

### 10.1.2 Esempio: il problema della ruota di scorta

Considerate il problema di cambiare una ruota bucata (Figura 10.2). L'obiettivo consiste nell'avere una ruota di scorta montata in maniera corretta sull'asse della macchina, mentre lo stato iniziale ha una ruota bucata sull'asse e una ruota di scorta nel bagagliaio. Per semplicità la nostra versione del problema è molto astratta e non prevede bulloni incastrati o altre complicazioni. Le azioni sono solo quattro: prendere la ruota di scorta dal bagagliaio, rimuovere quella bucata dall'asse, montare la ruota di scorta sull'asse e lasciare la macchina incustodita tutta la notte. Presumiamo che la macchina sia parcheggiata in un quartiere particolarmente malfamato, per cui l'effetto di lasciarla incustodita è la sparizione di tutte le ruote. Una soluzione del problema è  $[Rimuovi(Bucata, Asse), Rimuovi(Scorta, Bagagliaio), Monta(Scorta, Asse)]$ .

### 10.1.3 Esempio: il mondo dei blocchi

mondo dei blocchi

Uno dei più famosi domini di pianificazione è il **mondo dei blocchi**. Questo dominio consiste in un insieme di blocchi di forma cubica appoggiati a un tavolo.<sup>2</sup> I blocchi possono essere impilati ma solo un blocco può stare direttamente sopra un altro. Un braccio robotico può prendere un blocco e spostarlo in un'altra posizione, sul tavolo oppure su un altro blocco. Il braccio può tenere un solo blocco alla volta, per cui non può afferrarne uno che ne ha sopra un altro. Lo scopo sarà sempre quello di costruire una o più pile di blocchi, descritte specificando esattamente quali

---

*Init*(*Gomma*(*Bucata*)  $\wedge$  *Gomma*(*Scorta*)  $\wedge$  *Posizione*(*Bucata*, *Asse*)  $\wedge$  *Posizione*(*Scorta*, *Bagagliaio*))  
*Obiettivo*(*Posizione*(*Scorta*, *Asse*))  
*Azione*(*Rimuovi*(*ogg*, *pos*))  
 PRECOND: *Posizione*(*ogg*, *pos*)  
 EFFETTO :  $\neg$ *Posizione*(*ogg*, *pos*)  $\wedge$  *Posizione*(*ogg*, *Terreno*)  
*Azione*(*Monta*(*t*, *Asse*))  
 PRECOND: *Gomma*(*t*)  $\wedge$  *Posizione*(*t*, *Terreno*)  $\wedge$   $\neg$ *Posizione*(*Bucata*, *Asse*)  
 EFFETTO:  $\neg$ *Posizione*(*t*, *Terreno*)  $\wedge$  *Posizione*(*t*, *Asse*)  
*Azione*(*AbbandonaDiNotte*,  
 PRECOND:  
 EFFETTO:  $\neg$ *Posizione*(*Scorta*, *Terreno*)  $\wedge$   $\neg$ *Posizione*(*Scorta*, *Asse*)  
 $\wedge$   $\neg$ *Posizione*(*Scorta*, *Bagagliaio*)  $\wedge$   $\neg$ *Posizione*(*Bucata*, *Terreno*)  
 $\wedge$   $\neg$ *Posizione*(*Bucata*, *Asse*)  $\wedge$   $\neg$ *Posizione*(*Bucata*, *Bagagliaio*)

---

Figura 10.2 Il semplice problema della ruota bucata.

<sup>2</sup> Il mondo dei blocchi utilizzato nella ricerca sulla pianificazione è molto più semplice della versione di SHRDLU, mostrata nella Figura 1.4.

blocchi si trovano sopra a quali altri. Un obiettivo, per esempio, potrebbe essere quello di avere il blocco *A* su quello *B* e il blocco *B* su quello *C* (Figure 10.3 e 10.4).

