

Assignment 4

Task 1

Adult : adult1 adult2 adult3

Child : child1 child2 child3

Boat : boat

Side : left-side right-side

(: objects adult1 adult2 adult3
 child1 child2 child3
 boat
 left-side right-side)

ADULT(x) true iff x is an adult

CHILD(x) true iff x is a child

BOAT(x) true iff x is a boat

SIDE(x) true iff x is a side of river

(: predicates (ADULT ?a) (CHILD ?c) (BOAT ?b) (SIDE ?s)
(at ?x ?y)
)

(: init (ADULT adult1) (ADULT adult2) (ADULT adult3)
(CHILD child1) (CHILD child2) (CHILD child3)
(BOAT boat)
(SIDE left-side) (SIDE right-side)
(at adult1 (left-side)) (at adult2 (left-side)) (at adult3 (left-side))
(at child1 (left-side)) (at child2 (left-side)) (at child3 (left-side))
(at boat (left-side))
)

(: goal (and (at adult1 right-side) (at adult2 right-side) (at adult3 right-side)
(at child1 right-side) (at child2 right-side) (at child3 right-side)
(at boat right-side))
)

(: action move1a
 : parameters (?adult ?boat ?from ?to)
 : precondition (and (ADULT ?adult) (BOAT ?boat)
 (SIDE ?from) (SIDE ?to))
 (at ?adult ?from) (at ?boat ?from))
 : effect (and (at ?adult ?to) (at ?boat ?to)
 (not (at ?adult ?from)) (not (at ?boat ?from)))

(:action move (a1C
:parameters (?adult ?child ?boat ?from ?to)
:precondition (and (ADULT ?adult) (CHILD ?child) (BOAT ?boat)
(SID₅ ?from) (SID₆ ?to))
(at ?adult ?from) (at ?child ?from) (at ?boat ?from)
:effect (and (at ?adult ?to) (at ?child ?to) (at ?boat ?to)
(not (at ?adult ?from) (not (at ?child ?from)
(not (at ?boat ?from)))

(:action move 1C
:parameters (?child ?boat ?from ?to)
:precondition (and (CHILD ?child) (BOAT ?boat)
(SID₅ ?from) (SID₆ ?to))
(at ?child ?from) (at ?boat ?from)
:effect (and (at ?child ?to) (at ?boat ?to)
(not (at ?child ?from)) (not (at ?boat ?from)))

(:action move 2C
:parameters (?child1 ?child2 ?boat ?from ?to)
:precondition (and (CHILD ?child1) (CHILD ?child2) (BOAT ?boat)
(SID₅ ?from) (SID₆ ?to))
(at ?child1 ?from) (at ?child2 ?from) (at ?boat ?from)
:effect (and (at ?child1 ?to) (at ?child2 ?to) (at ?boat ?to)
(not (at ?child1 ?from) (not (at ?child2 ?from)
(not (at ?boat ?from)))

a complete plan

move 1a1c (adult1 child1 left-side right-side) \rightarrow 2a2c : 1a1c

move 1c (child1 right-side left-side) \rightarrow 2a3c : 1a0c

move 1a1c (adult2 child2 left-side right-side) \rightarrow 1a2c : 2a1c

move 1c (child2 right-side left-side) \rightarrow 1a3c : 2a0c

move 1a1c (adult3 child3 left-side right-side) \rightarrow 0a2c : 3a1c

move 1c (child3 right-side left-side) \rightarrow 0a3c : 3a0c

move 2c (child1 child2 left-side right-side) \rightarrow 0a1c : 3a2c

move 1c (child1 right-side left-side) \rightarrow 0a2c : 3a1c

move 2c (child1 child3 left-side right-side) \rightarrow 0a0c : 3a3c

Task 2

If each predicate have only 1 argument, there are 5 states for each predicate, so there are $2^{4 \times 5} = 2^{20}$ states

If each predicate have 4 arguments, there are 5^4 states for each predicate so there are $2^{4 \times 5^4} = 2^{2500}$ states. There are $[2^{20}, 2^{2500}]$ unique states

Task 3

after applying action $a_{AA}(B, C)$ to S

(A ttt 1)

(B ttt 1)

(C ttt 1)

(PPP1 B C)

(PPP2 A)

(PPP2 B)

(PPP3 C)

(EEE1 A C)

(EEE1 B C)

(EEE2 B)

(EEE2 C)

(EEE3 A)

Task 4

for online replanning:

If the boat has been blown off the course
it should try again.

for conditional planning:

(action move1a

:parameters (?adult ?boat ?from ?to)

:precondition (and (ADULT ?adult) (BOAT ?boat)
(SIDT ?from) (SIDT ?to))

(at ?adult ?from) (at ?boat ?from))

:effect (and (at ?adult ?to) (at ?boat ?to))

(not (at ?adult ?from)) (not (at ?boat ?from)))

OR

(and (ADULT ?adult) (BOAT ?boat))

(SIDT ?from) (SIDT ?to))

(at ?adult ?from) (at ?boat ?from))

(action move1c

:parameters (?child ?boat ?from ?to)

:precondition (and (CHILD ?child) (BoAT ?boat))

(SIDs ?from) (SIDs ?to))

(at ?child ?from) (at ?boat ?from))

:effect (and (at ?child ?to) (at ?boat ?to))

(not (at ?child ?from)) (not (at ?boat ?from)))

Or

(and (CHILD ?child) (BoAT ?boat))

(SIDs ?from) (SIDs ?to))

(at ?child ?from) (at ?boat ?from))

no need: