规划问题求解（AI Planning）

毛文吉

中国科学院自动化研究所

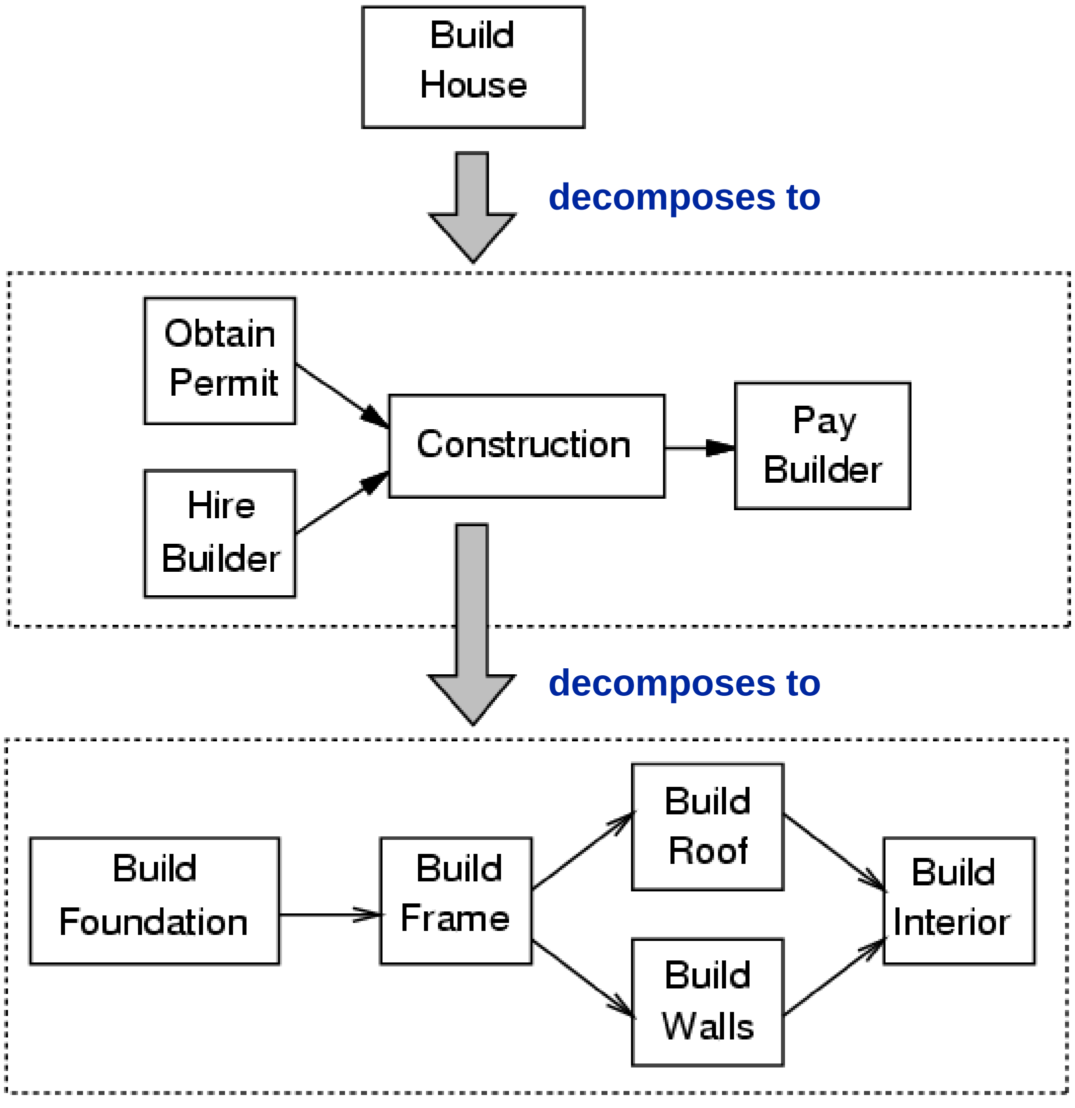
Outline

- STRIPS-like plan representation
- Planning with state-space search
- Partial-order planning (POP)
- Planning graphs (GraphPlan)
- Hierarchical task network (HTN) planning
- Probabilistic planning

