

COMP9414: Artificial Intelligence

Solutions 3: Constraint Satisfaction/Planning

1. Let Q_i be variables whose domain is $\{1, \dots, 8\}$. The value assigned to Q_i is the row of the Queen in column i .

Backtracking search starts by assigning 1 to Q_1 (assuming variables and values are chosen in order). Forward checking removes 1 and i from the domains of each other Q_i . Then Q_2 is assigned 3, and forward checking removes 3 from the domains of Q_3, \dots, Q_8 , 2 and 4 from Q_3 , 5 from Q_4 , 6 from Q_5 , 7 from Q_6 and 8 from Q_7 . Then Q_3 is assigned 5, etc.

Domain splitting is similar. At the first step, the domain of Q_1 is split into $\{1, 2, 3, 4\}$ and $\{5, 6, 7, 8\}$. These are both arc consistent. The first domain is split again into $\{1, 2\}$ and $\{3, 4\}$. Again these subproblems are arc consistent. The first domain is split again, into $\{1\}$ and $\{2\}$, and now arc consistency proceeds much as forward checking. However, this depends on the order of variable and problem selection.

2. $stack(A, B)$: precondition $clear(A), clear(B)$; add list $on(A, B)$; delete list $clear(B)$
 $unstack(A)$: precondition $clear(A), on(A, B)$; add list $on(A, Table), clear(B)$; delete list $on(A, B)$
3. The nonlinear planner introduces the two actions $stack(B, C)$ and $stack(A, B)$. The $clear(A)$ precondition of $stack(A, B)$ does not hold in the initial state, so $unstack(C)$ is added to the plan. Because $stack(A, B)$ deletes $clear(B)$, which is a precondition of $stack(B, C)$, $stack(B, C)$ must be before $stack(A, B)$. For the same reason, $unstack(C)$ must be before $stack(B, C)$. The plan is therefore $unstack(C), stack(B, C), stack(A, B)$.