知識工学

第六回　プランニングと問題解決．中間試験

中間試験（60分）

- 持ち込み許可物は【印刷物，参考書，ノート】
- 1枚目裏面を使っても解答用紙が足りなくなった人は，２枚目を渡しに行くので手を挙げること．なお，全ての解答用紙に氏名・学籍番号を記入．
- 複数ページの場合は，下部にページ番号を記入．

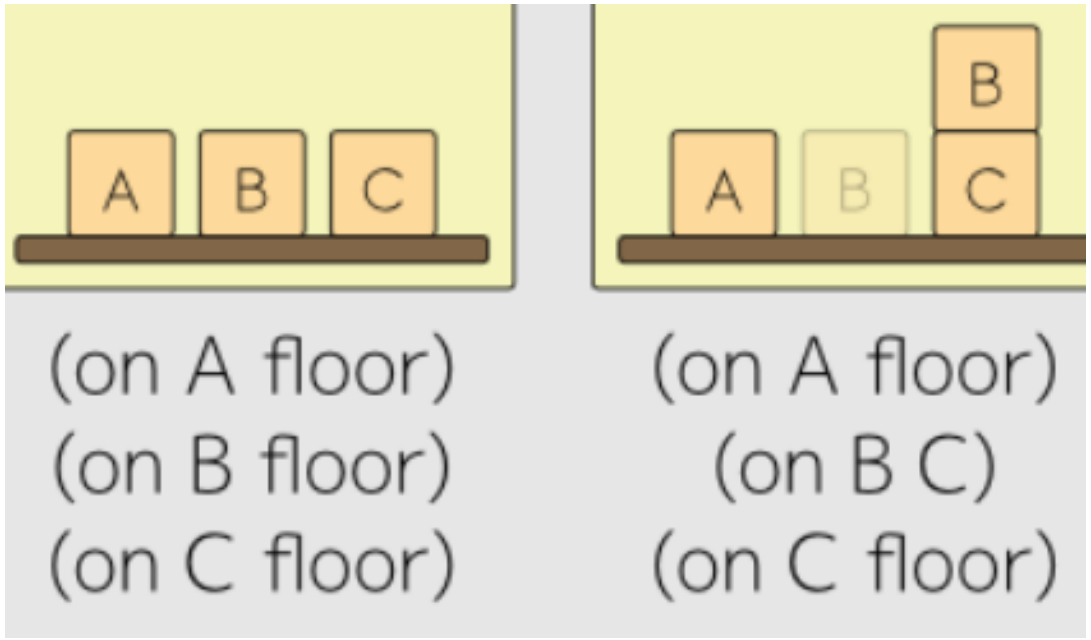
例）3ページ分使った場合：(1/3) (2/3) (3/3)

今日の内容

- ロボットは，センサーでものや周囲を認識しただけでは動けない
- 動くプランを立案させるにはどうするか

ロボットやエージェントから見た世界

- エージェント（ロボット）は，周囲の情報を**環境モデル**で認識している．
- 環境モデルとは，環境の情報を記号化したもの．
- 周囲を認識させ，自立的に動けるように設計することを**プランニング（自動計画）**という．

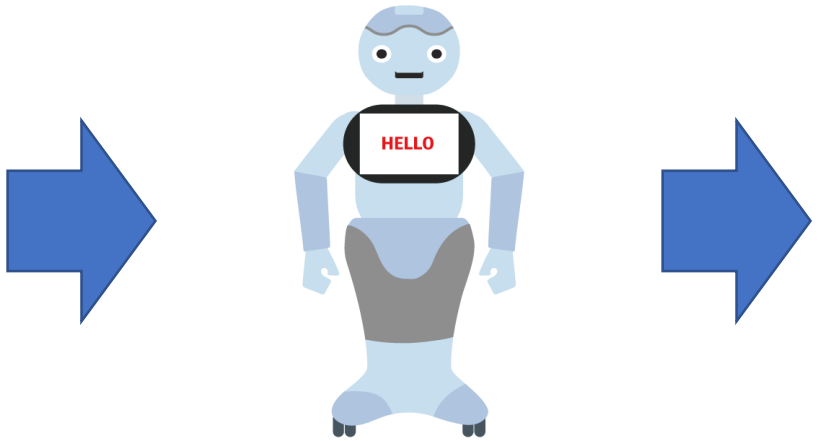


STRIPSプランニング

- 1971年に提案された，自動計画のための手法
- 述語論理によって環境を表現する
- 入力を与えると実行され，出力では実現のためのプランが得られる．

<u>入力</u>

1. 初期状態
2. 目標状態
3. 取りうる行動の集合

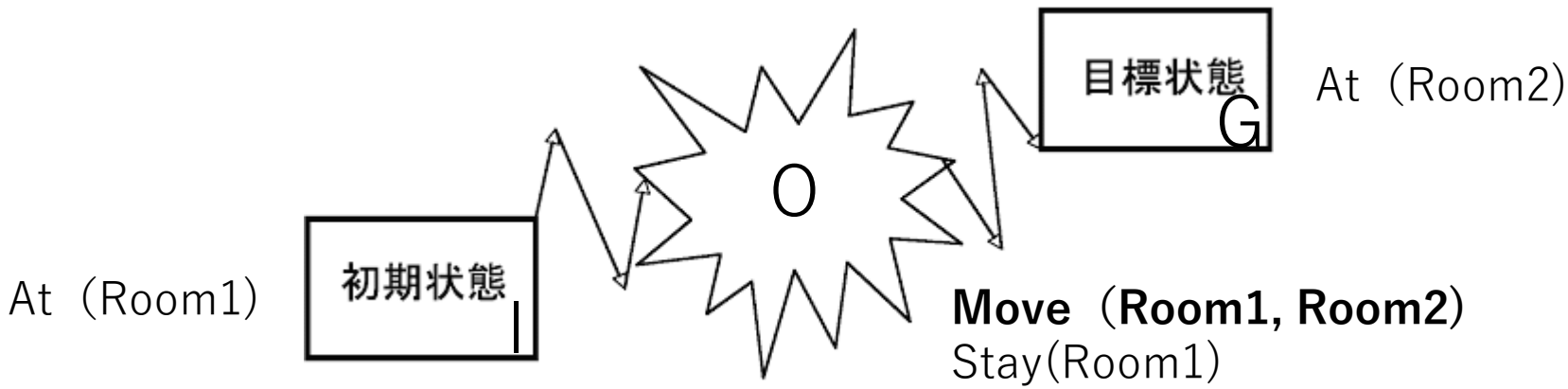


<u>出力</u>
目標状態を達成できる
手続きの系列

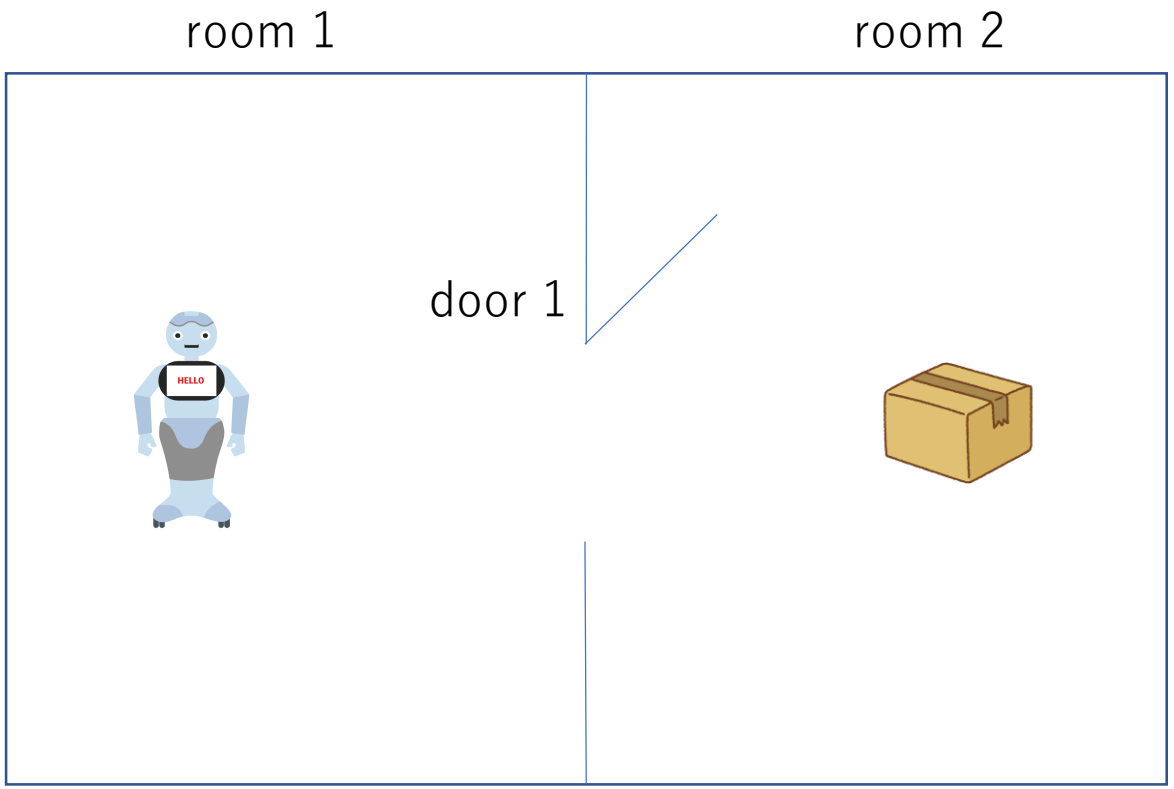
STRIPSの書き方

(O, I, G)の3つにより，1つの自動計画が表現される．

- O: Operators．取りうるアクションの集合
- I: Initial．初期状態を示す，真となっている命題の集合
- G: Goal．目標条件として，最終的に真となっているべき命題の集合

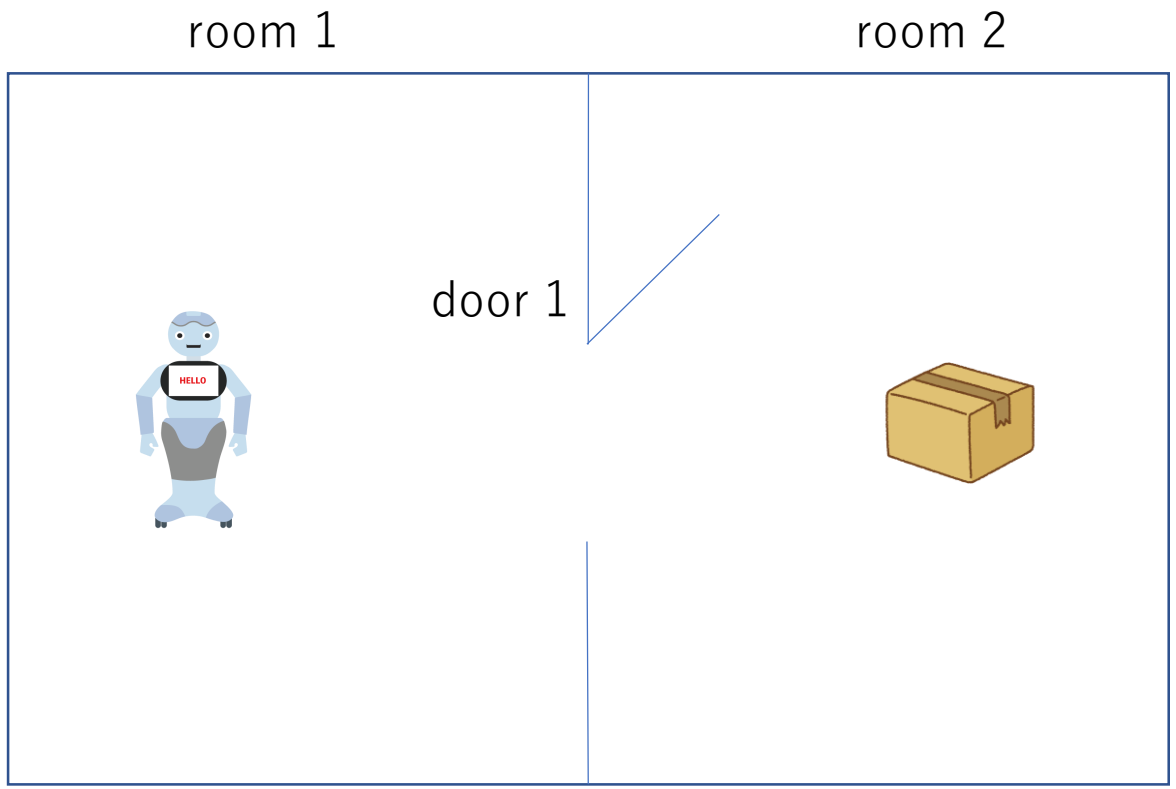


STRIPSプランニングの例



- ロボットは，Boxを取るために，その側まで行く．
- ロボットの内部にあるSTRIPSプランニングの仕組みに沿って行動する．
- ドアの認識だけでなく，照明の形や壁の材質などもセンサーで取得できる．本当なら，どれを取捨選択するかロボットに指示しなくてはならない．（**フレーム問題**）

STRIPSプランニングの例



初期状態の環境

inroom(A,R):物体AがRの中にある.

inroom(robot, room1)

inroom(box, room2)

status(D,S)：ドアDの状態がSである

status(door 1, open)

connect(D, RX, RY): ドアDは，部屋RXとRYをつないでいる.

connect(door1, room1, room2)