Outline

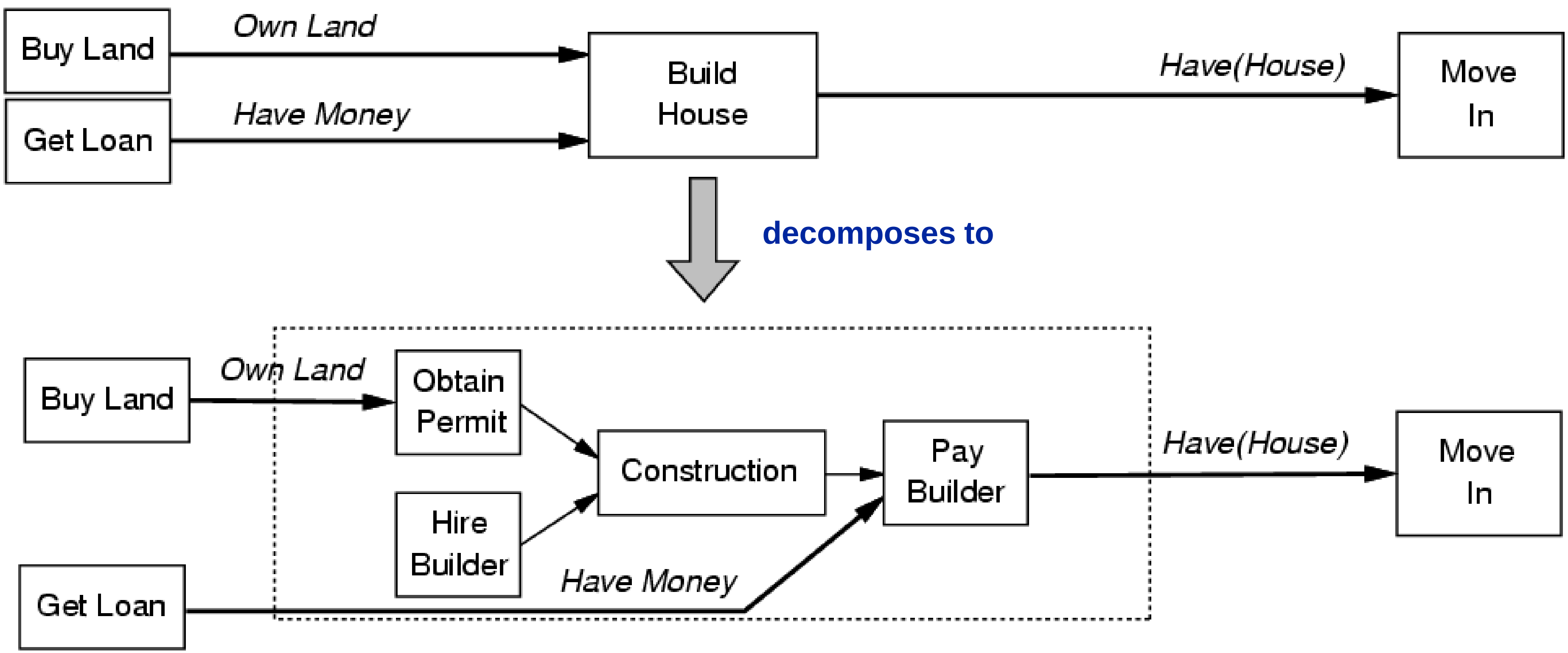
- STRIPS-like plan representation
- Planning with state-space search
- Partial-order planning (POP)
- Planning graphs (GraphPlan)
- Hierarchical task network (HTN) planning
- Probabilistic planning

Hierarchical decomposition of actions



Task reduction

从抽象到具体.



- Naturally encode hierarchical decomposition of actions
- Action hierarchy as domain-specific task knowledge

HTN（层次任务网络）planning

Hierarchical task network (HTN) planning [Erol et al 94]

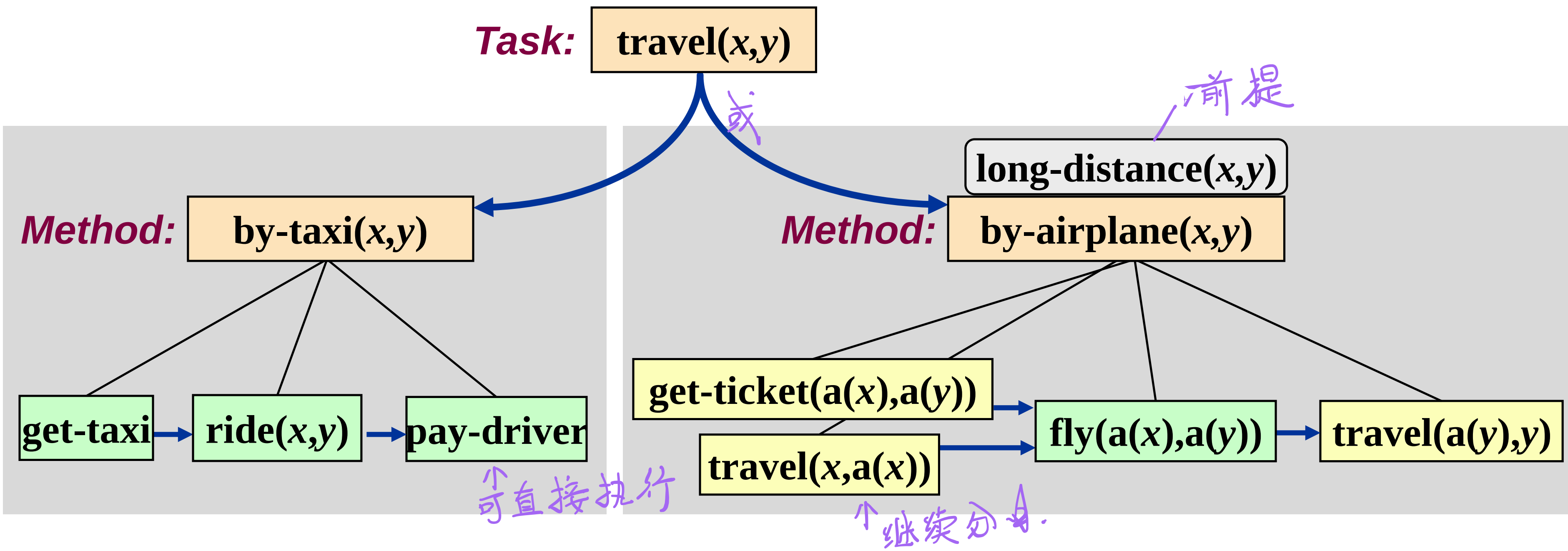
构建 层次结构.

- Capture hierarchical structure of the planning domain
- Planning domain contains non-primitive actions and schemas for reducing them

  抽象动作.

