

DISTILL with Loop Identification

Elly Zoe Winner and Manuela Veloso

Input: Minimal annotated consistent partial order $\mathcal{P}$,
current template T_i .

Output: New template T_{i+1} , updated with $\mathcal{P}$

```

procedure DISTILL ( $\mathcal{P}$ ,  $T_i$ ):
   $\mathcal{A} \leftarrow \text{Find\_Variable\_Assignment}(\mathcal{P}, T_i.\text{variables}, \emptyset)$ 
  until match or can't match do
    if  $\mathcal{A} = \emptyset$  then
      can't match
    else
       $\mathcal{N} \leftarrow \text{Make\_New\_If\_Statement}(\text{Assign}(\mathcal{P}, \mathcal{A}))$ 
      match  $\leftarrow \text{Is\_A\_Match}(\mathcal{N}, T_i)$ 
    if not can't match and not match then
       $\mathcal{A} \leftarrow \text{Find\_Variable\_Assignment}(\mathcal{P}, T_i.\text{variables}, \mathcal{A})$ 
  if can't match then
     $\mathcal{A} \leftarrow \text{Find\_Variable\_Assignment}(\mathcal{P}, T_i.\text{variables}, \emptyset)$ 
     $\mathcal{N} \leftarrow \text{Make\_New\_If\_Statement}(\text{Assign}(\mathcal{P}, \mathcal{A}))$ 
   $T_{i+1} \leftarrow \text{Add\_To\_Template}(\mathcal{N}, T_i)$ 

procedure Make\_New\_If\_Statement( $\mathcal{P}_A$ ):
   $N \leftarrow$  empty if statement
  for all terms  $t_m$  in initial state of  $\mathcal{P}_A$  do
    if exists a step  $s_n$  in plan body of  $\mathcal{P}_A$  such that
       $s_n$  needs  $t_m$  or goal state of  $\mathcal{P}_A$  needs  $t_m$  then
      Add\_To\_Conditions( $N$ , in\_current\_state ( $t_m$ ))
  for all terms  $t_m$  in goal state of  $\mathcal{P}_A$  do
    if exists a step  $s_n$  in plan body of  $\mathcal{P}_A$  such that
       $t_m$  relies on  $s_n$  then
      Add\_To\_Conditions( $N$ , in\_goal\_state ( $t_m$ ))
  for all steps  $s_n$  in plan body of  $\mathcal{P}_A$  do
    Add\_To\_Body( $N$ ,  $s_n$ )
  return  $N$ 

procedure Is\_A\_Match( $\mathcal{N}$ ,  $T_i$ ):
  for all if-statements  $I_n$  in  $T_i$  do
    if  $\mathcal{N}$  matches of  $I_n$  then
      return true

procedure Add\_To\_Template( $\mathcal{N}$ ,  $T_i$ ):
  for all if-statements  $I_n$  in  $T_i$  do
    if  $\mathcal{N}$  matches  $I_n$  then
       $I_n \leftarrow \text{Combine}(I_n, \mathcal{N})$ 
    return
  if  $\mathcal{N}$  is unmatched then
    Add\_To\_End( $\mathcal{N}$ ,  $T_i$ )

```

Table 1: The DISTILL algorithm: updating a template with a new observed plan.