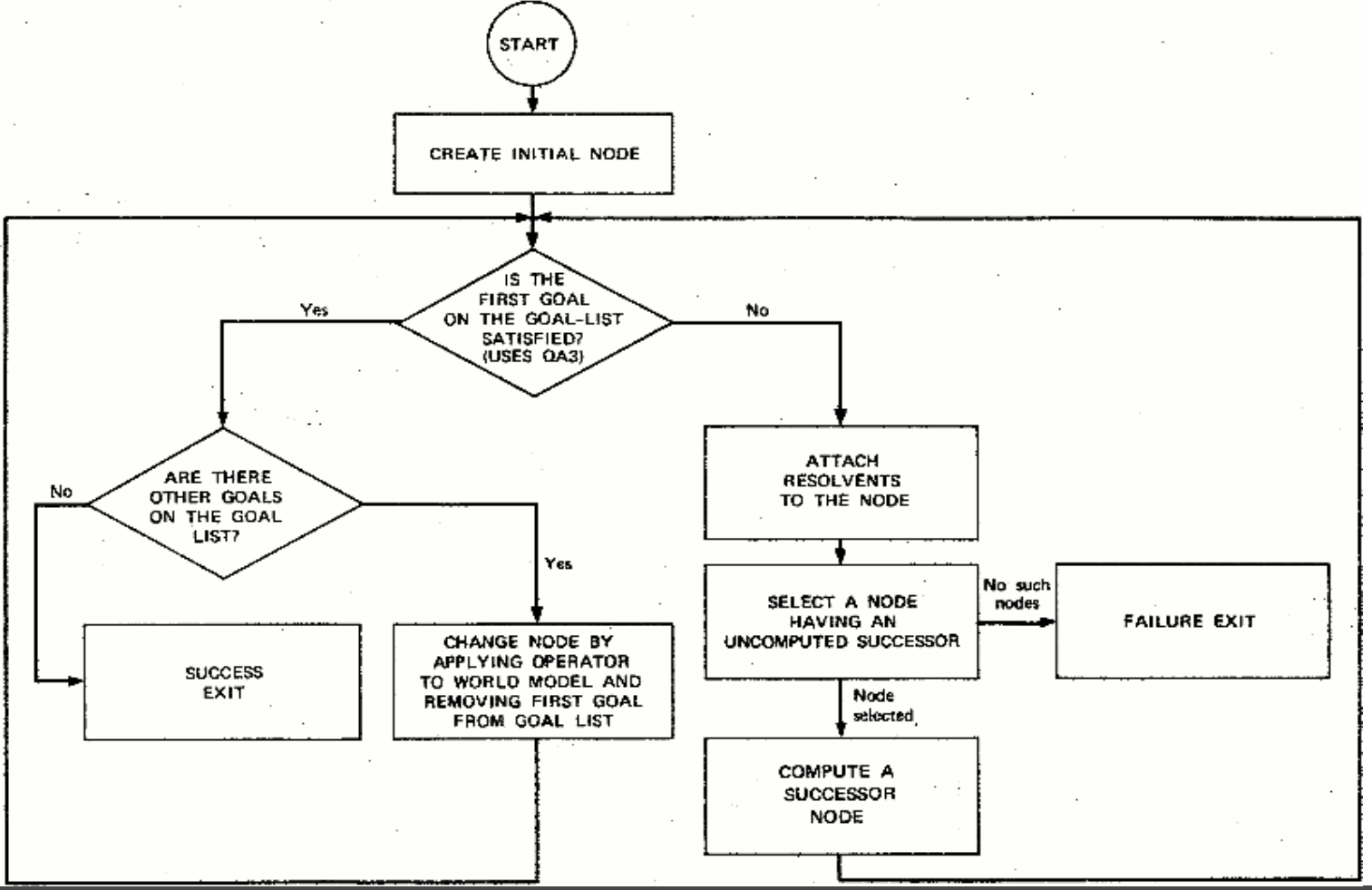
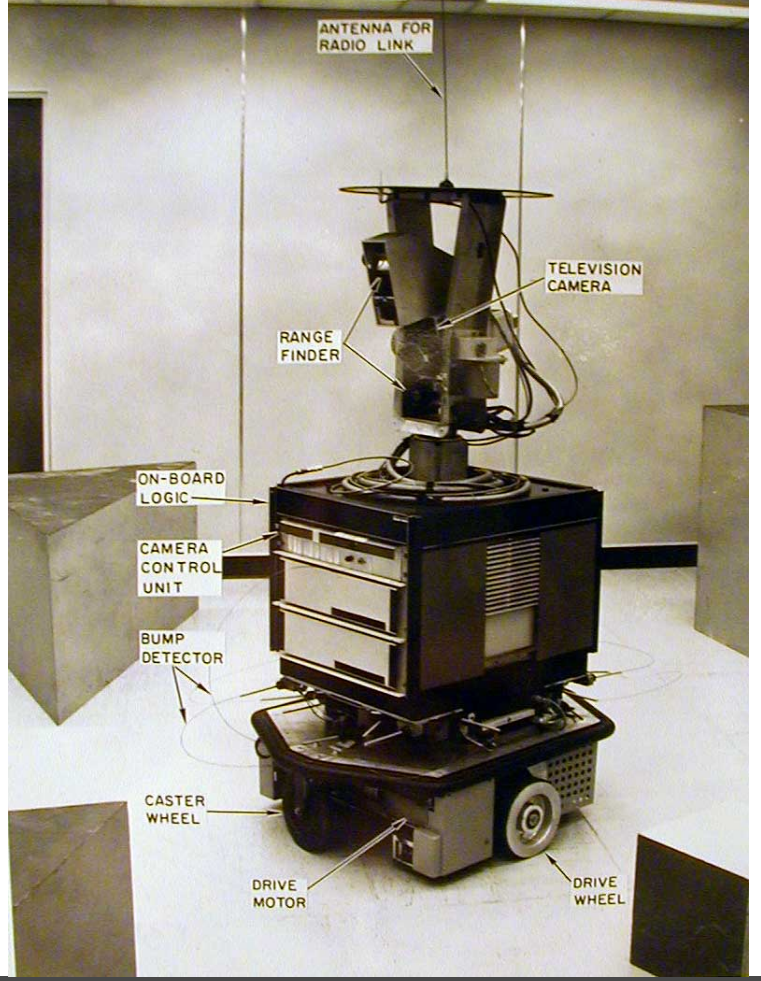




PLANNING

ECS170 Spring 2018
Josh McCoy, jamccoy@ucdavis.edu

Planning Algorithms

- They search over possible action sequences (i.e. plans).
- Possibility space defined at design-time and searched from at run-time.
- Look at the possible futures created by changing the world.
- Goal-driven (mostly).

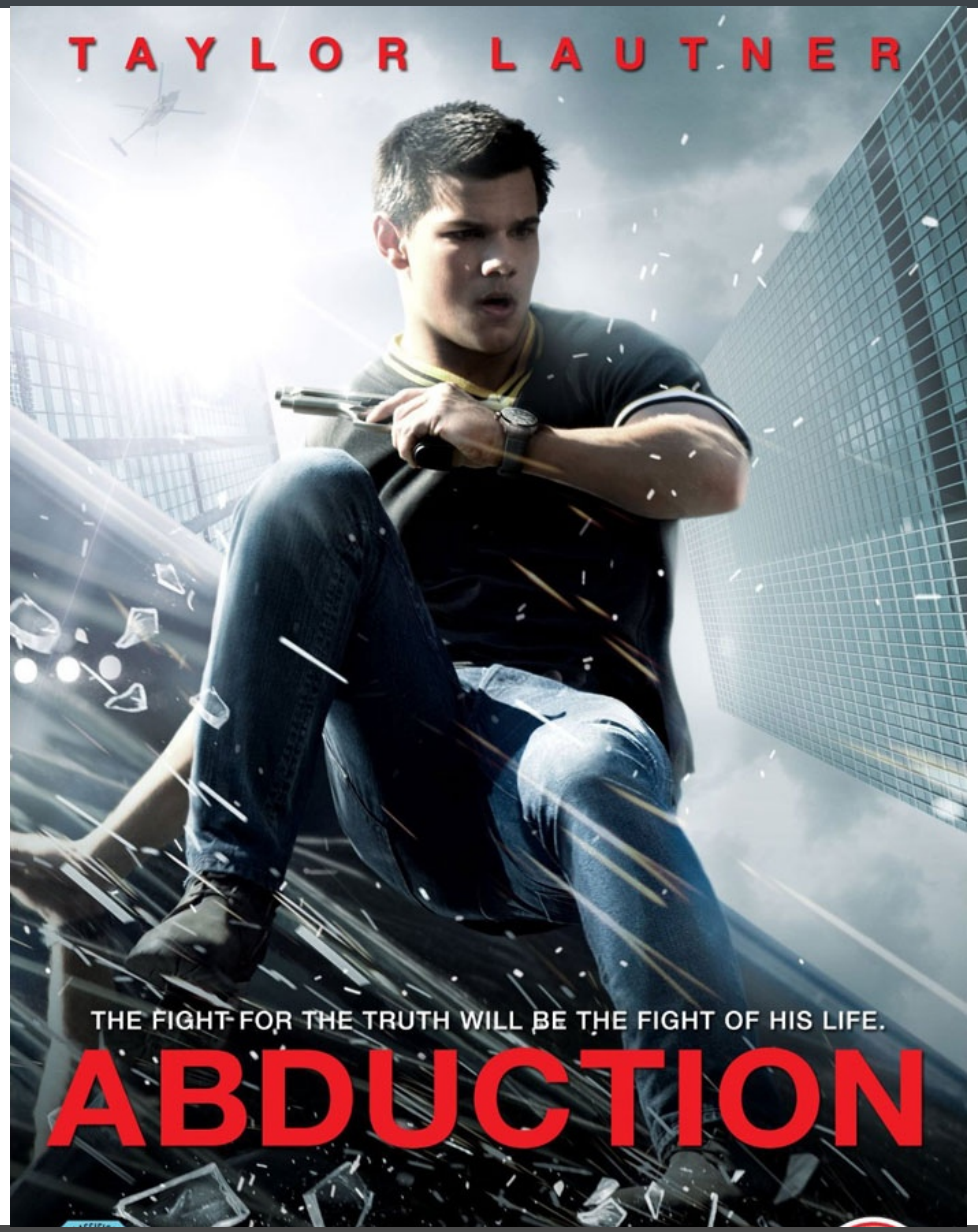
Types of Reasoning

Deduction

Induction

Abduction

Abduction!


TAYLOR LAUTNER
THE FIGHT FOR THE TRUTH WILL BE THE FIGHT OF HIS LIFE.
ABDUCTION

Automated Planning

Determine sequences of actions or strategies to achieve goals.

Used in:

- Intelligent agents
- Autonomous robots
- Unmanned vehicles
- Games

Attributes of Planners

- Parallel and serial actions.
- Actions with cost
- Stochastic or deterministic effects
- Minimum required actions
- Offline and online
- Single or multiple agents
- World state fully or partially observable
- Goals achieved or maintained
- Discrete or continuous state and actions
- Durative actions
- …

Classical Planning Problem

- Initial world state observable and known
- Instant and deterministic actions
- Serial actions
- Single Agent

STRIPS

https://www.youtube.com/watch?v=mQ7M-zhiu7U

STRIPS

- Possible world conditions P
- Operators/actions O. Each has four sets:
  - Condition that must be true to do the action.
  - Conditions that must be false to do the action.
  - Conditions that are true as a result of the action.
  - Conditions that are false as a result of the action.
- Initial state I
- Goal state G

STRIPS

STRIPS(*O*, s, *G*)
p = empty plan
loop…
    if *s* satisfies *G* then return *p*
    a = [an applicable action in *O*,
        relevant for *G*]
    if *a* = null, then return failure
    p' = STRIPS(*O*, *s*, precond(*a*))
    if *p'* = failure, then return failure
    s = apply *p'* to *s*
    s = apply *a* to *s*
    p = *p* + *p'* + *a*

Ghallab, Malik, Dana Nau, and Paolo Traverso. Automated planning: theory & practice. Access Online via Elsevier, 2004.

Simple STRIPS Encoding

Operators
- Preconditions
- Add List
- Delete List

Initial State

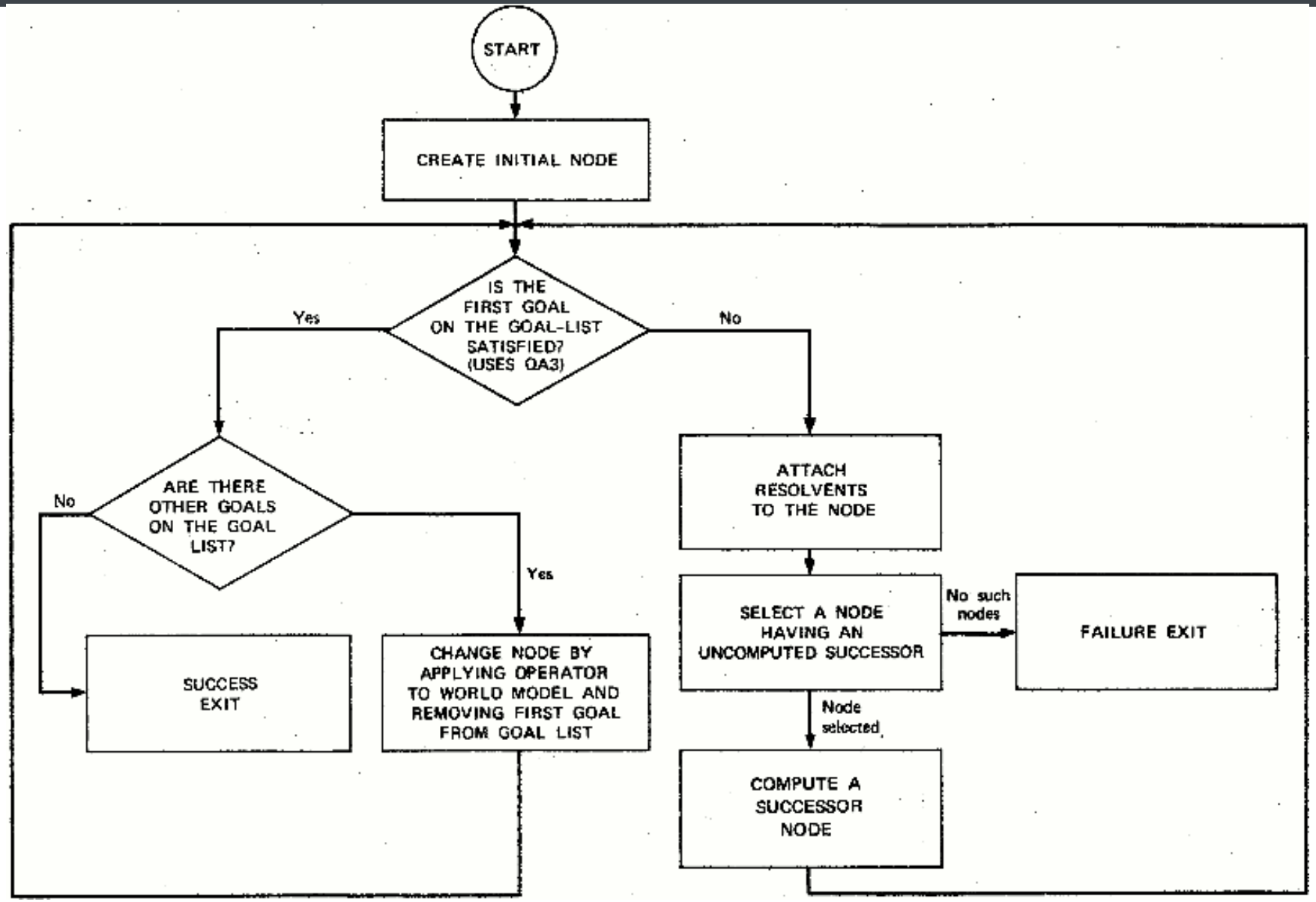
Goal State

GrabSpellBook
- floor(spell_book)
- +inventory(spell_book)
- -floor(spell_book)

Initial State
- floor(spell_book)

Goal State
- inventory(spell_book)

The Original STRIPS Algorithm



STRIPS Vagaries Explained

Resolvents are metric of how relevant the next possible actions are.

Compute a successor node: adds a node to the internal A*.

That's right: A*!

Both developed at SRI AI center.

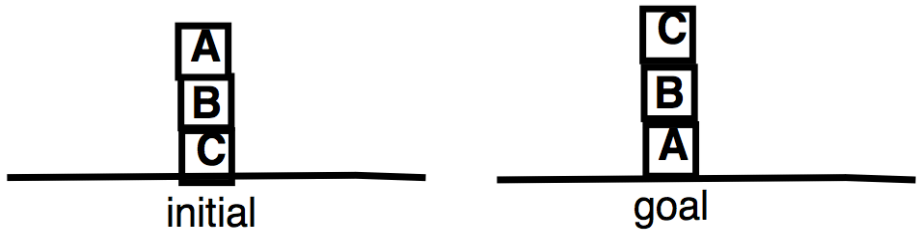
STRIPS uses it to expand plans based on:

g(n)+h(n) cost evaluation.

What could you use as the heuristic?

In a sense, this gives forward and backward search to the goal.

STRIPS - Block World




```
; Invert stack (good goal ordering)
> (gps '((a on b)(b on c) (c on table) (space on a) (space on table))
    '((b on a) (c on b)))
Goal: (B ON A)
Consider: (MOVE B FROM C TO A)
  Goal: (SPACE ON B)
  Consider: (MOVE A FROM B TO TABLE)
    Goal: (SPACE ON A)
    Goal: (SPACE ON TABLE)
    Goal: (A ON B)
  Action: (MOVE A FROM B TO TABLE)
  Goal: (SPACE ON A)
  Goal: (B ON C)
Action: (MOVE B FROM C TO A)
Goal: (C ON B)
Consider: (MOVE C FROM TABLE TO B)
  Goal: (SPACE ON C)
  Goal: (SPACE ON B)
  Goal: (C ON TABLE)
Action: (MOVE C FROM TABLE TO B)
((START)
 (EXECUTING (MOVE A FROM B TO TABLE))
 (EXECUTING (MOVE B FROM C TO A))
 (EXECUTING (MOVE C FROM TABLE TO B)))
```

Ghallab, Malik, Dana Nau, and Paolo Traverso. Automated planning: theory & practice. Access Online via Elsevier, 2004.

STRIPS - Monkey

Initial state: At(A), Level(low), BoxAt(C), BananasAt(B)

Goal state: Have(Bananas)

Actions:

// move from X to Y

Move(X, Y)

Preconditions: At(X), Level(low)

Postconditions: not At(X), At(Y)

// climb up on the box

ClimbUp(Location)

Preconditions: At(Location), BoxAt(Location), Level(low)

Postconditions: Level(high), not Level(low)

// climb down from the box

ClimbDown(Location)

Preconditions: At(Location), BoxAt(Location), Level(high)

Postconditions: Level(low), not Level(high)

// move monkey and box from X to Y

MoveBox(X, Y)

Preconditions: At(X), BoxAt(X), Level(low)

Postconditions: BoxAt(Y), not BoxAt(X), At(Y), not At(X)