- Reduction schemas: 主观性比较大
  - Given by the designer
  - Describe preferred ways to accomplish a task

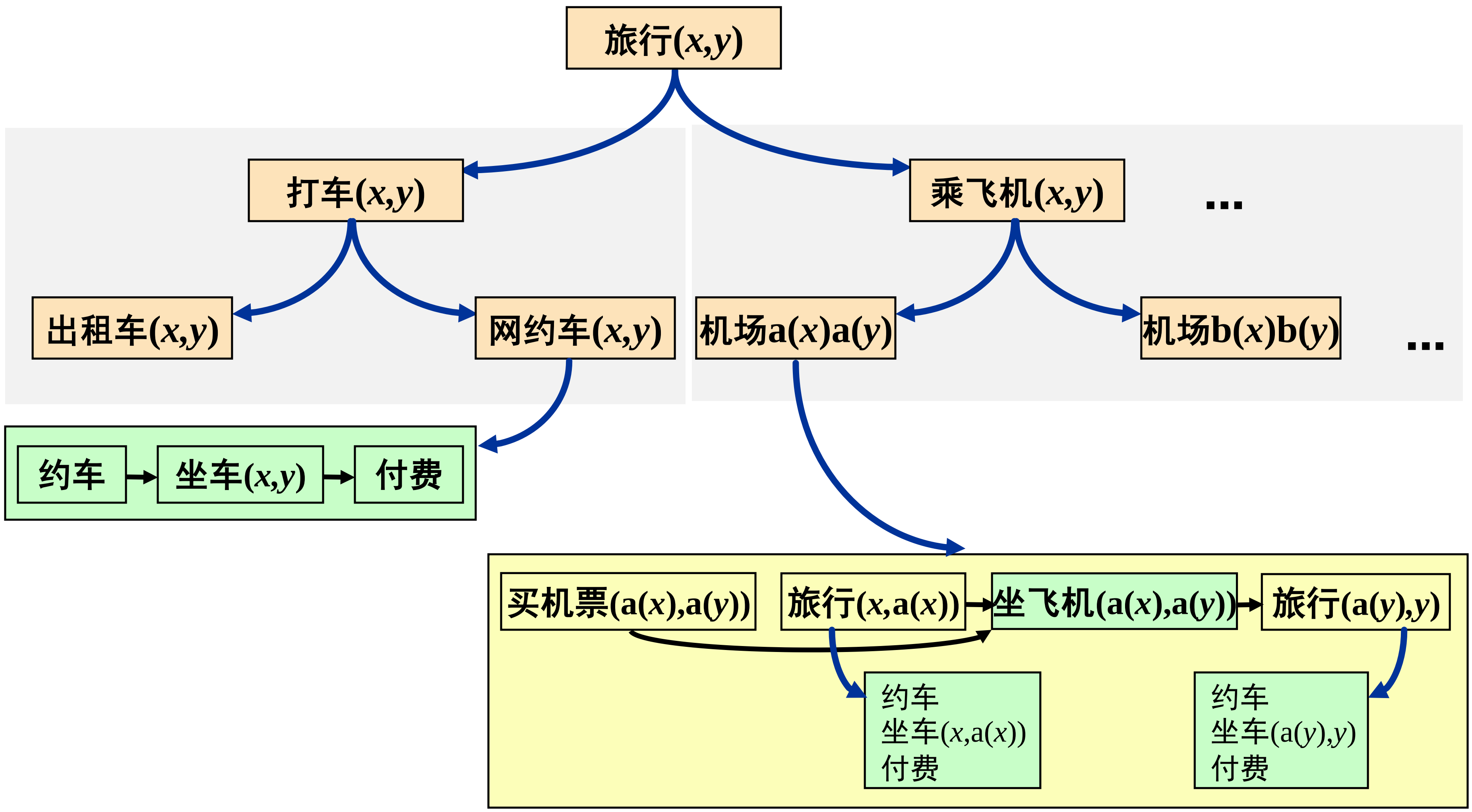
Task network example



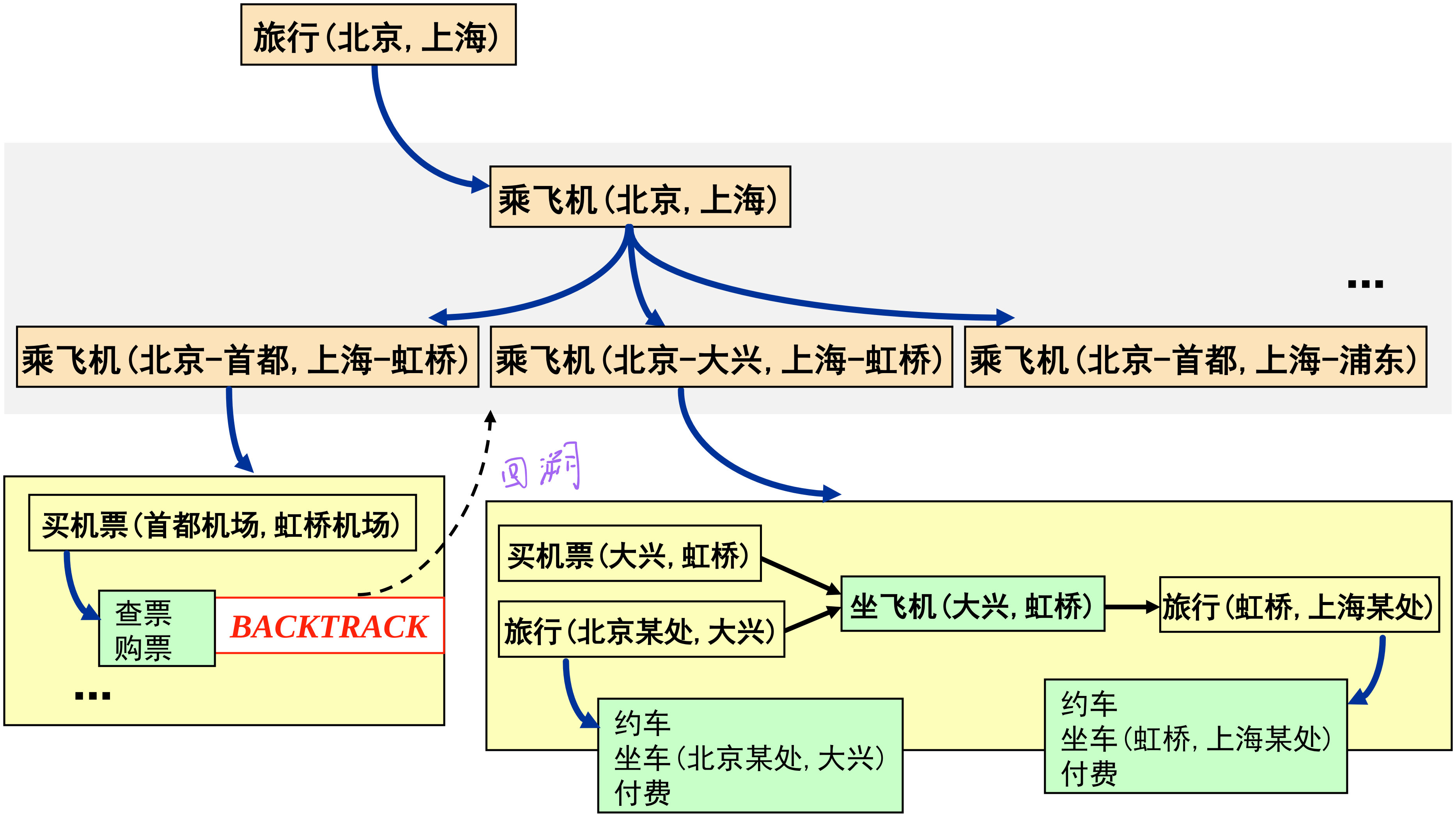
- *Tasks* rather than goals

  (任务分解. 不是目标分解)
  - *Primitive*（原始的）vs *nonprimitive*（抽象的）actions
- *Methods* describe task reduction
