

# A More Rigorous Proof of Deadlock Freedom

**Theorem**  $Spec \Rightarrow DeadlockFree$

1.  $Spec \Rightarrow \Box LInv$

PROOF: This is a standard invariance proof, which is omitted.

2. SUFFICES ASSUME:  $\Box LInv \wedge \Box [Next]_{vars} \wedge Fairness$

PROVE:  $DeadlockFree$

PROOF: By 1 and the definition of  $Spec$ .

3. SUFFICES ASSUME:  $\Box \neg Success$

PROVE:  $(T0 \vee T1) \leadsto FALSE$

PROOF: This is a standard temporal proof by contradiction, since  $DeadlockFree$  equals  $(T0 \vee T1) \leadsto Success$ .

4.  $T0 \leadsto FALSE$

4.1.  $T0 \leadsto \Box(pc[0] = \text{"e2"})$

PROOF: Assumption  $\Box LInv$  implies process 0 is never at  $e3$  or  $e4$ . Therefore, by the code and assumption  $Fairness$ , we see that if  $T0$  is true and process 0 never reaches  $cs$  (which is implied by the assumption  $\Box \neg Success$ ), then process 0 eventually reaches  $e2$  and stays there forever.

"The code" is shorthand for "the step 2 assumptions  $\Box [Next]_{vars}$  and  $\Box LInv$ ".

4.2.  $\Box(pc[0] = \text{"e2"}) \leadsto \Box((pc[0] = \text{"e2"}) \wedge \neg x[1])$ .

4.2.1. SUFFICES ASSUME:  $\Box(pc[0] = \text{"e2"})$

PROVE:  $TRUE \leadsto \Box \neg x[1]$

PROOF: By the  $\Box \leadsto$  Rule.

4.2.2.  $TRUE \leadsto (\Box(pc[1] = \text{"ncs"}) \vee \Box T1)$ .

PROOF: The code and assumption  $Fairness$  imply that if process 1 never reaches  $cs$  (by the assumption  $\Box \neg Success$ ), then eventually it must either reach and remain forever at  $ncs$ , or  $T1$  must become true and remain true forever.

4.2.3.  $\Box(pc[1] = \text{"ncs"}) \Rightarrow \Box \neg x[1]$ .

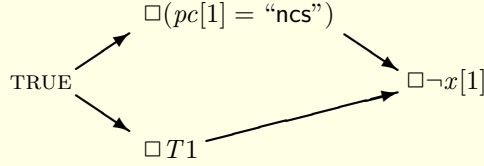
PROOF:  $\Box LInv$  implies  $x[1]$  equals FALSE when process 1 is at  $ncs$ .

4.2.4.  $\Box T1 \leadsto \Box \neg x[1]$

PROOF:  $(pc[0] = \text{"e2"})$  implies  $x[0]$ , so the step 4.2.1 assumption implies  $\Box x[0]$ . The code,  $Fairness$ ,  $\Box \neg Success$ , and  $\Box x[0]$  imply that  $T1$  leads to process 1 reaching and remaining forever at  $e4$  with  $x[1]$  equal to FALSE.

4.2.5. Q.E.D.

PROOF: By 4.2.1–4.2.4 and Leads-To Induction, with this proof graph:



4.3.  $\Box((pc[0] = \text{"e2"}) \wedge \neg x[1]) \rightsquigarrow \text{FALSE}$

PROOF: The code and *Fairness* imply that  $(pc[0] = \text{"e2"}) \wedge \Box \neg x[1]$  leads to process 0 reaching *cs*, contradicting  $\Box \neg \text{Success}$ .

4.4. Q.E.D.

PROOF: By 4.1–4.3 and Leads-To Induction, with this proof graph:

$$T0 \longrightarrow \Box(pc[0] = \text{"e2"}) \longrightarrow \Box((pc[0] = \text{"e2"}) \wedge \neg x[1]) \longrightarrow \text{FALSE}$$

5.  $T1 \rightsquigarrow \text{FALSE}$

5.1.  $T1 \Rightarrow \Box T1$

PROOF: From the code, we see that if *T1* is true and process 1 never reaches *cs* (which is implied by the assumption  $\Box \neg \text{Success}$ ), then *T1* remains forever true.

5.2.  $\Box T1 \rightsquigarrow (T0 \vee \Box(T1 \wedge \neg T0))$

PROOF: By the tautologies  $F \rightsquigarrow (G \vee (F \wedge \Box \neg G))$  and  $\Box F \wedge \Box G \equiv \Box(F \wedge G)$ .

5.3.  $\Box(T1 \wedge \neg T0) \rightsquigarrow \Box(T1 \wedge \neg x[0])$

PROOF: By the code and *Fairness*,  $\Box \neg T0$  implies that eventually process 0 is always at *ncs*, which implies that *x*[0] always equals FALSE.

5.4.  $\Box(T1 \wedge \neg x[0]) \rightsquigarrow \text{FALSE}$

PROOF: The code, *Fairness*, and  $\Box \neg x[0]$  imply that process 1 eventually reaches *e2*. Assumption *Fairness* and  $\Box \neg x[0]$  then imply that process 1 reaches *cs*, contradicting the assumption  $\Box \neg \text{Success}$ .

5.5. Q.E.D.

PROOF: By 5.1–5.4, step 4, and Leads-To Induction, with this proof graph:

?

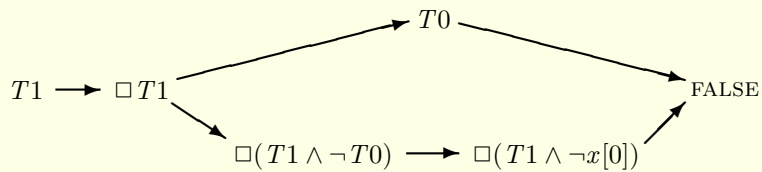
←

→

C

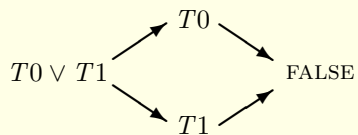
I

S



6. Q.E.D.

PROOF: By steps 3–5 and Leads-To Induction, with this simple proof graph:



?

←

→

C

I

S