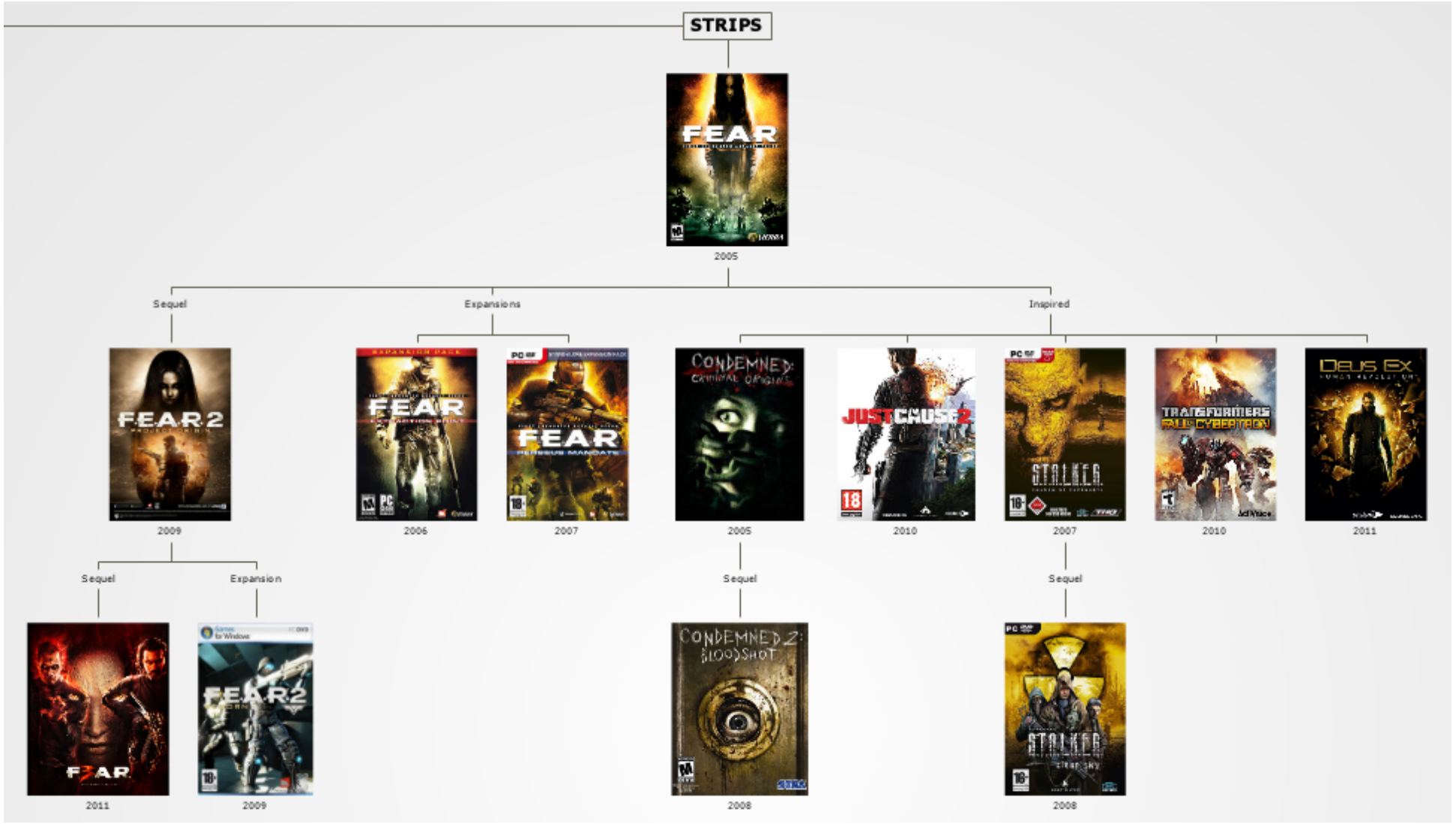
// take the bananas

TakeBananas(Location)

Preconditions: At(Location), BananasAt(Location), Level(high)

Postconditions: Have(bananas)

STRIPS in Games



http://aigamedev.com/open/review/planning-in-games/

Implementations of STRIPS

- http://www.primaryobjects.com/2015/11/06/artificial-intelligence-planning-with-strips-a-gentle-introduction/
- https://stripsfiddle.herokuapp.com/

A* GOAP Example

- In this example, each element in the state vector takes on Boolean values

{
Recipe: true if character has a recipe
Pizza Phone #: true if character has a phone number for ordering pizza
Food: true if something to eat is in possession
Ingredients: true if the ingredients to make the recipe are in possession
Silverware: true if the silverware is in possession
Hungry: true if the character is hungry
Requires Silverware: true if the food requires silverware to eat
}

Facing Your F.E.A.R.