  - Ways to decompose tasks into subtasks

Task network example



Task network example



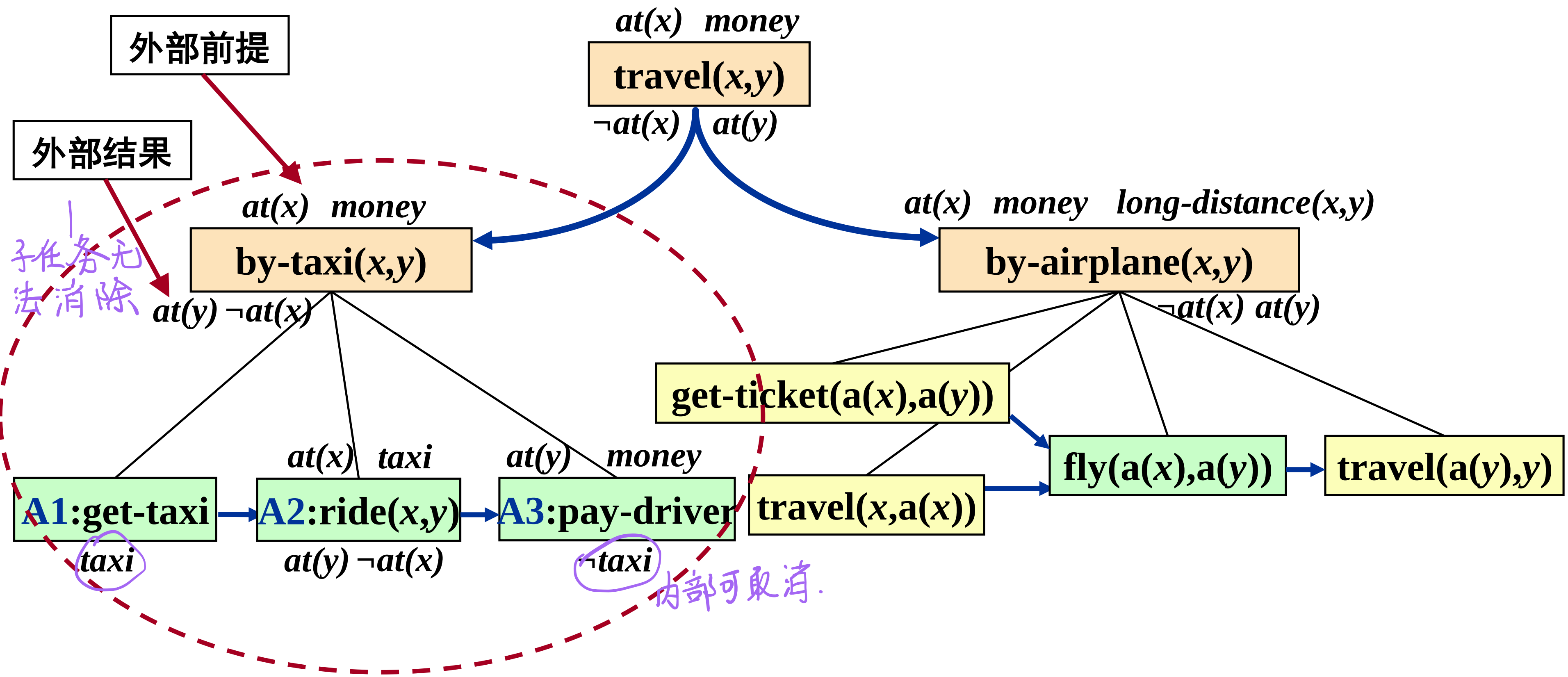
HTN planning intuition

Problem reduction:

- Decompose tasks into subtasks 分解任务为子任务.
- Handle constraints 处理冲突
- Resolve interactions 解决交互问题
- If necessary, backtrack and try other decompositions

如有必要，需要回溯和尝试其他分解.

