The state is defined as a class. It has the attribute tod, where, for every plane, we will have the time of day that plane is at. As for the attribute schedule, it is a list of lists containing dictionaries, where, for every plane, we will have the legs (each a dictionary) assigned to it, in order. Lastly, the attribute remaining is a list of dictionaries, containg the legs that have yet to be added.

The operators are just adding a new leg to the schedule of any given airplane.

When we jump from one state, N, to another, P, the cost function grows 1/profit\_c\_i(L->P) and our heuristic decreases 1/max(profit\_c\_i(L->P)). As such and looking at the inequality h(N) <= c(N,P) + h(P), we can say that h(P) = h(N) – 1/max(profit\_c\_i(L->P)).

Thus: 0 <= c(N,P) – 1/max(profit\_c\_i(P)). Since c(N,P) is 1/profit\_c\_i(L->P), this equality is verified and our heuristic is consistent. As such, A\* returns the optimal solution.

Let C = bound. From one state, N, to the next, P, g(n) grows C – profit\_c\_i(N->P) and h(n) decreases C – max(profit\_c\_i(N->P)). That is, c(N,P) = C – profit\_c\_i(N->P) and h(P) = h(N) – (C - max(profit\_c\_i(N->P))). These equations will be replaced in the consistency inequation.

Replacing in h(N) <= c(N,P) + h(P), we get: 0 <= –profit\_c\_i(N->P) + max(profit\_c\_i(N->P). Thus, the equality is verified and our heuristic is consistent. As such, A\* returns the optimal solution.

//Replacing h(P) – h(N) in h(N) <= c(N,P) + h(P), we get 0 <= c(N,P) – (C – max(profit\_c\_i(N->P))). Replacing c(N,P), this simplifies to: 0 <= –profit\_c\_i(N->P) + max(profit\_c\_i(N->P). Thus, the equality is verified and our heuristic is consistent. As such, A\* returns the optimal solution.

The cost function is defined as the inverse of the profit, that is, for every leg added to a plane of a given class, i, the cost will increase by 1/profit\_c\_i(leg).  
The heuristic is defined as the sum of the 1/max(profit\_c\_i(leg)) for every leg that has yet to be added to the airplanes’ schedules.

The cost function of one action is defined as the maximum of the maximum profits of each leg + 1 (constant defined as "bound") minus the profit of the added leg. That is, if a leg was added to an airplane of a class i, the cost will increase by: bound – profit\_c\_i(leg).

The heuristic is defined as the summation of the value bound – max(profit\_c\_i(leg)), for every leg that has yet to be added to the airplanes’ schedules.