

COMP9414: 人工智能解决方案3: 约束满足/规划

1. 让 Q_i 是变量，其域为 $\{1, \dots, 8\}$ 。分配给 Q_i 的值是第 i 列的女王行。
回溯搜索开始时，将1分配给 Q_1 （假设变量和值是按顺序选择的）。前向检查将1和 i 从其他 Q_i 的域中删除。然后 Q_2 被分配为3，向前检查将3从 Q_3, \dots, Q_8 的域中删除，2和4从 Q_3 ，5从 Q_4 ，6从 Q_5 ，7从 Q_6 ，8从 Q_7 。然后 Q_3 被分配为5，等等。
域的拆分也是类似的。在第一步， Q_1 的域被分割成 $\{1, 2, 3, 4\}$ 和 $\{5, 6, 7, 8\}$ 。这些都是弧形一致的。第一个域再次被分割成 $\{1, 2\}$ 和 $\{3, 4\}$ 。同样，这些子问题是弧形一致的。第一个域再次被分割成 $\{1\}$ 和 $\{2\}$ ，现在弧形一致性的进行与正向检查差不多。然而，这取决于变量和问题选择的顺序。
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. 非线性规划器引入了两个动作 $stack(B, C)$ 和 $stack(A, B)$ 。 $stack(A, B)$ 的 $clear(A)$ 前提条件在初始状态下不成立，所以 $unstack(C)$ 被添加到计划中。因为 $stack(A, B)$ 删除了 $clear(B)$ ，而 $clear(B)$ 是 $stack(B, C)$ 的一个前提条件，所以 $stack(B, C)$ 必须在 $stack(A, B)$ 之前。出于同样的原因， $unstack(C)$ 必须在 $stack(B, C)$ 之前。因此计划是 $unstack(C), stack(B, C), stack(A, B)$ 。