Task network example

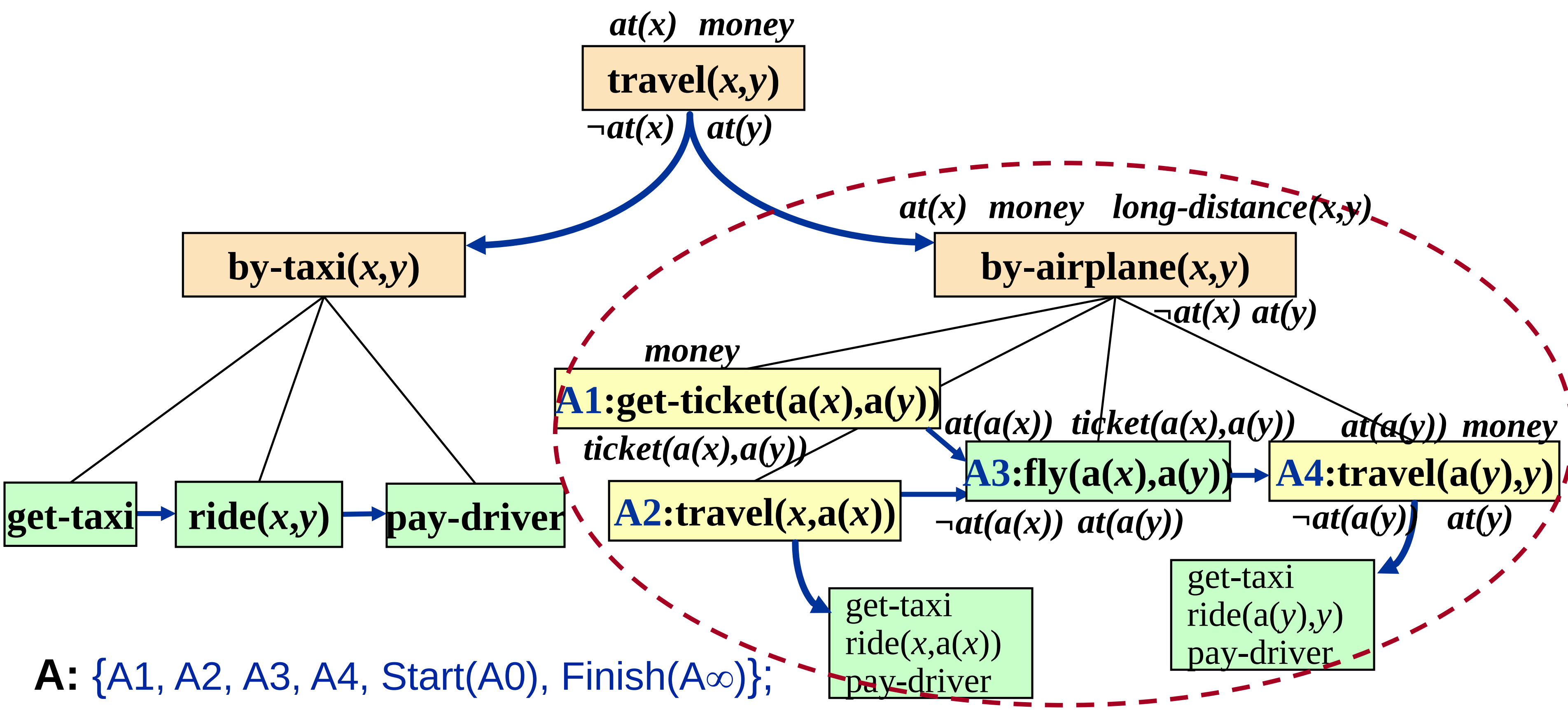


Primitive actions: {A1, A2, A3, Start(A0), Finish(A∞)};

Orderings: {A1<A2, A2<A3}

Causal links: {(A0 $\xrightarrow{\text{at(x)}}$ A2), (A0 $\xrightarrow{\text{money}}$ A3), (A1 $\xrightarrow{\text{taxi}}$ A2), (A2 $\xrightarrow{\text{at(y)}}$ A3), (A2 $\xrightarrow{\text{at(y)}}$ A∞)}

Task network example



A: {A1, A2, A3, A4, Start(A0), Finish(A∞)};

O: {A1<A3, A2<A3, A3<A4}

L: {(A0 $\xrightarrow{\text{money}}$ A1), (A0 $\xrightarrow{\text{at(x)}}$ A2), (A0 $\xrightarrow{\text{money}}$ A2), (A0 $\xrightarrow{\text{money}}$ A4), (A1 $\xrightarrow{\text{tkt(a(x),a(y))}}$ A3), (A2 $\xrightarrow{\text{at(a(x))}}$ A3), (A3 $\xrightarrow{\text{at(a(y))}}$ A4), (A4 $\xrightarrow{\text{at(y)}}$ A∞)}

Basic HTN Procedure

1. Input a planning problem P
2. If P contains only *primitive* tasks, then resolve the conflicts and return *solution*; If the conflicts cannot be resolved, return *failure*

total ordering primitives satisfying constraints

子任务分解

3. Choose a *non-primitive* task *t* in P
4. Choose an expansion for *t*
5. Replace *t* with the expansion
6. Find interactions among the tasks in P and suggest ways to handle them
7. Choose one way and apply it
7. Go to step 2

Find conflicts and suggest ways to resolve them

Properties of HTN planning

形式化语义，表达能力更强

Sound, *complete* and has formal semantics [Erol et al 94]

- Strictly more expressive than classical planning
- Encode expert knowledge essential to task reasoning 使用了领域知识
- Handle actions and propositions with variables

The idea of *task decomposition* is critical

- Efficient ways to represent complex task knowledge
- Practical planner applied to real-world problems 和真实世界类似
- Critical applications in large-scale intelligent systems

在大规模智能系统有重大应用.

