

```
function PDR(model  $M$ ,  $p \in AP$ )
  if  $S_0 \wedge \neg p$  is SAT return “ $M \not\models \text{AG } p$ ”;
   $F_0 := S_0$ ;  $k := 0$ ;
  while true do
    extendFrontier( $M, k$ )
    propagateClauses( $M, k$ )
    if  $F_i = F_{i+1}$  for some  $i$  then return “ $M \models \text{AG } p$ ”;
     $k := k + 1$ 
  end while
end function
```

**procedure** *extendFrontier*(model  $M$ ,  $k : \mathbb{N}$ )

$F_{k+1} := \{s \mid p \in L(s)\};$

**while**  $F_k \wedge R \wedge \neg p'$  is SAT **do**

$s' :=$  state labeled with  $\neg p$  extracted from satisfying assignment

$s :=$  predecessor of  $s'$  extracted from satisfying assignment

*removeCTI*( $M, s, k$ )

**end while**

**end procedure**

```
procedure propagateClauses( $k : \mathbb{N}$ )
  for  $i$  from 1 to  $k$ 
    for every clause  $c \in F_i$ 
      if  $F_i \wedge R \wedge \neg c'$  is UNSAT then
         $F_{i+1} := F_{i+1} \wedge c$ 
      end if
    end for
  end for
end procedure
```

**Figure 10.11**

Propagation of clauses into other frames.

```
procedure removeCTI(model  $M$ ,  $s : S$ ,  $i : \mathbb{N}$ )
if  $S_0 \wedge s$  is SAT then abort " $M \not\models \text{AG } p$ ";  

while  $F_i \wedge R \wedge \neg s \wedge s'$  is SAT do
    for  $l$  from 0 to  $i$ 
         $F_l := F_l \wedge \neg s$ 
    end for
     $t :=$  predecessor of  $s$ , extracted from SAT witness
     $\text{removeCTI}(M, t, i - 1)$ 
end while
end procedure
```