A* GOAP Example

Look up recipe (lur)

P: {N, _, _, _, _, _, _}

E: {Y, _, _, _, _, _, _}

Look up phone number (luph)

P: {_, N, _, _, _, _, _}

E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)

P: {Y, _, _, N, _, _, _}

E: {_, _, _, Y, _, _, _}

Call Pizza (cp)

P: {_, Y, N, _, _, _, _}

E: {_, _, Y, _, _, _, N}

Cook (c)

P: {Y, _, N, Y, _, _, _}

E: {_, _, Y, N, _, _, Y}

Get silverware (gs)

P: {_, _, _, _, N, _, _}

E: {_, _, _, _, Y, _, _}

Eat with silverware (ews)

P: {_, _, Y, _, Y, Y, Y}

E: {_, _, N, _, _, N, _}

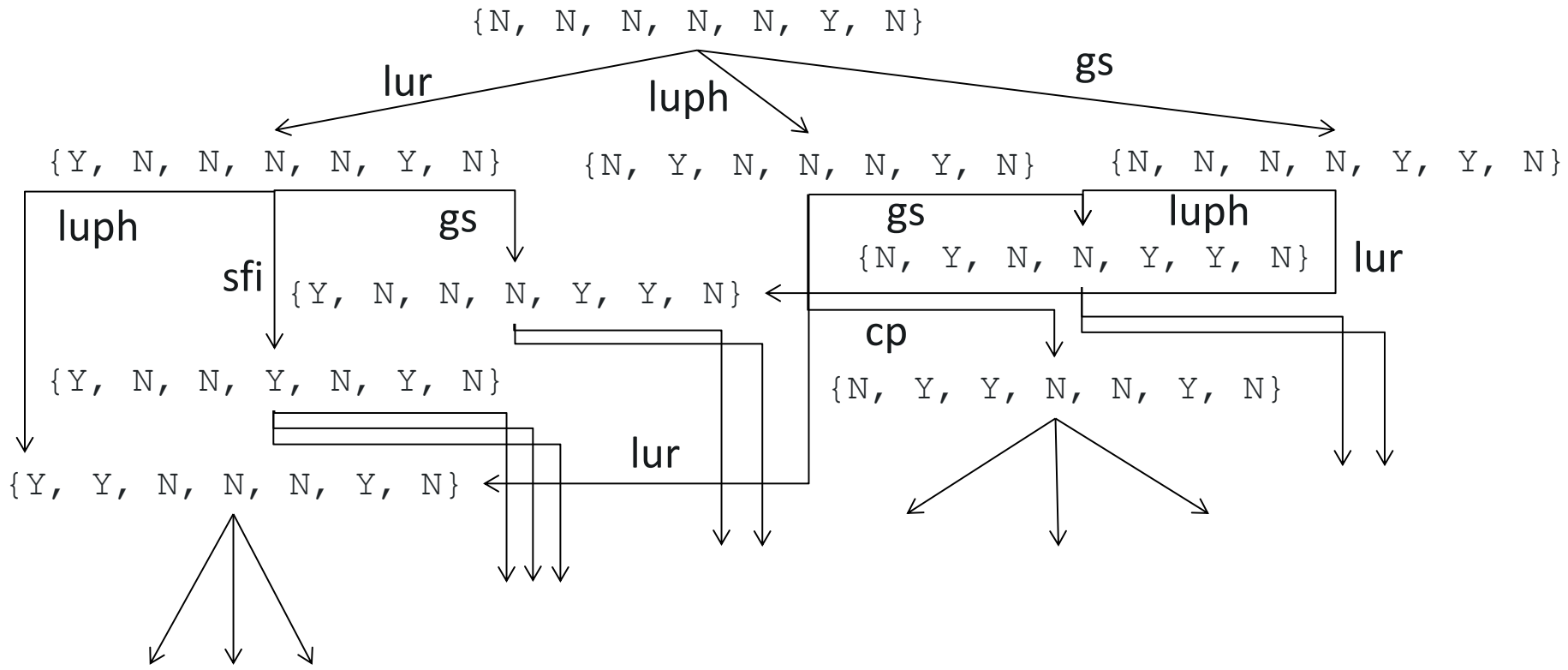
Eat with hands (ewh)

P: {_, _, Y, _, N, Y, N}

E: {_, _, N, _, _, N, _}

{Recipe, Phone #, Food, Ingredients, Silverware, Hungry, Need Silverware}

A* GOAP Example



A* GOAP Example

Initial State: {N, N, N, N, N, Y, N}
Goal State: {_, _, _, _, _, N, _}
Heuristic: number of state elements different from goal

Open list: [{N, N, N, N, N, Y, N} cost 1]
Valid actions: lur, luph, gs
Open list:
{**Y**, N, N, N, N, Y, N} lur cost: 2,
{N, **Y**, N, N, N, Y, N} luph cost: 2,
{N, N, N, N, **Y**, Y, N} gs cost: 2

Closed list: [{N, N, N, N, N, Y, N} cost: 1]

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{Y, N, N, N, N, Y, N} lur cost: 2

Valid actions: luph, sfi, gs

Open list:

{N, Y, N, N, N, Y, N} luph cost: 2,

{N, N, N, N, Y, Y, N} gs cost: 2

{Y, **Y**, N, N, N, Y, N} luph cost: 3

{Y, N, N, **Y**, N, Y, N} sfi cost: 3

{Y, N, N, N, **Y**, Y, N} gs cost: 3

Closed list:

{N, N, N, N, N, Y, N} cost: 1

{Y, N, N, N, N, Y, N} lur cost: 2

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{N, Y, N, N, N, Y, N} luph cost: 2,