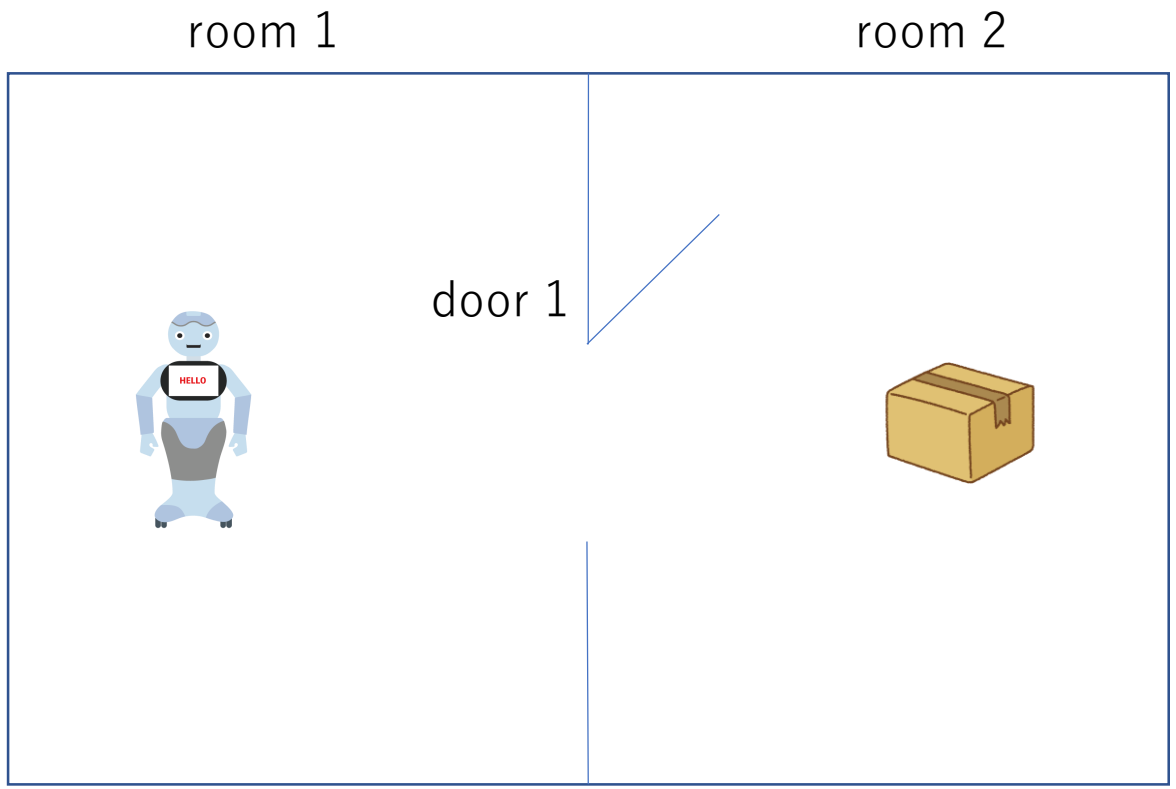
nextto(A,B)：物体Aが物体Bの側にある

なし

目標状態

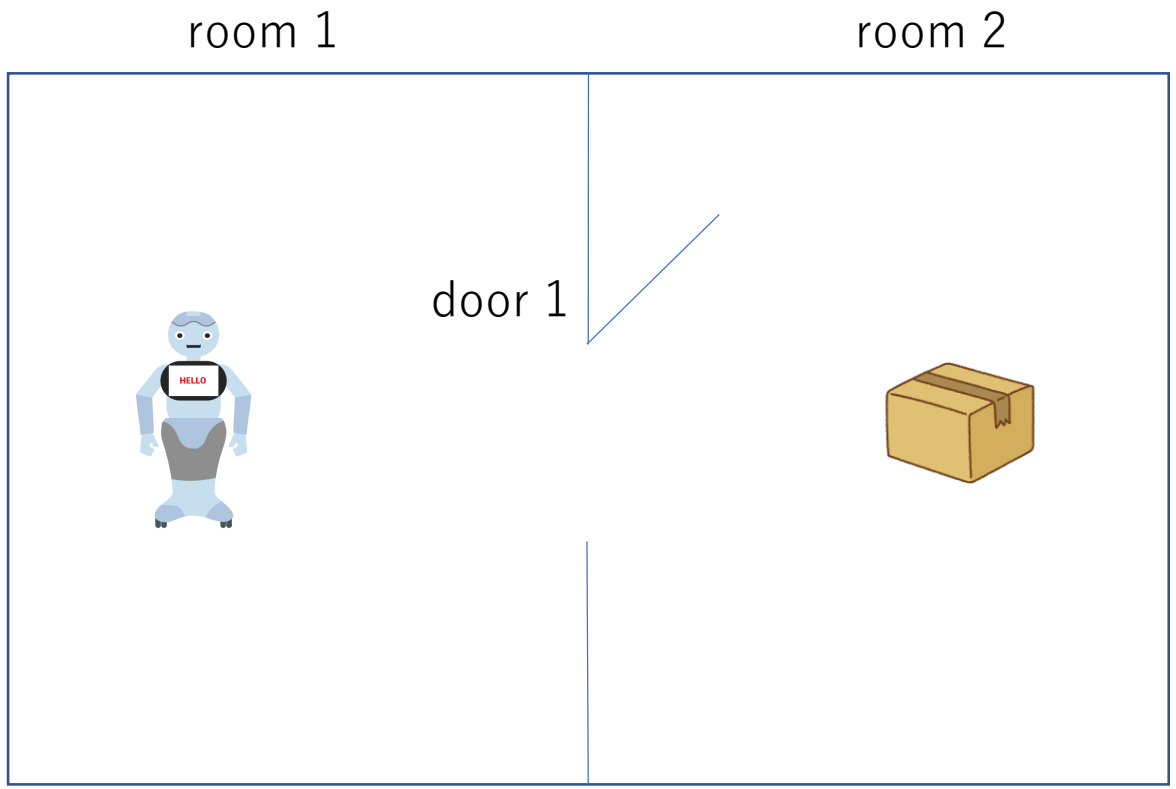
nextto(robot, box)

STRIPSプランニングの例



- 初期状態から目標状態までの差異を取り出し，目標達成のためのオペレータを選択する．
- そのオペレータを適用できる条件を満たせるように**副目標**を設定し，副目標を実現するオペレータを選択することを繰り返していく．
  例）荷物を取り出す前に，梱包を解く

STRIPSプランニングの例



オペレータ

gotod(RX, DX): 部屋RXにて，ドアDXに近づく

条件リスト：connect(DX, RX, RY)
inroom(robot, RX)

削除リスト：nextto(robot, A)

