OPERATOR loadcargo c1 p1 sfo
IF
  AT c1 sfo
  AT p1 sfo
ADD
  IN c1 p1
DELETE
  AT c1 sfo

OPERATOR unload c1 p1 jfk
IF
  IN c1 p1
  AT p1 jfk
ADD
  AT c1 jfk
DELETE
  IN c1 p1

OPERATOR fly p1 sfo jfk
IF
  AT p1 sfo
ADD
  AT p1 jfk
DELETE
  AT p1 sfo

INITIAL
AT c1 sfo
AT c2 jfk
AT p1 sfo
AT p2 jfk

OPERATOR loadcargo c2 p1 jfk
IF
  AT c2 jfk
  AT p1 jfk
ADD
  IN c2 p1
DELETE
  AT c2 jfk

OPERATOR unload c2 p1 sfo
IF
  IN c2 p1
  AT p1 sfo
ADD
  AT c2 sfo
DELETE
  IN c2 p1

GOALS
AT c1 jfk
AT c2 sfo

OPERATOR fly p1 jfk sfo
IF
  AT p1 jfk
ADD
  AT p1 sfo
DELETE
  AT p1 jfk

Green arrows indicate establishes
Red arrows indicate threatens
Blue arrows indicate before

This plan is obtained by performing backtracking a little differently than was described on the course website. In this case, if a loop of before arrows is found, the algorithm examines other ways of satisfying the precondition that is problematic. For example, in the operator "unload c2 p1 sfo", the precondition "AT p1 sfo" could not be satisfied by the initial conditions because it created a loop of before arrows. Rather than trying a different instantiation of the variables in the new operator, it first tries to satisfy the precondition "AT p1 sfo" in some other way. In this case, it finds the operator "fly p1 jfk sfo" to satisfy the precondition.