Valid actions: lur (duplicate state), cp, gs

Open list:

{N, N, N, N, Y, Y, N} gs cost: 2

{Y, Y, N, N, N, Y, N} luph cost: 3

{Y, N, N, Y, N, Y, N} sfi cost: 3

{Y, N, N, N, Y, Y, N} gs cost: 3

{N, Y, **Y**, N, N, Y, **N**} cp cost: 3

{N, Y, N, N, **Y**, Y, N} gs cost: 3

Closed list:

{N, N, N, N, N, Y, N} cost: 1

{Y, N, N, N, N, Y, N} lur cost: 2

{N, Y, N, N, N, Y, N} luph cost: 2

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{N, N, N, N, Y, Y, N} gs cost: 2

Valid actions: lur (duplicate), luph (duplicate)

Open list:

{Y, Y, N, N, N, Y, N} luph cost: 3

{Y, N, N, Y, N, Y, N} sfi cost: 3

{Y, N, N, N, Y, Y, N} gs cost: 3

{N, Y, Y, N, N, Y, N} cp cost: 3

{N, Y, N, N, Y, Y, N} gs cost: 3

Closed list:

{N, N, N, N, N, Y, N} cost: 1

{Y, N, N, N, N, Y, N} lur cost: 2

{N, Y, N, N, N, Y, N} luph cost: 2

{N, N, N, N, Y, Y, N} gs cost: 2

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{N, Y, N, N, N, Y, N} luph cost: 2,

Valid actions: lur (duplicate state), cp, gs

Open list:

{N, N, N, N, Y, Y, N} gs cost: 2

{Y, Y, N, N, N, Y, N} luph cost: 3

{Y, N, N, Y, N, Y, N} sfi cost: 3

{Y, N, N, N, Y, Y, N} gs cost: 3

{N, Y, **Y**, N, N, Y, **N**} cp cost: 3

{N, Y, N, N, **Y**, Y, N} gs cost: 3

Closed list:

{N, N, N, N, N, Y, N} cost: 1

{Y, N, N, N, N, Y, N} lur cost: 2

{N, Y, N, N, N, Y, N} luph cost: 2

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{N, N, N, N, Y, Y, N} gs cost: 2

Valid actions: lur (duplicate), luph (duplicate)

Open list:

{Y, Y, N, N, N, Y, N} luph cost: 3

{Y, N, N, Y, N, Y, N} sfi cost: 3

{Y, N, N, N, Y, Y, N} gs cost: 3

{N, Y, Y, N, N, Y, N} cp cost: 3

{N, Y, N, N, Y, Y, N} gs cost: 3

Closed list:

{N, N, N, N, N, Y, N} cost: 1

{Y, N, N, N, N, Y, N} lur cost: 2

{N, Y, N, N, N, Y, N} luph cost: 2

{N, N, N, N, Y, Y, N} gs cost: 2

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

DIY: 6 Steps Left

A* GOAP Example

{Y, Y, N, N, N, Y, N} luph cost: 3

Valid actions: sfi, cp, gs

Open list:

{Y, N, N, Y, N, Y, N} sfi cost: 3
{Y, N, N, N, Y, Y, N} gs cost: 3
{N, Y, Y, N, N, Y, N} cp cost: 3
{N, Y, N, N, Y, Y, N} gs cost: 3
{Y, Y, N, **Y**, N, Y, N} sfi cost: 4
{Y, Y, **Y**, N, N, Y, N} cp cost: 4
{Y, Y, N, N, **Y**, Y, N} gs cost: 4

Closed list:

{N, N, N, N, N, Y, N} cost: 1
{Y, N, N, N, N, Y, N} lur cost: 2
{N, Y, N, N, N, Y, N} luph cost: 2
{N, N, N, N, Y, Y, N} gs cost: 2
{Y, Y, N, N, N, Y, N} luph cost: 3

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{Y, N, N, Y, N, Y, N} sfi cost: 3

Valid actions: luph (duplicate), c, gs

Open list:

{Y, N, N, N, Y, Y, N} gs cost: 3
{N, Y, Y, N, N, Y, N} cp cost: 3
{N, Y, N, N, Y, Y, N} gs cost: 3
{Y, Y, N, Y, N, Y, N} sfi cost: 4
{Y, Y, Y, N, N, Y, N} cp cost: 4
{Y, Y, N, N, Y, Y, N} gs cost: 4
{Y, N, **Y**, Y, N, Y, **Y**} c cost: 4
{Y, N, N, Y, **Y**, Y, N} gs cost: 4

Closed list:

{N, N, N, N, N, Y, N} cost: 1
{Y, N, N, N, N, Y, N} lur cost: 2
{N, Y, N, N, N, Y, N} luph cost: 2
{N, N, N, N, Y, Y, N} gs cost: 2
{Y, Y, N, N, N, Y, N} luph cost: 3
{Y, N, N, Y, N, Y, N} sfi cost: 3

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{Y, N, N, N, Y, Y, N} gs cost: 3

Valid actions: luph (dup), sfi (dup)

Open list:

{N, Y, Y, N, N, Y, N} cp cost: 3
{N, Y, N, N, Y, Y, N} gs cost: 3
{Y, Y, N, Y, N, Y, N} sfi cost: 4
{Y, Y, Y, N, N, Y, N} cp cost: 4
{Y, Y, N, N, Y, Y, N} gs cost: 4
{Y, N, Y, Y, N, Y, Y} c cost: 4
{Y, N, N, Y, Y, Y, N} gs cost: 4

Closed list:

{N, N, N, N, N, Y, N} cost: 1
{Y, N, N, N, N, Y, N} lur cost: 2
{N, Y, N, N, N, Y, N} luph cost: 2
{N, N, N, N, Y, Y, N} gs cost: 2
{Y, Y, N, N, N, Y, N} luph cost: 3
{Y, N, N, Y, N, Y, N} sfi cost: 3
{Y, N, N, N, Y, Y, N} gs cost: 3

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{N, Y, Y, N, N, Y, N} cp cost: 3

Valid actions: lur (dup), gs, ewh

Open list:

{N, Y, N, N, Y, Y, N} gs cost: 3
{Y, Y, N, Y, N, Y, N} sfi cost: 4
{Y, Y, Y, N, N, Y, N} cp cost: 4
{Y, Y, N, N, Y, Y, N} gs cost: 4
{Y, N, Y, Y, N, Y, Y} c cost: 4
{Y, N, N, Y, Y, Y, N} gs cost: 4
{N, Y, Y, N, **Y**, Y, N} gs cost: 4
{N, Y, **N**, N, N, **N**, N} ewh cost: 3

Closed list:

{N, N, N, N, N, Y, N} cost: 1
{Y, N, N, N, N, Y, N} lur cost: 2
{N, Y, N, N, N, Y, N} luph cost: 2
{N, N, N, N, Y, Y, N} gs cost: 2
{Y, Y, N, N, N, Y, N} luph cost: 3
{Y, N, N, Y, N, Y, N} sfi cost: 3
{Y, N, N, N, Y, Y, N} gs cost: 3
{N, Y, Y, N, N, Y, N} cp cost: 3

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{N, Y, N, N, Y, Y, N} gs cost: 3

Valid actions: lur (dup), cp (dup)

Open list:

{Y, Y, N, Y, N, Y, N} sfi cost: 4
{Y, Y, Y, N, N, Y, N} cp cost: 4
{Y, Y, N, N, Y, Y, N} gs cost: 4
{Y, N, Y, Y, N, Y, Y} c cost: 4
{Y, N, N, Y, Y, Y, N} gs cost: 4
{N, Y, Y, N, Y, Y, N} gs cost: 4
{N, Y, N, N, N, N, N} ewh cost: 3

Closed list:

{N, N, N, N, N, Y, N} cost: 1
{Y, N, N, N, N, Y, N} lur cost: 2
{N, Y, N, N, N, Y, N} luph cost: 2
{N, N, N, N, Y, Y, N} gs cost: 2
{Y, Y, N, N, N, Y, N} luph cost: 3
{Y, N, N, Y, N, Y, N} sfi cost: 3
{Y, N, N, N, Y, Y, N} gs cost: 3
{N, Y, Y, N, N, Y, N} cp cost: 3
{N, Y, N, N, Y, Y, N} gs cost: 3

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}

A* GOAP Example

{N, Y, N, N, N, N, N} ewh cost: 3

Popped Goal!

Closed list:

{N, N, N, N, N, Y, N} cost: 1

{Y, N, N, N, N, Y, N} lur cost: 2

{N, Y, N, N, N, Y, N} luph cost: 2

{N, N, N, N, Y, Y, N} gs cost: 2

{Y, Y, N, N, N, Y, N} luph cost: 3

{Y, N, N, Y, N, Y, N} sfi cost: 3

{Y, N, N, N, Y, Y, N} gs cost: 3

{N, Y, Y, N, N, Y, N} cp cost: 3

{N, Y, N, N, Y, Y, N} gs cost: 3

Plan: look up phone number, call pizza, eat with hands

Look up recipe (lur)
P: {N, _, _, _, _, _, _}
E: {Y, _, _, _, _, _, _}

Look up phone # (luph)
P: {_, N, _, _, _, _, _}
E: {_, Y, _, _, _, _, _}

Shop for ingredients (sfi)
P: {Y, _, _, N, _, _, _}
E: {_, _, _, Y, _, _, _}

Call Pizza (cp)
P: {_, Y, N, _, _, _, _}
E: {_, _, Y, _, _, _, N}

Cook (c)
P: {Y, _, N, Y, _, _, _}
E: {_, _, Y, N, _, _, Y}

Get silverware (gs)
P: {_, _, _, _, N, _, _}
E: {_, _, _, _, Y, _, _}

Eat w/ silverware (ews)
P: {_, _, Y, _, Y, Y, Y}
E: {_, _, N, _, _, N, _}

Eat w/ hands (ewh)
P: {_, _, Y, _, N, Y, N}
E: {_, _, N, _, _, N, _}