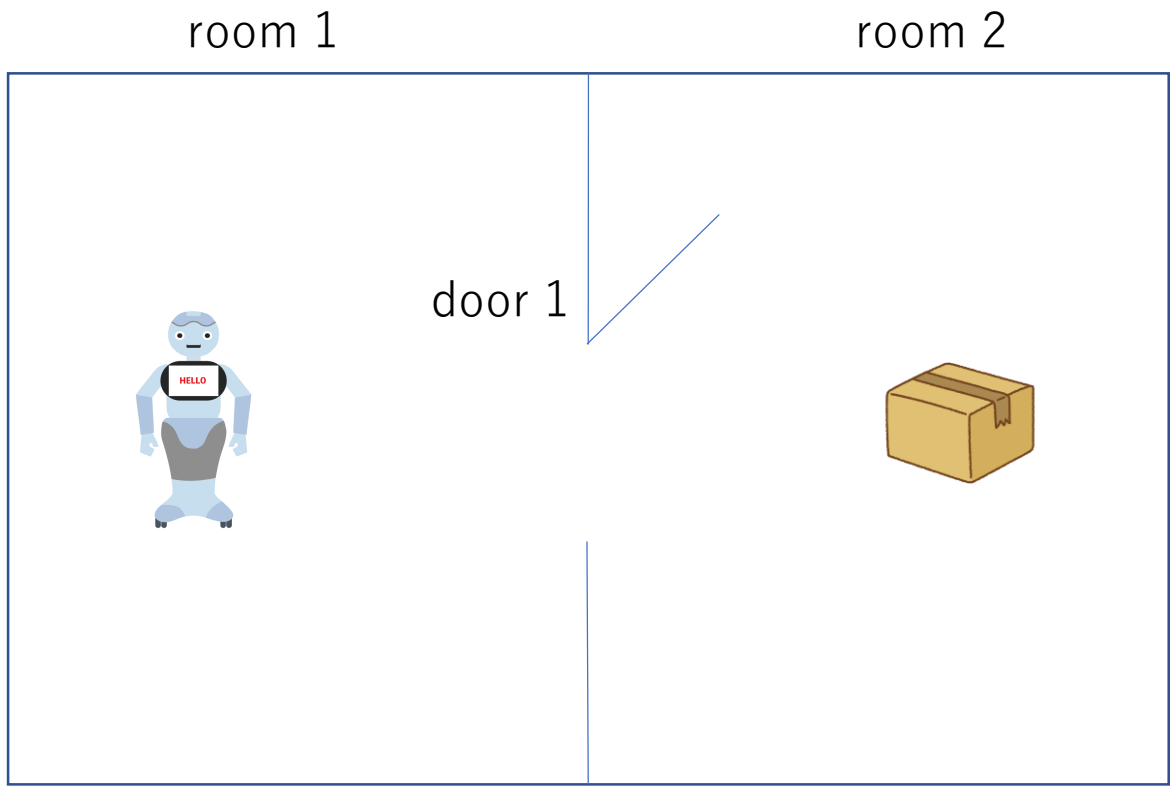
追加リスト：nextto(robot, DX)

環境モデル

1. inroom(robot, room1)
2. inroom(box, room2)
3. connect(door1, room1, room2)
4. status(door 1, open)

条件を満たしたときにgotodを作用させ，環境モデルの内容を削除，追加する．

STRIPSプランニングの例

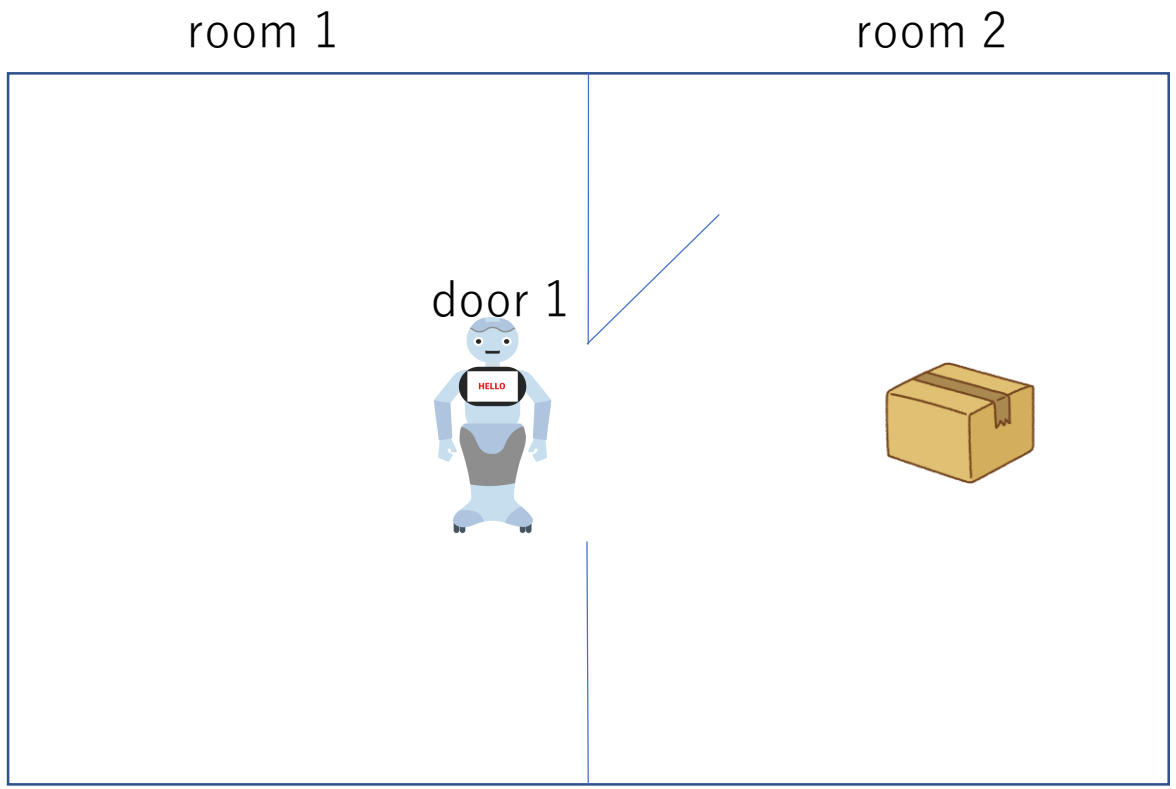


オペレータの適用

- 環境モデルの中で，条件リストにマッチするものがあればgoto(robot, Y)を手続きに追加.
  → goto(robot, door1)を追加.

