip 1} & \text{trip 2} & \text{trip 3} & \dots & \text{trip n} \\ \text{flight 1} & 1 & 0 & 1 & \dots & 1 \\ \text{flight 2} & 1 & 1 & 0 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \text{flight m} & 0 & 1 & 1 & \dots & 1 \end{array}$$

$m = 10^2 – 10^4$
$n = 3*10^4 – 10^6$

OR Solution Approach

- Generate and Optimize
  - Generate a large number of ‘good’ legal columns and select the best ones
  - Generation of ‘good’ columns is a time-consuming task
  - Selection of ‘good’ columns requires an efficient IP Solver

Solution Approach (2)

- Small amount of RAM required for the generation phase
- Clever problem specific heuristics for sub-problem selection & the (DbD) solution strategy
- Powerful IP Optimizer (able to identify reasonable solutions from ~1,000,000 columns)

Solution Approach (3)

- Used in the production environment for many years by several European airlines
- Computer generated solution were often inferior to the ones of human experts and/or users could further improve the solution!
- Need to solve larger problems with stable heuristic processes

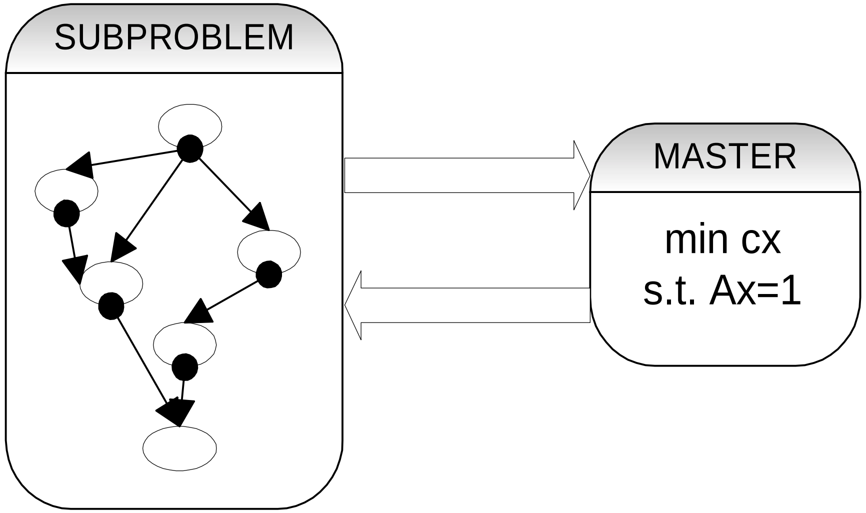
Column Generation (CG)

- Known for many decades for the solution of large LP problems
- Main Idea of CG approach:
  - Consider only a small number of variables at a time
  - Solve a small LP (**master problem**) and get a primal and a dual solution
  - Generate new attractive columns (**sub-problem**), with negative reduced cost, by using the dual solution of the **master problem** in order to improve the previous LP solution
  - Repeat the procedure until no further improvement can be made

Column Generation Requirements for LP & IP

- Efficient data structures for the implicit representation of all problem variables
  - Large amounts of RAM
- Fast algorithms for generation of new legal and promising columns
- LP Optimizer
- No need for strong IP optimizer!

Column Generation & Scheduling



- Master Problem ensures covering of tasks
- Sub problem usually has the structure of a graph
- Nodes are simple or composite activities (i.e. flights, duties)
- Arcs connect activities that are legal to be connected in pairs