Useremo *On*(*b*, *x*) per indicare che il blocco *b* si trova su *x*, dove *x* può essere un altro blocco o il tavolo. L'azione di muovere il blocco *b* da sopra *x* a sopra *y* si scriverà *Move*(*b*, *x*, *y*). Ora, una delle precondizioni per muovere *b* è che non ci sia un blocco su di esso. Nella logica del primo ordine scriveremmo  $\neg \exists x \text{ On}(x, b)$  o, alternativamente,  $\forall x \neg \text{On}(x, b)$ . Il linguaggio PDDL non consente quantificatori, perciò introduciamo un predicato *Libero*(*x*), che è vero quando non c'è nulla sopra *x*. La descrizione completa del problema è riportata nella Figura 10.3).

L'azione *Move* sposta il blocco *b* da *x* a *y* se sia *b* che *y* sono liberi. Alla fine della mossa *b* sarà libero, ma *y* no. Un primo tentativo per lo schema *Move* è:

*Azione*(*Move*(*b*, *x*, *y*),  
 PRECOND: *On*(*b*, *x*)  $\wedge$  *Libero*(*b*)  $\wedge$  *Libero*(*y*),  
 EFFETTO: *On*(*b*, *y*)  $\wedge$  *Libero*(*x*)  $\wedge$   $\neg$ *On*(*b*, *x*)  $\wedge$   $\neg$ *Libero*(*y*)).

---

*Init*(*On*(*A*, *Tavolo*)  $\wedge$  *On*(*B*, *Tavolo*)  $\wedge$  *On*(*C*, *A*)  
 $\wedge$  *Blocco*(*A*)  $\wedge$  *Blocco*(*B*)  $\wedge$  *Blocco*(*C*)  
 $\wedge$  *Libero*(*A*)  $\wedge$  *Libero*(*B*)  $\wedge$  *Libero*(*C*))  
*Obiettivo*(*On*(*A*, *B*)  $\wedge$  *On*(*B*, *C*))  
*Azione*(*Move*(*b*, *x*, *y*),  
 PRECOND: *On*(*b*, *x*)  $\wedge$  *Libero*(*b*)  $\wedge$  *Libero*(*y*)  $\wedge$  *Blocco*(*b*)  $\wedge$  *Blocco*(*y*)  $\wedge$  (*b*  $\neq$  *x*)  $\wedge$  (*b*  $\neq$  *y*)  $\wedge$  (*x*  $\neq$  *y*)  
 EFFETTO: *On*(*b*, *y*)  $\wedge$  *Libero*(*x*)  $\wedge$   $\neg$ *On*(*b*, *x*)  $\wedge$   $\neg$ *Libero*(*y*)  
*Azione*(*MoveSulTavolo*(*b*, *x*),  
 PRECOND: *On*(*b*, *x*)  $\wedge$  *Libero*(*b*)  $\wedge$  *Blocco*(*b*)  $\wedge$  (*b*  $\neq$  *x*)  
 EFFETTO: *On*(*b*, *Tavolo*)  $\wedge$  *Libero*(*x*)  $\wedge$   $\neg$ *On*(*b*, *x*)

---

Figura 10.3 Un problema di pianificazione nel mondo dei blocchi: la costruzione di una torre di tre blocchi. Una soluzione è costituita dalla sequenza [*Move*(*B*, *Tavolo*, *C*), *Move*(*A*, *Tavolo*, *B*)].

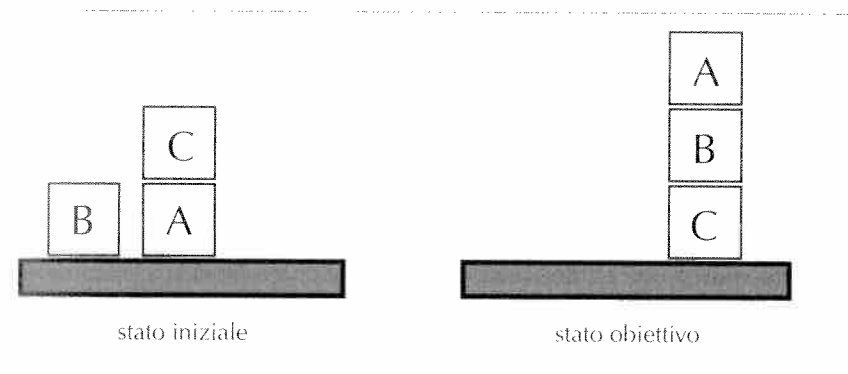


Figura 10.4 Diagramma del problema dei blocchi descritto nella Figura 10.3.