- 追加後は環境モデルに，nextto(robot, door1)を追加.
- 元の位置Aからは離れるため，側に物があれば環境モデルから削除する.

STRIPSプランニングの例



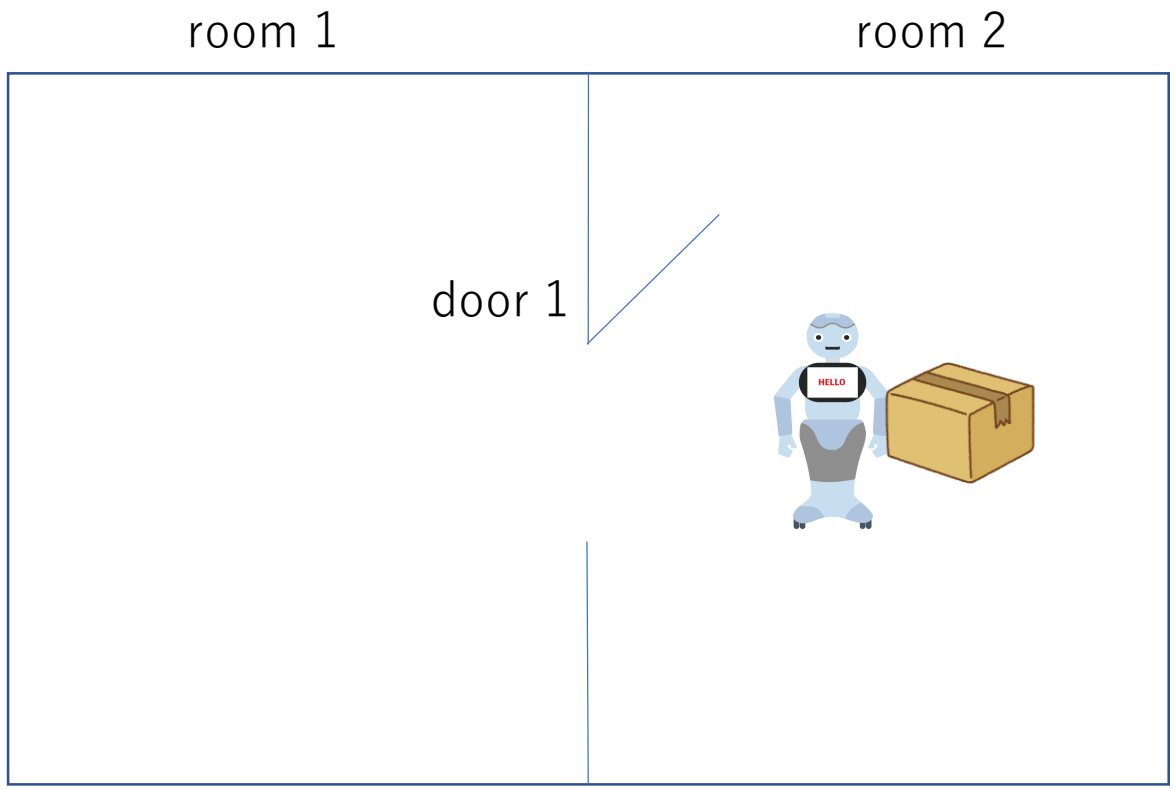
更新された環境モデル

1. inroom(robot, room1)
2. inroom(box, room2)
3. status(door 1, open)
4. connect(door1, room1, room2)
5. nextto(robot, door 1)

ここから，gothrough(room 1, room 2)やgoto(room2, box)などのオペレータを手続きに加えていく．

※ gothrough(door 1, room 2) ：door 1を通りroom2に行く．

STRIPSプランニングの例



出力として得られる手続きの系列

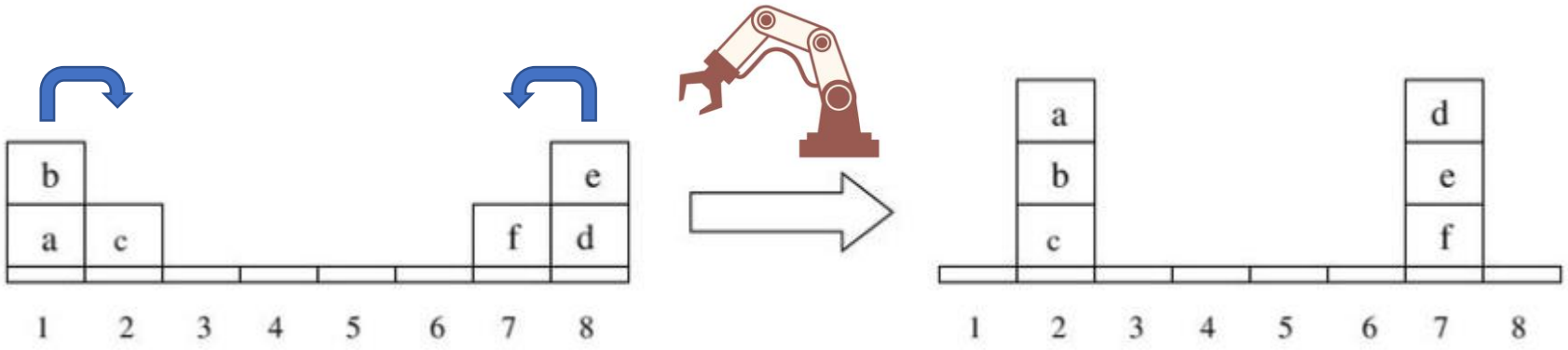
1. gotod(door 1)オペレータによりdoor 1の近くに行く
2. gothroughd(door 1, room 2)オペレータにより，door 1を通ってroom 2へ行く
3. goto(room2, box)オペレータにより，boxの近くへ行く

