

```

procedure forward_checking_treesearch (U, L, f, FTAB, T, R);
  u := first(U);
  for each l ∈ L do
    if FTAB(u, l) == 1 then

      begin
        f' := f ∪ {(u, l)};
        U' := remainder(U);
        if isempty(U')
          then output(f')
        else
          begin
            NEWFTAB := copy(FTAB);
            OK := forward_check(NEWFTAB, u, l, U', L, T, R, f');
            if OK then forward_checking_treesearch (U', L, f', NEWFTAB, T, R)
          end
        end
      end
    end for
  end forward_checking_treesearch;

```

```

function forward_check(FTAB, u, l, future_units, L, T, R, f');
  for each u' ∈ future_units do
    forward_check := false;
    for each l' ∈ L with FTAB(u', l') == 1 do
      if compatible(u, l, u', l', T, R, f')
        then forward_check := true
      else FTAB(u', l') := 0
    end for
    if forward_check == false then break
  end for
  end forward_check;

```