Outline

- STRIPS-like plan representation
- Planning with state-space search
- Partial-order planning (POP)
- Planning graphs (GraphPlan)
- Hierarchical task network (HTN) planning
- ➢ Probabilistic planning

Classical planning

- Most advances seen *first* in classical planning

  很多实际环境不满足
- Many practical environments don't satisfy classical planning *assumptions*
  - Possible to handle *minor* assumption *violations* through re-planning 重规划 and execution monitoring 监测
  - Advantageous to rely on *widely-used* (and efficient) classical planning techniques

  有指导意义
- Classical planning techniques often shed light on the effective ways of handling *non-classical* planning problems
  - Most of the efficient techniques for handling non-classical scenarios are based on classical ideas/advances

Classical planning assumptions

- Discrete time 时间离散
- All effects are immediate 结果实时.
- Fully observable environment 完全可观测环境
- Deterministic effects 确定性结果.
- No dynamic changes in plan execution 在规划执行过程中没有动态变化
- Planning agent is the sole agent of changes 单智能体变化

…… ……

Classical planning assumptions

- Discrete time
- All effects are immediate
- **Fully observable environment**
- **Deterministic effects**
- **No dynamic changes in plan execution**
- Planning agent is the sole agent of changes

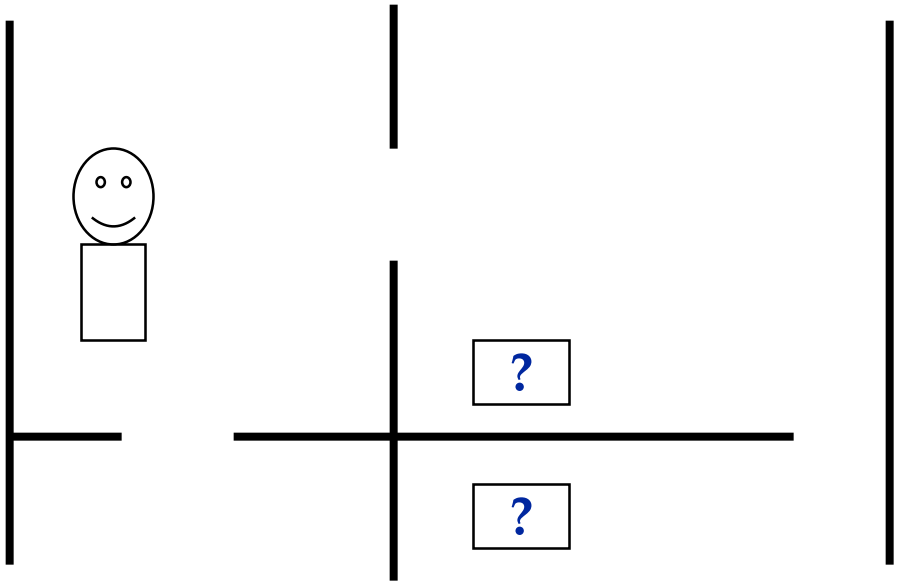
Uncertainty

… …

Sources of uncertainty

不能观测所有状态

When we try to execute plans, uncertainty from several different sources can affect plan success.

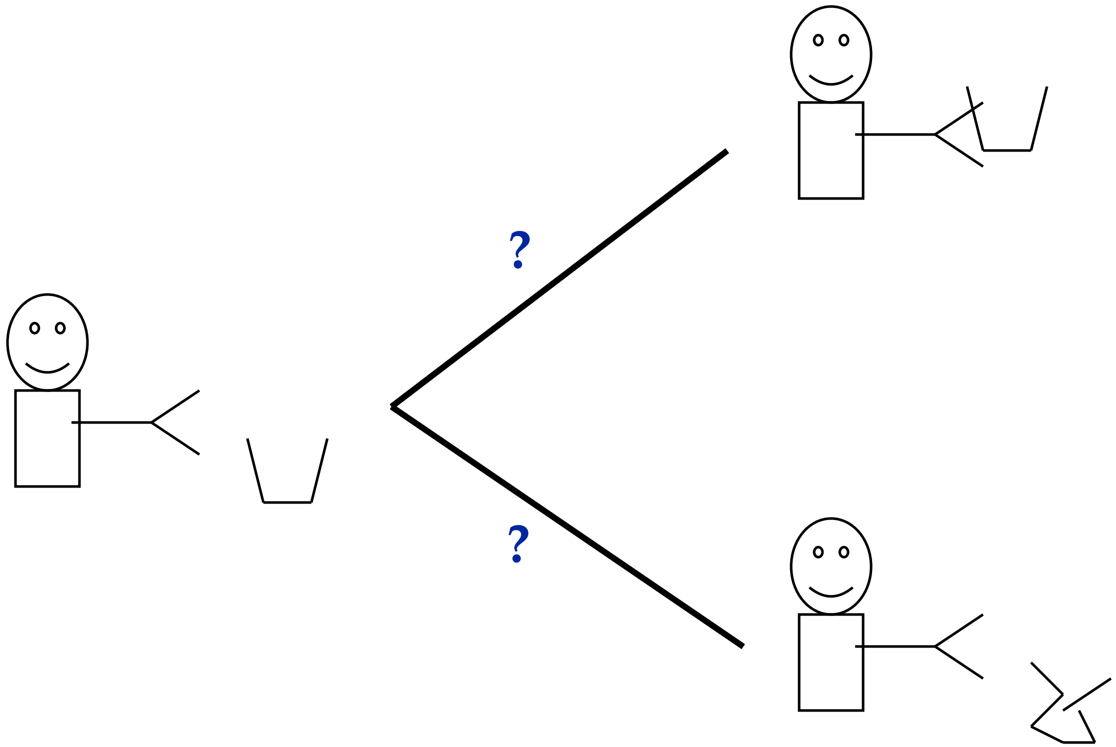


Firstly, we might have uncertainty about the *state of the world*.

