Homework 2.
Assigned [Sep 28, 2004]
Due [October 11th, 2004]

Qn I. Consider the planning problem from the first question of the first homework (reproduced below for your convenience)

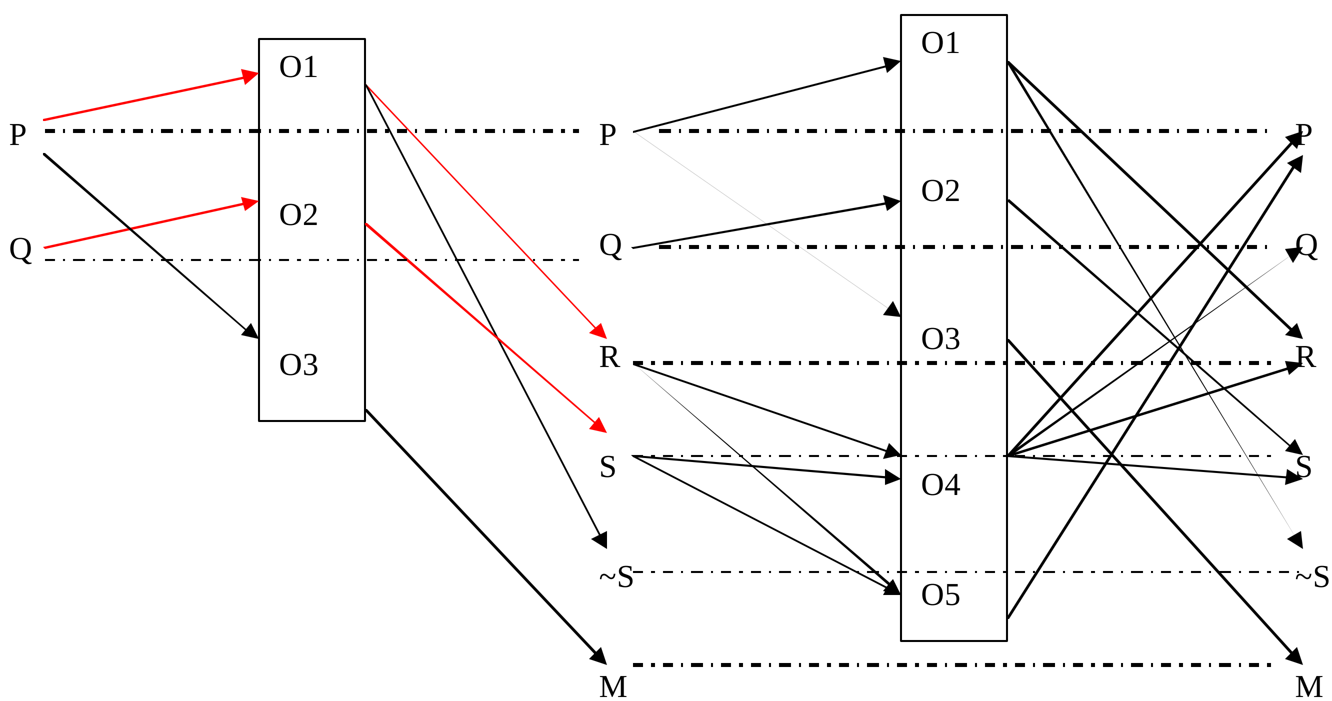
----

operator O1 prec: P Eff: R, ~S	operator O2 prec: Q Eff: S	operator O3 prec: P Eff: M	operator O4 prec: R,S Eff: P,Q,R,S	operator O5 prec: R,S Eff: P

The **initial state** is **{P,Q}** and the desired **goals** are **{P,Q,R,S}**

I.A Draw the "relaxed planning graph" for this problem (relaxed planning graph ignores negative interactions--ie, no mutexes).

I.A.1 Answer: Planning Graph Without Mutexes.



