

# Hierarchical Task Network

Dra. M<sup>a</sup> Dolores Rodríguez Moreno

# Objectives

## Specific Objectives

- Understand HTN

## Source

- Stuart Russell & Peter Norvig (2009). Chater 11. Artificial Intelligence: A Modern Approach. (3rd Edition). Ed. Pearsons
- M. Ghallab, D. Nau & P. Traverso. Automated Planning and Acting. (Chapters 2& 3). Cambridge University Press, 2016
- Dana Nau's slides for Automated Planning. Licensed under License <https://creativecommons.org/licenses/by-nc-sa/2.0/>

# Outline

- Motivation
- Definition
- Example
- HTN vs STRIPS
- HTN planners
- Conclusions

# Motivation

- Domain-independent planner: many combinations of vehicles and routes to travel to a destination that's far away
- Experienced human: small number of “recipes” e.g: flying
  1. buy ticket from local airport to remote airport
  2. travel to local airport
  3. fly to remote airport
  4. travel to final destination
- How to enable planning systems to make use of such recipes?

# Outline

- Motivation
- **Definition**
- Example
- HTN vs STRIPS
- HTN planners
- Conclusions

# Definition

- Known as a task reduction or Hierarchical Task Network
- Problems and operators are arranged in a network or  $\{ \}$  of task (called actions) that correspond to transition states
- High-level tasks are reduced into low levels tasks
- A method maps a task in a network of partially ordered tasks with few restrictions
- The algorithm iteratively expands tasks and resolves conflicts until a plan consisting of primitives and free tasks conflict is found

# Outline

- Motivation
- Definition
- **Example**
- HTN vs STRIPS
- HTN planners
- Conclusions

## HTN: Example (I)

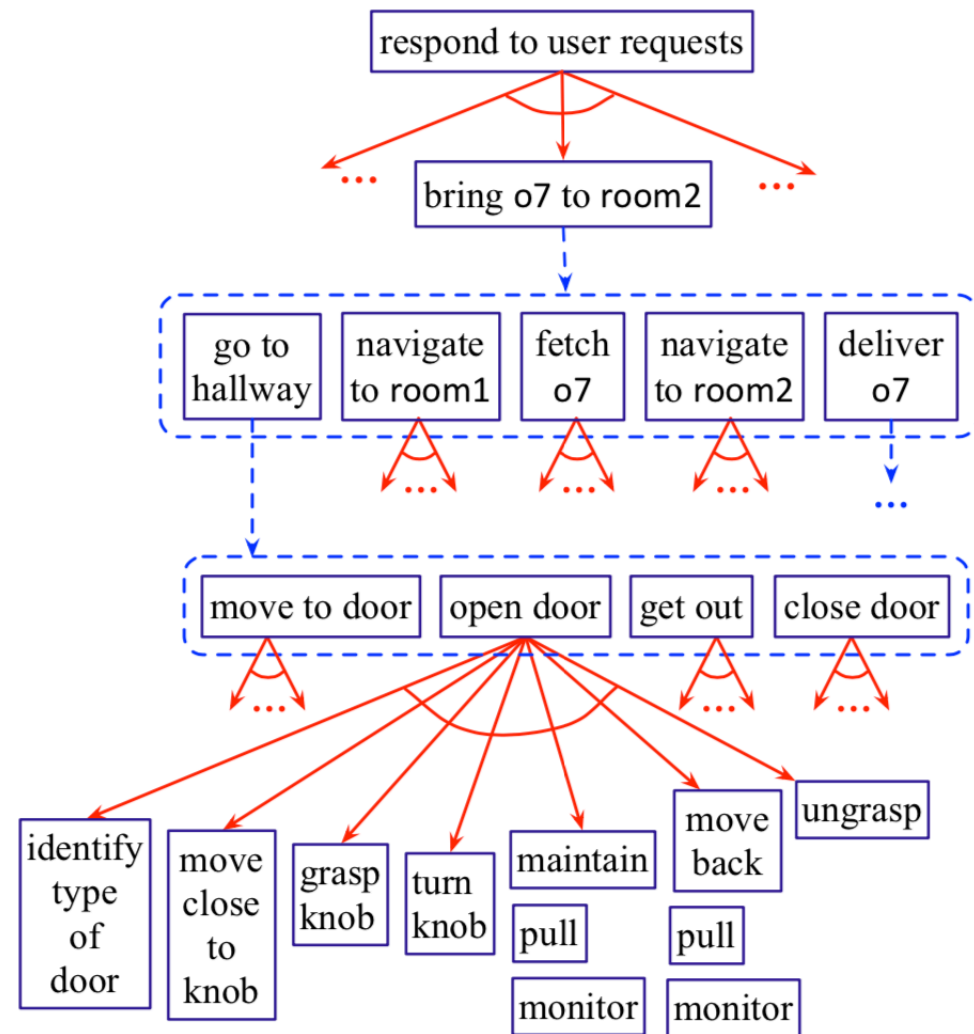
- Want to: bring object o7 to room2
- Might want to specify that the proper way to accomplish this task is as follows:

Have a robot r that fetch o7, navigate to room2 and deliver o7

- A good set of methods can enable an HTN planner to perform well on benchmark problems
- A drawback of this approach is that it requires the domain author to write and debug a potentially complex set of domain-specific recipes (see next figure)



## HTN: Example (II)



# Outline

- Motivation
- Definition
- Example
- **HTN vs STRIPS**
- HTN planners
- Conclusions

# HTN vs STRIPS

- Differences between HTN and STRIPS - style planners lies in what they do and how they plan for:
  - STRIPS: the objective is to find a  $\{$  of ordered actions from the IS to goals. It finds appropriate operators that have the desired effects and making their preconditions, the sub-goals
  - HTN: it plans looking for task network that may include other things besides goals. Plan by decomposing tasks and conflict resolution. Methods should contain all possible ways to get tasks, which is much more tedious

# Outline

- Motivation
- Definition
- Example
- HTN vs STRIPS
- **HTN planners**
- Conclusions

# HTN planners

- NONLIN
- O-PLAN
- DEVISER
- SIPE
- NMRA
- SHOP: plans the tasks in the order they are executed. Domain Independent
- TALPlanner (Temporal Action Logic planner) domain dependent

# Outline

- Motivation
- Definition
- Example
- HTN vs STRIPS
- HTN planners
- **Conclusions**

## Conclusions

- HTN planners can encode “recipes” as collections of methods and operators
- Writing a knowledge base can be more complicated
- Specify standard ways of solving problems
- Can speed up planning by many orders of magnitude (e.g., polynomial time versus exponential time)