Sources of uncertainty

结果不确定

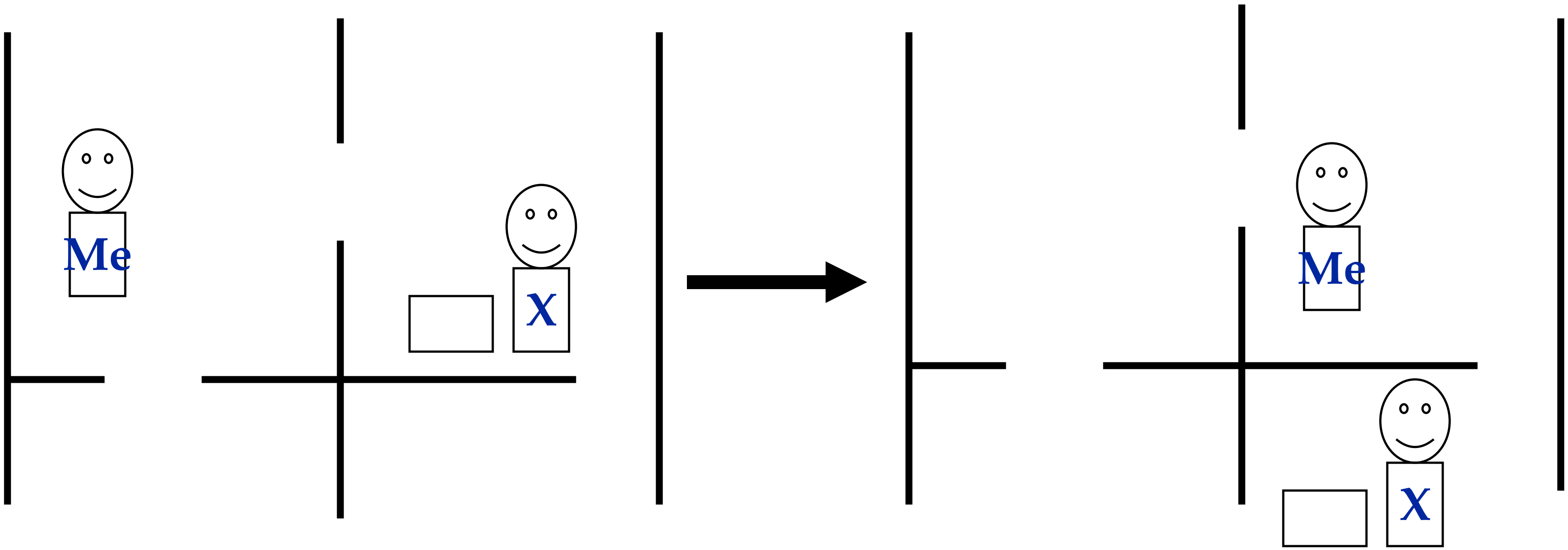
Actions we take might have *uncertain effects* even when we know the world state.



Sources of uncertainty

其他 agent 会改变状态，

External agents might be changing the world while we execute our plan.



重规划

Dealing with uncertainty: Re-planning

- Make a plan assuming nothing bad will happen
- Monitor for problems during execution
- Build a *new plan* if a problem is found
  - Either re-plan to the goal state
  - Or try to amend the existing plan

①无法实现实时监测

②结果可能无法即时挽回.

条件规划

Dealing with uncertainty: Conditional planning

- Deal with *contingencies* (偶发事情、意外) at planning time before they occur
- Utilize *conditional planning*（条件规划）to address every possible contingency in the policy

E.g. if <*test1*> then *plan_A*
else if <*test2*> then *plan_B*
else if <*test3*> then *plan_C*
else …

无法考虑所有情况

Trade-offs in strategies for uncertainty

可能来不及

- Re-planning: Can't predict the plan steps are needed *before* the contingency is discovered
  - Re-planning agent: "Why are you taking an umbrella? It's not raining!"

考虑不了所有情况

- Conditional planning: Impossible to plan for *every* contingency
  - Conditional planning agent: "Why are you leaving the house? Class may be cancelled. It might rain. You might have won the lottery. Was that an earthquake?…"

只考虑比较重要的情况

Probabilistic planning: The middle ground

- Partial knowledge about *uncertainty*: different contingencies have different probabilities of occurring
- Plan ahead for *likely* contingencies that may need steps taken before they occur
- Use probability theory to judge plans that address some contingencies:

  seek a plan that is above some minimum probability of *success*

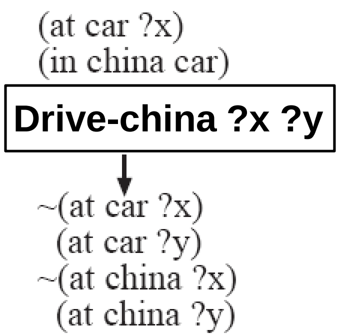
Some issues to think about

- How to represent the occurrence of *likely* contingencies in action representation?
- How do we figure out the probability of *plan success*? What are the representations to support this?
- Can we distinguish bad *outcomes* (e.g. not holding the glass) from *really* bad outcomes (e.g. broken the glass, spilled the sulfuric acid..)? 后果有多严重 (效用价值)

