```
init(a) := 1;
init(b) := 0;
next(a) := case
             !a : 0;
             b  : 1;
             1  : {0,1};
           esac;
next(b) := case
             a & next(a) : !b;
             !a : 1;
             1  : {0,1};
           esac;
```

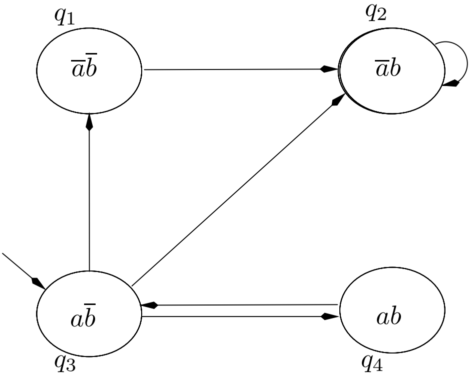

Figure 3.33. An SMV program and its model $\mathcal{M}$.

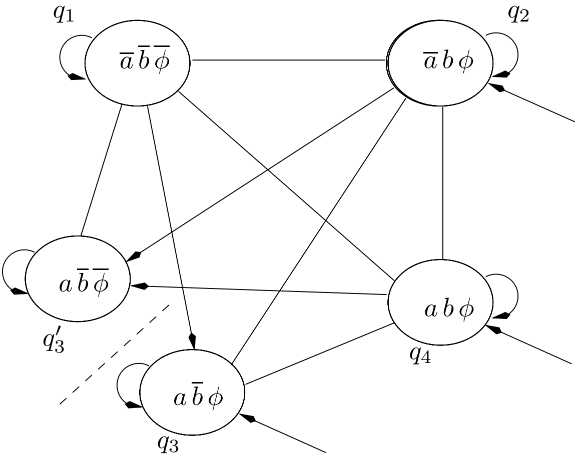

Figure 3.34. Automaton accepting precisely traces satisfying $\phi \overset{\mathrm{def}}{=} a \mathrm{\ U\ } b$. The transitions with no arrows can be taken in either direction. The acceptance condition is that the path of the automaton cannot loop indefinitely through q_3.

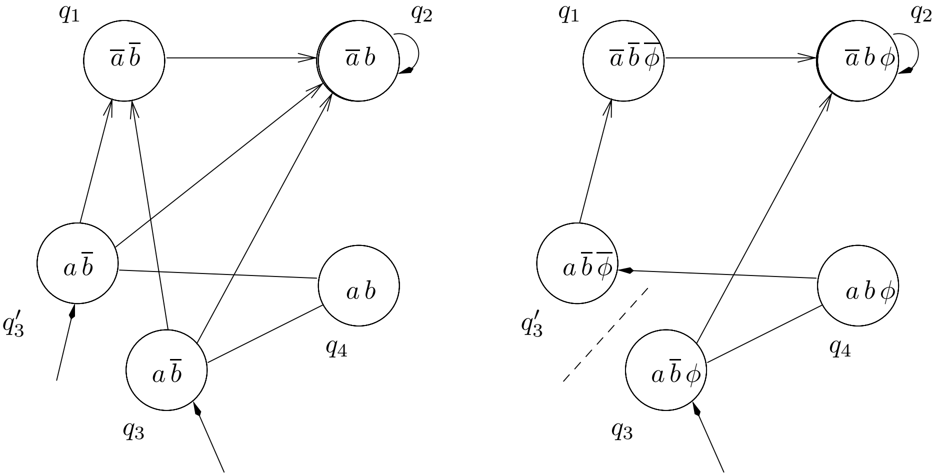

Figure 3.35. Left: the system $\mathcal{M}$ of Figure 3.33, redrawn with an expanded state space; right: the expanded $\mathcal{M}$ and $A_{a\mathrm{U}b}$ combined.

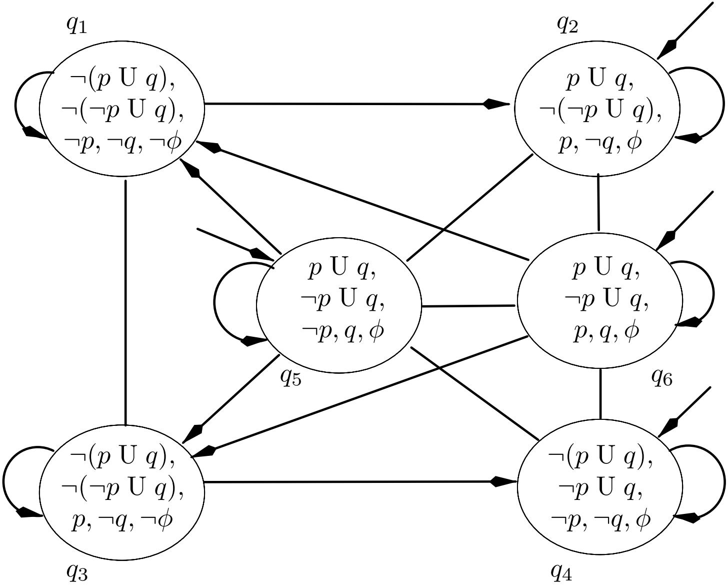
q_1
$\neg(p \text{ U } q)$, $\neg(\neg p \text{ U } q)$, $\neg p, \neg q, \neg\phi$
q_2
$p \text{ U } q$, $\neg(\neg p \text{ U } q)$, $p, \neg q, \phi$
$p \text{ U } q$, $\neg p \text{ U } q$, $\neg p, q, \phi$
q_5
$p \text{ U } q$, $\neg p \text{ U } q$, p, q, ϕ
q_6
$\neg(p \text{ U } q)$, $\neg(\neg p \text{ U } q)$, $p, \neg q, \neg\phi$
q_3
$\neg(p \text{ U } q)$, $\neg p \text{ U } q$, $\neg p, \neg q, \phi$
q_4

Figure 3.36. Automaton accepting precisely traces satisfying $\phi \stackrel{\text{def}}{=} (p \text{ U } q) \vee (\neg p \text{ U } q)$. The transitions with no arrows can be taken in either direction. The acceptance condition asserts that every run must pass infinitely often through the set $\{q_1, q_3, q_4, q_5, q_6\}$, and also the set $\{q_1, q_2, q_3, q_5, q_6\}$.