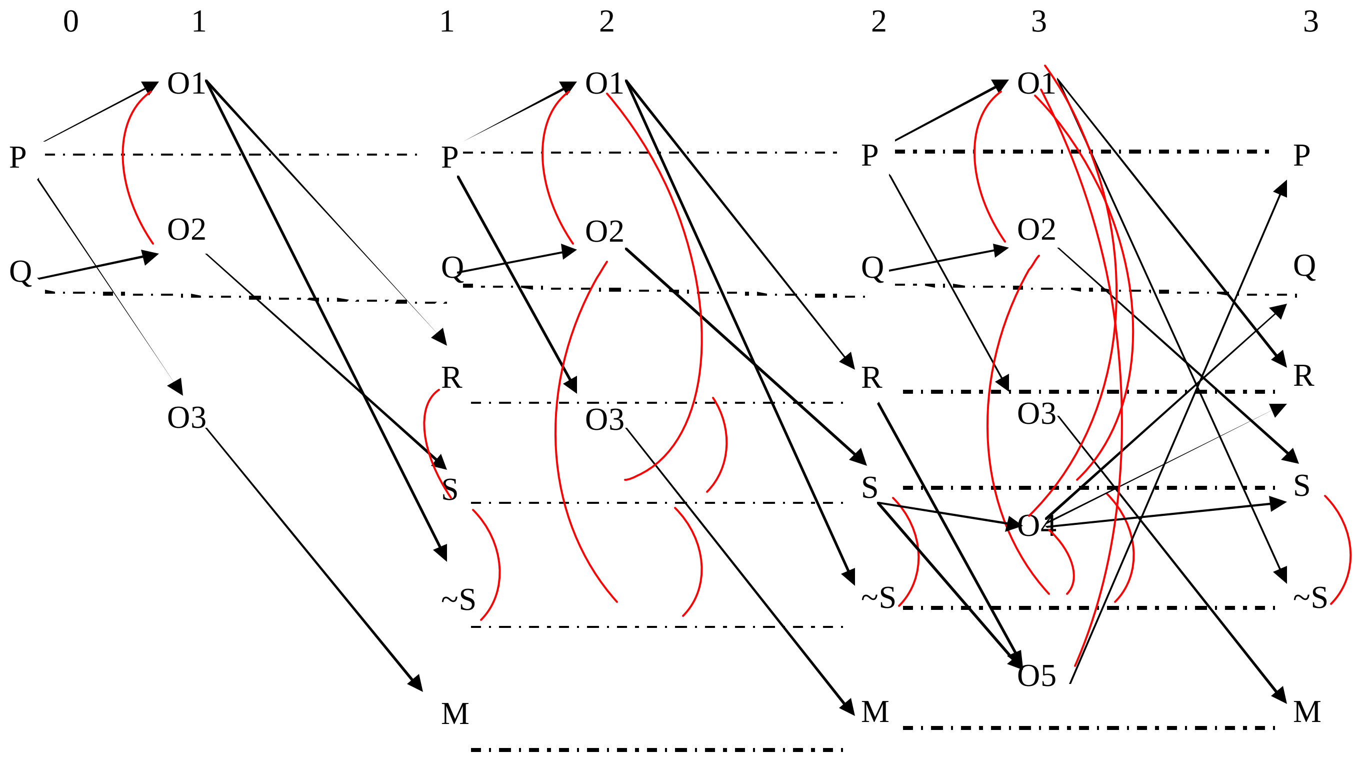
Mark a relaxed plan that supports the top level goals in this relaxed planning graph.

I.A.2 Answer: Relaxed plan is: **P = {O1, O2}**

I.A.3. What is the heuristic value of the goal set {P,Q,R,S} in terms of:
i. Sum heuristic ii. Level heuristic iii. relaxed plan heuristic

STATE	SUM	SET LEVEL	RELAXED PLAN
P,Q,R,S	2	1	2

I.B. Now draw the standard mutex graph (as described in the text and used by planning graph--don't need to use serial graph).



With respect to this standard graph, what is the heuristic cost of goal set{P,Q,R,S} using SUM and Level heuristics? What is the value of the adjusted sum heuristic (recall that it is equal to relaxed plan length + -ve interaction penalty).

STATE	SUM	SET LEVEL	ADJSUM
P,Q,R,S	2	2	3

ADjsum = 2 (relaxplan) + 1 (2 non-mutex level – 1 first level in the graph) = 3

Qn II

Consider the following problem. There are two actions: A1 and A2

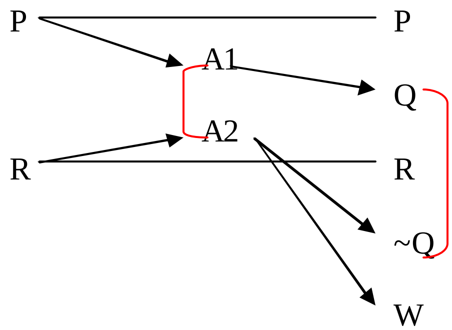
A1: prec: p eff: q

A2 prec: r eff: ~q,w

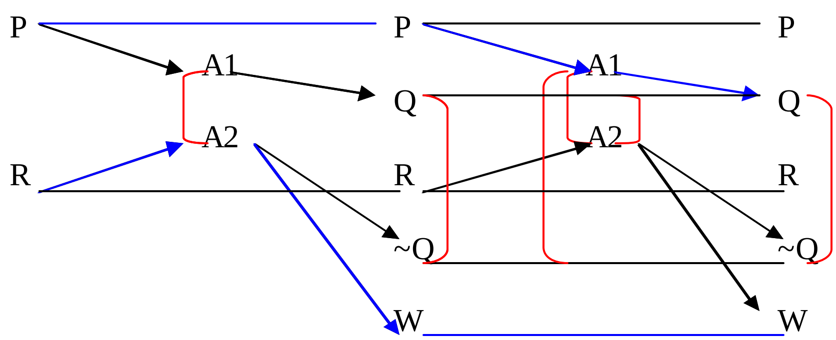
We start with init state where p and r are true.

