# 1.206J Airline Schedule Planning
# Problem Set #3
# Due date: 4/23/03

Instructions: Answer each part of the questions completely. You may use any books and other outside materials to work on the problems and you may work in groups. However, students should hand in their own write-up. Please answer the question as fully as possible (after all some do not have “right” answers, we are just seeing how you would approach and think about the problem) and explicitly state any assumptions that you are using.

## Problem #1: Generating Pairings on a Duty-Based Network

A typical crew pairing problem has hundreds of millions of feasible pairings. Therefore, we use *branch-and-price* to solve these problems. In class, we talked about how to generate pairings by solving a multi-label shortest path problem over a flight connection network. In this exercise, we develop an alternative approach to the crew pairing pricing problem.

For this exercise, assume the following rules:

- Duties cannot contain more than 8 hours of flying time.
- Duties cannot contain more than 14 hours of duty time.
- The minimum sit time between two flights in a duty in 30 minutes.
- The maximum sit time between two flights in a duty is 90 minutes.
- Pairings cannot span more than 4 days.
- The minimum rest time between duties is 8 hours.
- The maximum rest time between duties is 18 hours.

1. Instead of thinking about a pairing as a sequence of flights, we can think of it as a sequence of duties. How would you define a network for the pricing problem in this case? What constitutes a node in this network? An arc? Demonstrate how a path in this network corresponds to a pairing and how the cost of this path corresponds to the reduced cost of the pairing.
2. In order to use this network to find the most negative reduced cost pairing, we must solve a multi-label shortest path problem. Define the elements included in a label.
3. Outline an algorithm for solving this multi-label shortest path problem. Pay particular attention to describing when one label dominates another.

4. Compare these two approaches (i.e. flight-based versus duty-based pairing generation). Which network would you expect to have more nodes? Would you expect more or fewer labels kept at each node, on average? Which method do you think would work better in a domestic network? An international network?
5. Consider the special case where pairings are not allowed to have more than two duties. How would you modify the duty-based network to take this into account? Do you still need to use a multi-label shortest path algorithm, or can you solve a basic shortest path problem?

## **Problem 2: An Alternative Way to Integrate Crew Pairing and Maintenance Routing**

In the extended crew pairing model (*ECP*) we start with a restricted master problem that guarantees feasibility, and then iteratively add maintenance routing *columns* to improve solution quality. In this exercise, we consider an alternative approach, where we instead start with an unconstrained version of the crew pairing problem and add maintenance routing *constraints* to enforce feasibility.

1. In the first step of this approach we solve an *unconstrained crew pairing problem* – that is, we solve the crew pairing problem in which *all* forced turns are included. How can you determine if this solution is maintenance feasible?
2. If the solution is maintenance feasible, is it optimal for the integrated maintenance routing/crew pairing problem? Why or why not?
3. Suppose the crew pairing solution is maintenance-infeasible. Describe a constraint that you can add to the crew pairing problem that will eliminate the infeasible set of pairings.
4. Suppose that you add this constraint to the unconstrained crew pairing problem and then re-solve. You now need to again check for maintenance feasibility. It is possible that you will be faced with the exact same feasibility problem. Explain why this is so. How could you construct an alternative constraint, based not on pairings but on forced turns, which would ensure that each maintenance feasibility problem would be unique?
5. In each iteration of *ECP*, we add a column corresponding to a maintenance-feasible forced turn set. We demonstrated how the algorithm could be made more efficient by only adding columns that corresponded to *maximally independent* forced turn sets – that is, forced turn sets that are maintenance feasible but for which increasing the set by adding any further forced turns yields infeasibility. Can you explain what it would mean to similarly restrict the constraints in our new approach to be *minimally infeasible*? How would this improve the performance of

the algorithm? Describe (in words) the problem of identifying such a set. [You do not have to formulate this problem mathematically.]

6. Consider this approach of alternating between solving the unconstrained crew pairing problem and adding constraints based on minimally infeasible forced turns sets. Is this algorithm guaranteed to terminate in a finite number of steps? Why or why not? If it terminates, are you guaranteed an optimal solution to the integrated crew pairing/maintenance routing problem?