Master Problem

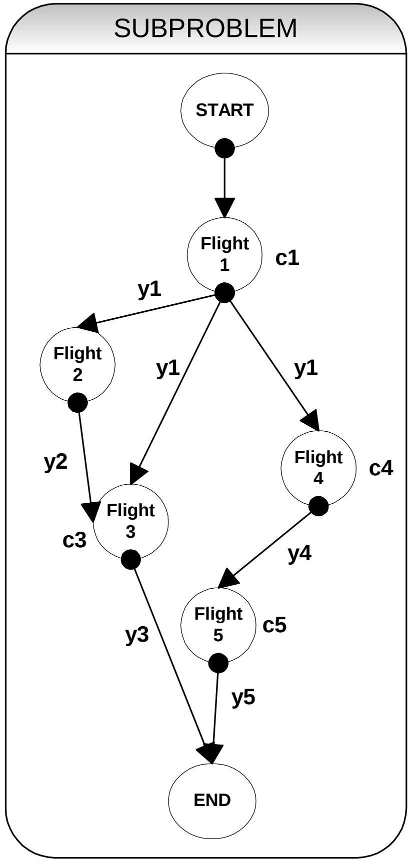
MASTER

min cx
s.t. Ax=1

- Relaxed IP model
- One constraint for each task
- In each step solve a problem that has the basis of previous iteration and the newly generated attractive columns
- Return primal and dual solution

			Schedule 1	Schedule 2	…	Schedule n
		Flight1	1	1	…	1
A	=	Flight2	1	0	…	0
		…	…	…	…	…
		Flight5	0	1	…	1

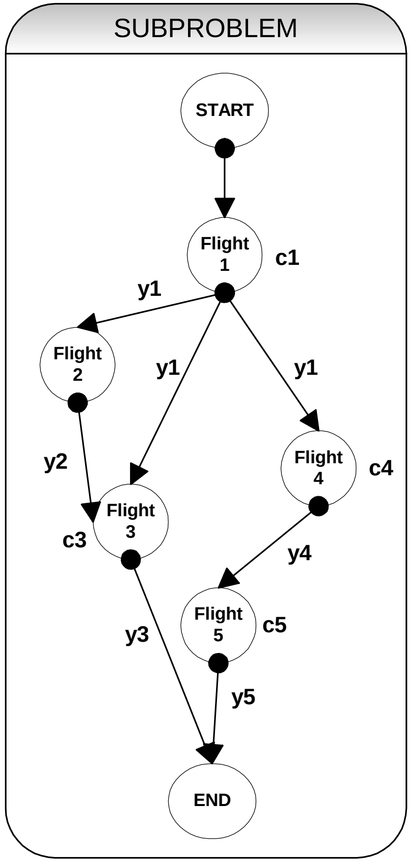
Sub-problem



- Basic structure is a graph or a connection matrix
- Nodes are the flights
- Arcs connect flights that can be legally connected as a pair
- Cost of a node is the cost of including the corresponding flight in some pairing
- Cost of an arc is the dual of the constraint of the source node flight

  “the source node is present for all possible pairings after this point”

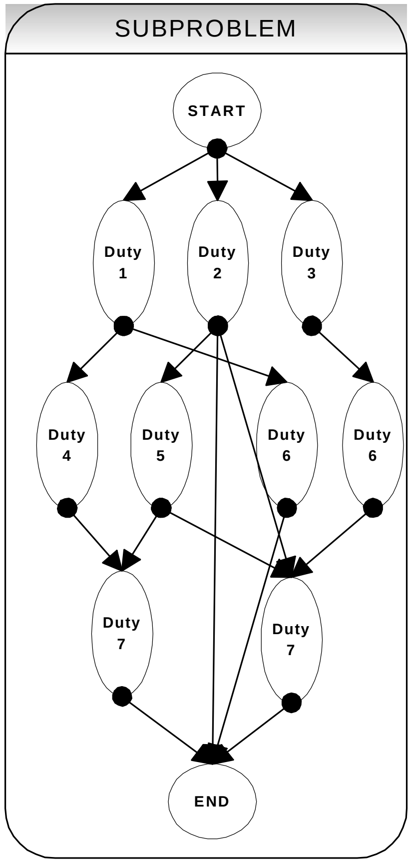
Sub-problem (2)



- Legality Rules
- The reduced cost of a new pairing (start to end) is the cost of the path
- Pairing: Flight1, Flight4, Flight5, has reduced cost (c1+c4+c5) – (y1+y4+y5)
- A k-shortest path type algorithm provides the best candidate pairings

- **ASSUMPTION:** The cost of a schedule is the sum of the costs of all flights (additive function)
  - Often OK even if cost is non-linear!