得られた手続きを順番に実行することで，目標状態にする．
（手続きの系列を全て確定させてから動作する）

半順序プランニング

- 順番が決まっているタスクと決まっていないタスクが混在した状態のプランニングを**半順序プランニング**という

例）aを持ち上げる前には，bの移動が先という優先順位がある
しかし，bを移動させるか，eを移動させるかはどちらからでも良い



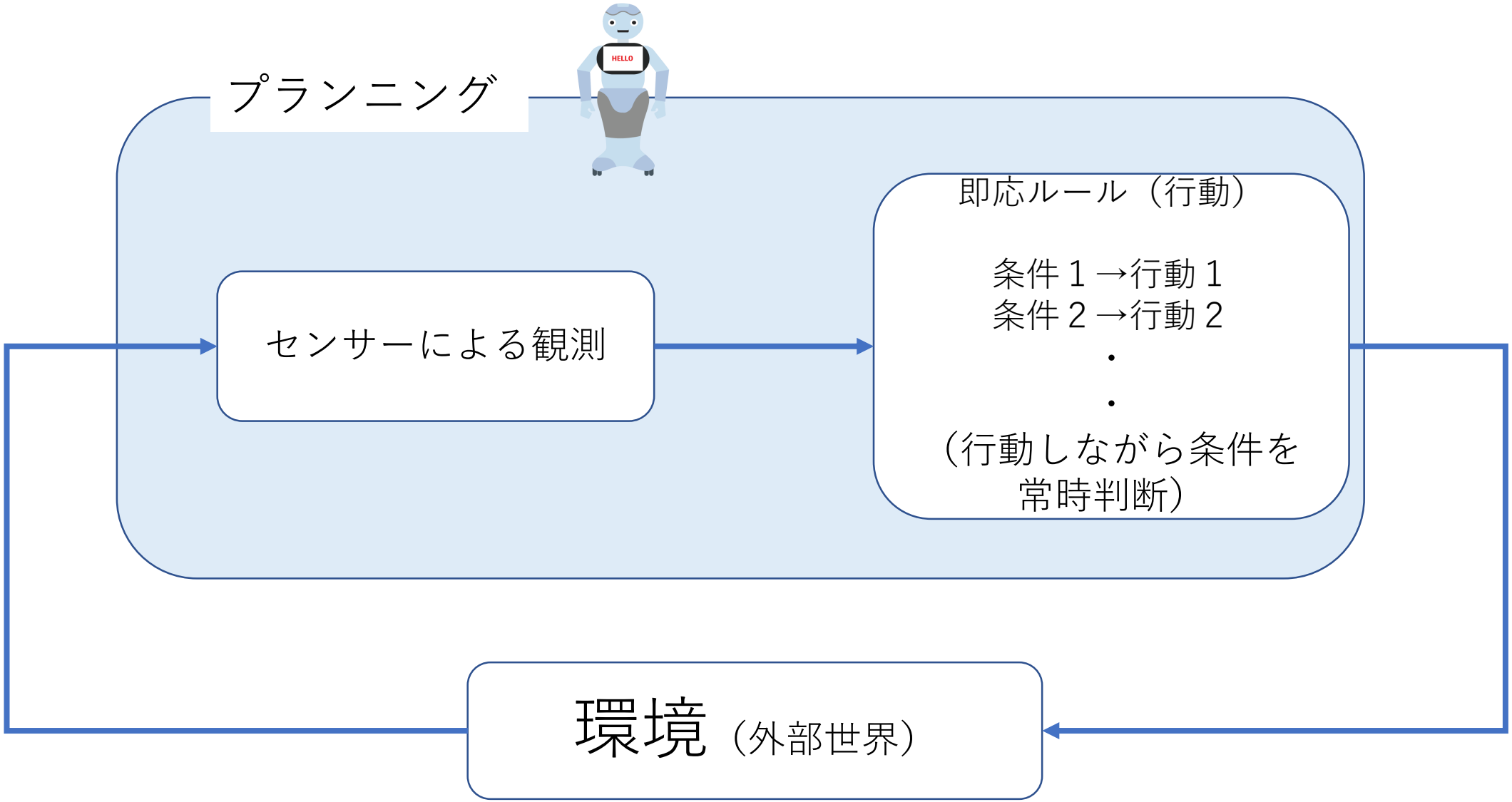
初期状態　　目標状態

即応プランニング

- これまでのプランニングでは，環境の測定，プラン作成などで多くの計算時間，待機時間が必要
- その上，環境は変化しないことが前提とされている

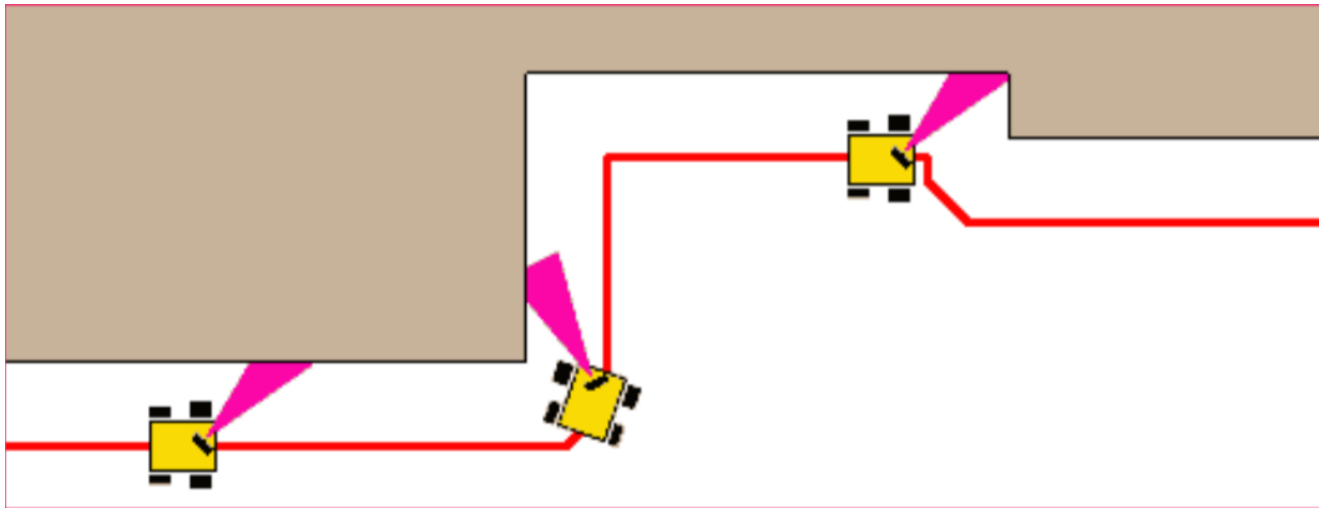
- そのため，動的な環境（現実世界）での**即応プランニング**が誕生した.

即応プランニングの仕組み

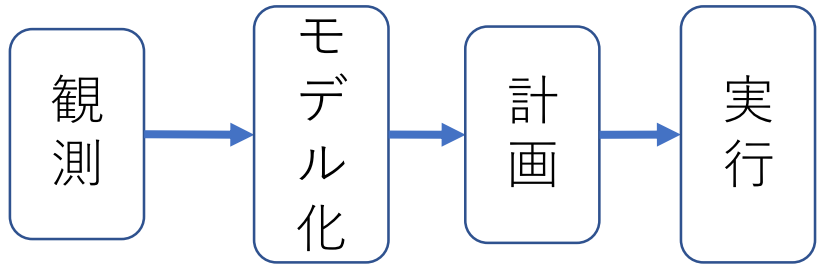


即応ルールの例

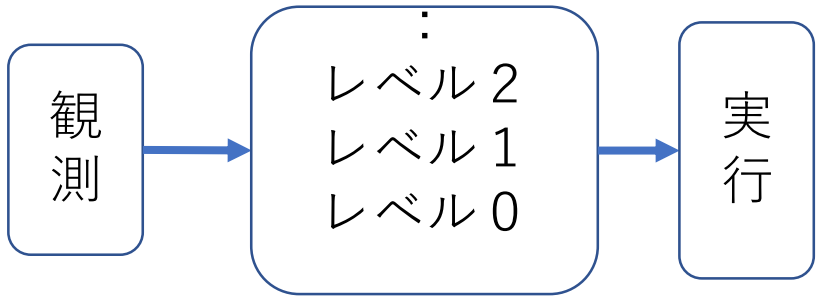
- ルールA（凹コーナを曲がる）
  - IF 前方の壁が10cm以内に近づいて，左10cm 以内に壁がある
  - THEN 時計方向に40° 回転する
- ルールB（凸コーナを曲がる）
  - IF 左右5cm，前方10cm以内に障害物なし
  - THEN 10cm前方に進み，反時計方向に40° 回転する
- ルールC（壁に近づきすぎたら離れる）
  - IF 壁に5cm以内に近づいた
  - THEN 右に13.5° 曲がる
- ルールD（壁から離れすぎたら近づく）
  - IF 壁から5cm以上離れた
  - THEN 左に13.5° 曲がる



包摂アーキテクチャ



- これまでは，周囲観測，環境モデリング，プランニング，実行が順序立てて，独立的に行われていた．



- 包摂アーキテクチャでは，動作を階層的に表現する．
- 低レベルは反射などの本能的動作，高次に行くほどより抽象的になる．

  例）回避→うろつく→探索する

レベルの例

レベル３：地図作成
レベル２：探索
レベル１：徘徊
レベル０：障害物回避

これらはセンサ入力をもとに，並列して処理が進む.

1989年のGhengphis robot　→
開発者はロドニー・ブルックス.
1990年にはiRobot社を設立
2002年にはルンバをリリース

本日はレビューシートなし
お疲れさまでした.