**and our goals are q and w.

II.a. Show how graphplan solves this problem--assuming that only static interference relations are marked. No mutex propagation is done. Show all the steps in the graph construction, search and memo finding. This is a really small problem.

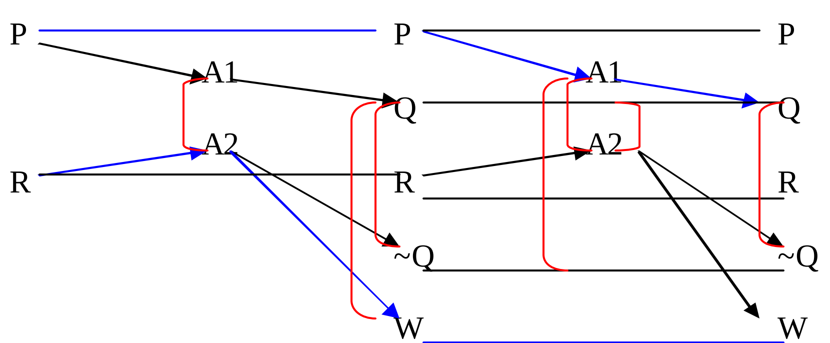


Since only static interference is considered we stop at level 1 of the graph construction phase, and start searching for a solution. Given that Q and W are not mutexes. So, we can support Q with A1, and W with A2. At this time we stop since A1 and A2 are mutex with each other. Given that there are no more choices for the subgoals, we write a memo at level 1 {Q,W: 1} giving the explanation for the failure.



At level 2, we can search again for a solution. This time, we can choose to support Q with A1, and W with its persistent action. We have to satisfy then the preconditions of such actions, subgoals P and W. P can only be supported by its persistent action, and W by A2. This time we have regressed to the initial state, and we have found a solution to our problem: **A2-A1.**

II.b. Now do this problem assuming that mutex propagation using the normal rules of Graphplan is done: With normal mutex propagation, we have to build our planning graph up to level 2, without search because even tough our goals are present at level 1, they are also mutex to each other.



The search is conducted in a similar way to that one of II.a , finding the same solution. **A2-A1**.

II.c. Comment on the relation between memoization and mutex propagation as evidenced by parts a and b.

C). Memos can be considered as finding mutexes during search, some of them could be higher order ones (e.g. more than 2 relations). In order to find a Memo, we need to fail first, and find the explanation for such failure. So, memos are built on a demand driven basis while mutexes are computed globally during the propagation of the planning graph. If mutex propagation gets relaxed as in II.a, more memos will be found since we may fail more times (we are not seeing the mutexes up front), such memos will be then used as kind of mutexes to prune inconsistent states later during search.

Qn III

For this problem, you will use the planning graph in the last level II.b above.

III.a. Convert the planning graph into a CSP encoding (the problem is small enough that you can write the entire encoding down). Show a solution for this CSP encoding, and show how it corresponds to a plan.

CSP Encoding

Variables and Domains:

P-2 : {#, NooP-P2}, Q-2:{#,NooP-Q2,A1-2}, R-2:{#,Noop-R2}, ~Q-2:{#,Noop-~Q2,A2-2}, W-2:{#,NooP-W2,A2-2}

P-1: {#, NooP-P1}, Q-1:{#,A1-1}, R-1:{#,Noop-R1}, ~Q-1:{#,A2-1}, W-1:{#,A2-1}

P-0:{#,T}, R-0:{#,T}

Goals: Q-2!=#, W-2!=#

Activation Constraints:

P-2=NooP-P2 => P-1!=#, Q-2=NooP-Q2 => Q-1!=#, Q-2=A1-2 => P-1!=#
R-2=NooP-R2 => R-1!=#, ~Q-2=NooP-~Q2 => ~Q-1 !=#, ~Q-2=A2-2 => R-1!=#
W-2=NooP-W2 => W-1 !=#, W-2=A2-2 => R-1!=#

P-1=NooP-P1 => P-0!=#, Q-1=A1-1 => P-0!=# R-1=NooP-R1 => R-0!=#,
~Q-1=A2-1 => R-0!=#, W-2=A2-1=> R-0!=#

Mutex Constraints:

Q-2=A1-2 => ~Q-2!=A2-2, Q-2=A1-2 => W-2!=A2-2,
Q-2=A1-2 => ~Q-2!=NooP-~Q2, Q-2=Noop-Q2 => ~Q-2!=A2-2,
Q-2=Noop-Q2 => W-2!=A2-2, Q-2=Noop-Q2 => ~Q-2!=NooP-~Q2,
~Q-2=A2-2 => Q-2!=A1-2, ~Q-2=A2-2 => Q-2!=NooP-Q2,
~Q-2=NooP-~Q2 => Q-2!=A1-2, ~Q-2=NooP-~Q2 => Q-2!=NooP-Q2
W-2=A2-2 => Q-2!=A1-2, W-2=A2-2 => Q-2!=NooP-Q2,

Q-1=A1-1 => ~Q-1!=A2-1, Q-1=A1-1 => W-1!=A2-1,

~Q-1=A2-1 => Q-1!=A1-1,   W-1=A2-1 => Q-1!=A1-1,

Solution:

Q-2=A1-2,	P-2=NooP-P2,	R-2=NooP-R2,	~Q-2=#,	W-2=#
Q-1=#,	P-1=NooP-P1,	R-1=NooP-R1,	~Q-1=A2-1	W-1=A2-1
P-0=T,	R-O=T			

III.b. `Do it with SAT encoding of the planing graph.`

SAT Encoding

Initial State: P-0 & R-0

Goal State: Q-2 & W-2

Graph Propagation: (cond x at k => one of its supporting actions)

P-2 => NooP-P2   Q-2 => NooP-Q2 || A1-2   ~Q-2 => NooP-~Q2 || A2-2

R-2 => NooP-R2   W-2 => NooP-W2 || A2-2

P-1 => NooP-P1   Q-1 => A1-1   ~Q-1 => A2-1

R-1 => NooP-R1   W-1=>A2-1

Actions => Preconditions

A1-2 => P-1, A2-2=> R-1, NooP-P2 => P-1,   NooP-Q2=>Q-1,

NooP-~Q2=>~Q-1, NooP-R2=>R-1,   NooP-W2=>W-1,

A1-1 => P-0, A2-1=> R-0, NooP-P1 => P-0,   NooP-R1=>R-0,

Mutexes:

~Q-2 || ~(~Q-2),   ~A1-2 || ~A2-2,   ~A1-2 || ~NooP-~Q2

~A2-2 || ~NooP-Q2, ~Q-1 || ~(~Q-1),   ~Q-1 || ~W-1,   ~A1-1 || ~A2-1

Solution: A2-1 & NooP-P1 & NooP-W2 & A1-2

III.c. `(Added since I extended the deadline to 10/11 ;-) Do an "explanatory axiom" (or backward proof based) encoding of this problem (for the same length as the planning graph you used in the previous parts). Mark which constraints are similar, different, stronger etc. compared to III.b.`

Propositions corresponding to the initial conditions and goals are true at their respective levels (similar, in fact same):

P_0 & R_0 & Q_2 & W_2

At least one of the actions at each level will occur (stronger, although hard to tell exactly, hence it would be better to say that they could possibly be stronger):

$A1\_1 \vee A2\_1$

$A1\_2 \vee A2\_2$

Actions imply their preconditions and effects (somewhat different, now actions also imply their effects in addition to only their preconditions):

$A1\_1 \Rightarrow P\_0 \ \& \ Q\_1$

$A2\_1 \Rightarrow R\_0 \ \& \ {\sim}Q\_1 \ \& \ W\_1$

$A1\_2 \Rightarrow P\_1 \ \& \ Q\_2$

$A2\_2 \Rightarrow R\_1 \ \& \ {\sim}Q\_2 \ \& \ W\_2$

A proposition P changes values between j and j+1 only if an action occurs that makes it so (different)
$\sim Q\_0 \ \& \ Q\_1 \Rightarrow A1\_1$
$Q\_0 \ \& \sim Q\_1 \Rightarrow A2\_1$
$\sim W\_0 \ \& \ W\_1 \Rightarrow A2\_1$
$\sim Q\_1 \ \& \ Q\_2 \Rightarrow A1\_2$
$Q\_1 \ \& \sim Q\_2 \Rightarrow A2\_2$
$\sim W\_1 \ \& \ W\_2 \Rightarrow A2\_2$

No pair of interacting actions must occur together (somewhat different, in general there could be more mutexes than interactions, recall that interactions are static and mutexes may also be dynamic)
$\sim A1\_1 \vee \sim A2\_1$
$\sim A1\_2 \vee \sim A2\_2$