Duty Based Sub-problem



- Embed legality of duties
- Nodes of the network are legal duties
- Two duties that can be legally followed are connected with an arc
- Dual of each node is the sum of duals of the legs of the corresponding duty
- Cost of each node is the cost of the corresponding duty
- Number of nodes increases
- Number of arcs decreases
- Network is smarter and is easier to look for legal pairings

Search for new attractive pairings

- Sub-problem network (flight or duty) cannot embed all legality rules
- k-shortest path algorithms may produce a large number of illegal pairings!
- DFS + shortest path always produces legal pairings

Search for new attractive pairings (2)

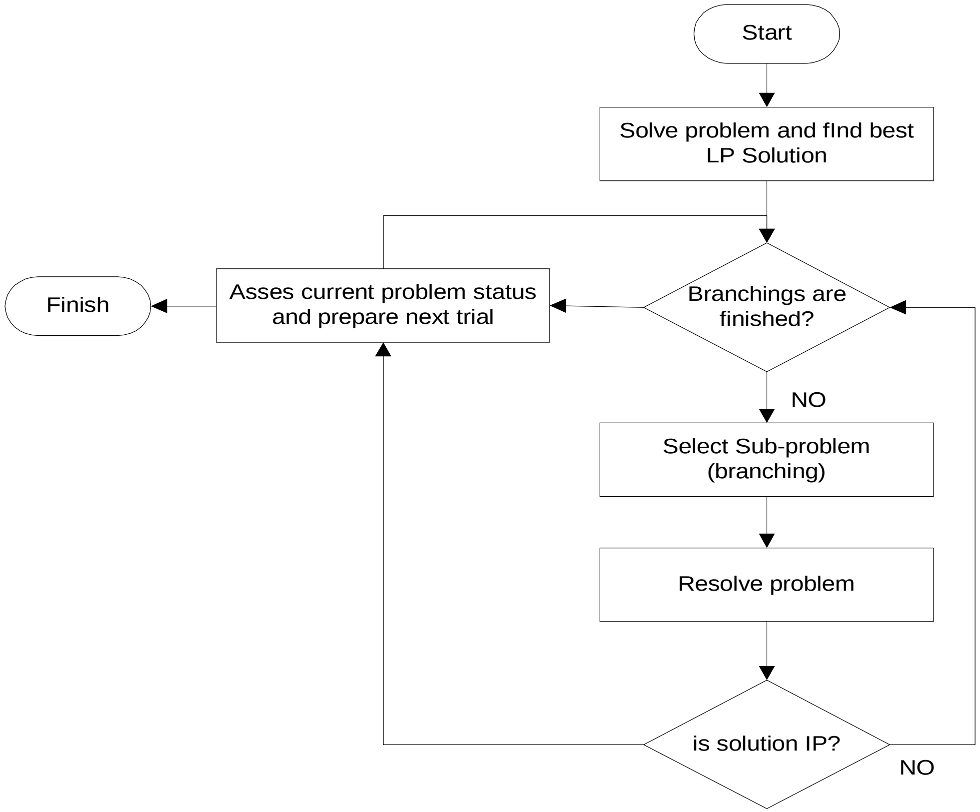
- Build new legal pairings using a depth first search procedure
- DFS proceeds using the shortest path results for each node

(more in the next presentation)

IP Solution (1)

- An LP (fractional) solution is always known but an IP solution is actually required
- Procedure for IP solution creation:
  - Reduce problem dimensions by freezing a part of the solution and re-applying the CG strategy on the remaining problem
  - At a certain point when problem dimensions are small an IP solution can be located with some other IP optimization method

IP Solution (2)



Discussion

- Sub-problem identification and the iterative process that will lead us to a good solution is the key to success
  - Intelligent domain specific criteria for the selection of sub-problems (DbD)
  - Problem independent strategy via the use of LP and duality theory (CG)