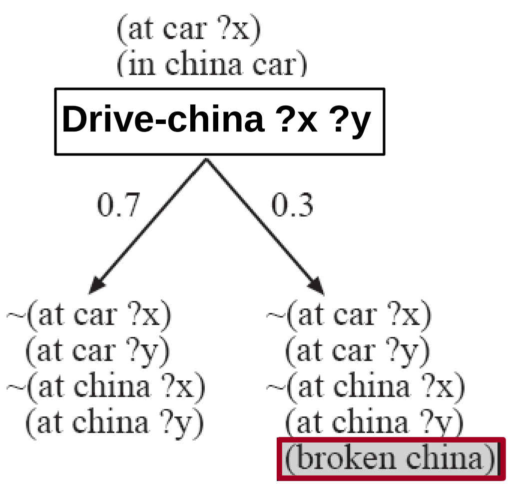
Some issues to think about

- How to represent the occurrence of *likely* contingencies in action representation?
- How do we figure out the probability of *plan success*? What are the representations to support this?
- Can we distinguish bad *outcomes* (e.g. not holding the glass) from *really* bad outcomes (e.g. broken the glass, spilled the sulfuric acid..)?
  - Probabilistic action representation
  - Probability of plan success
  - Plan utility

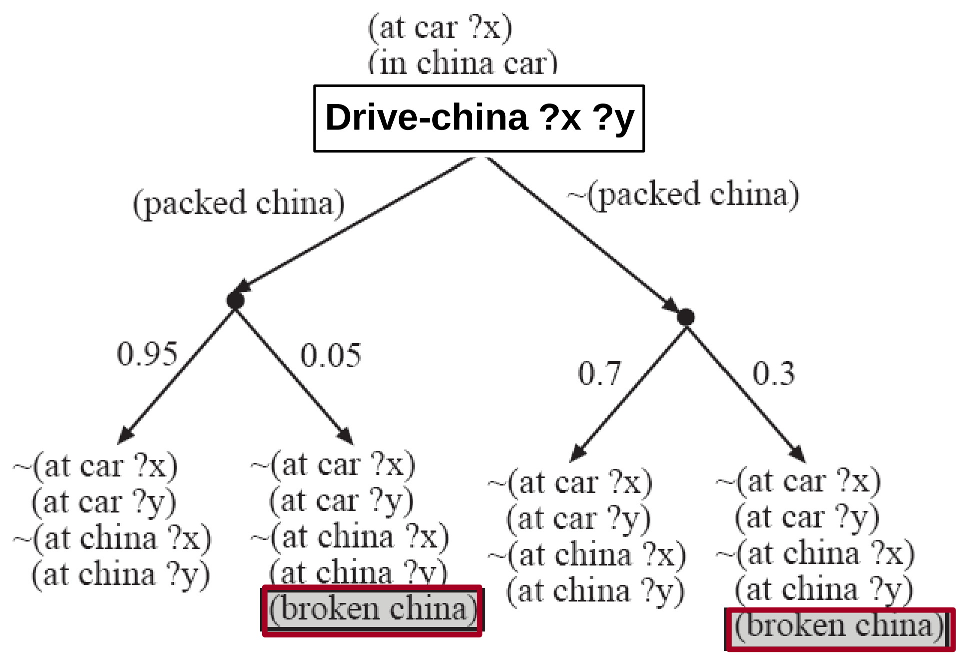
Representing actions with uncertain outcomes



Deterministic effects



Non-deterministic effects

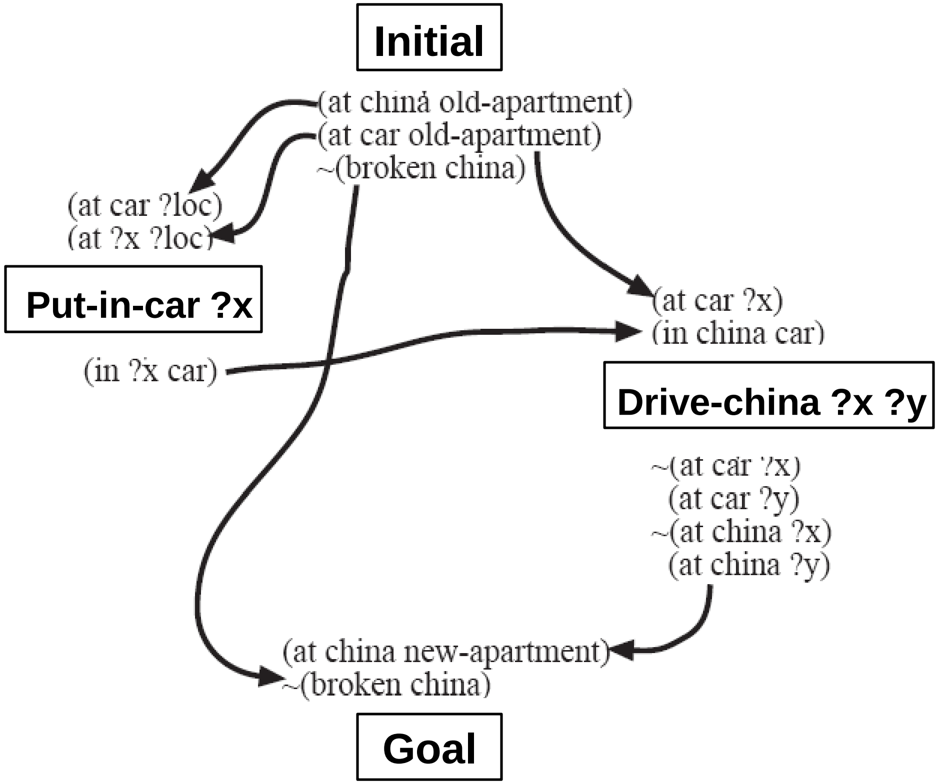


Non-deterministic effects with conditional probabilities

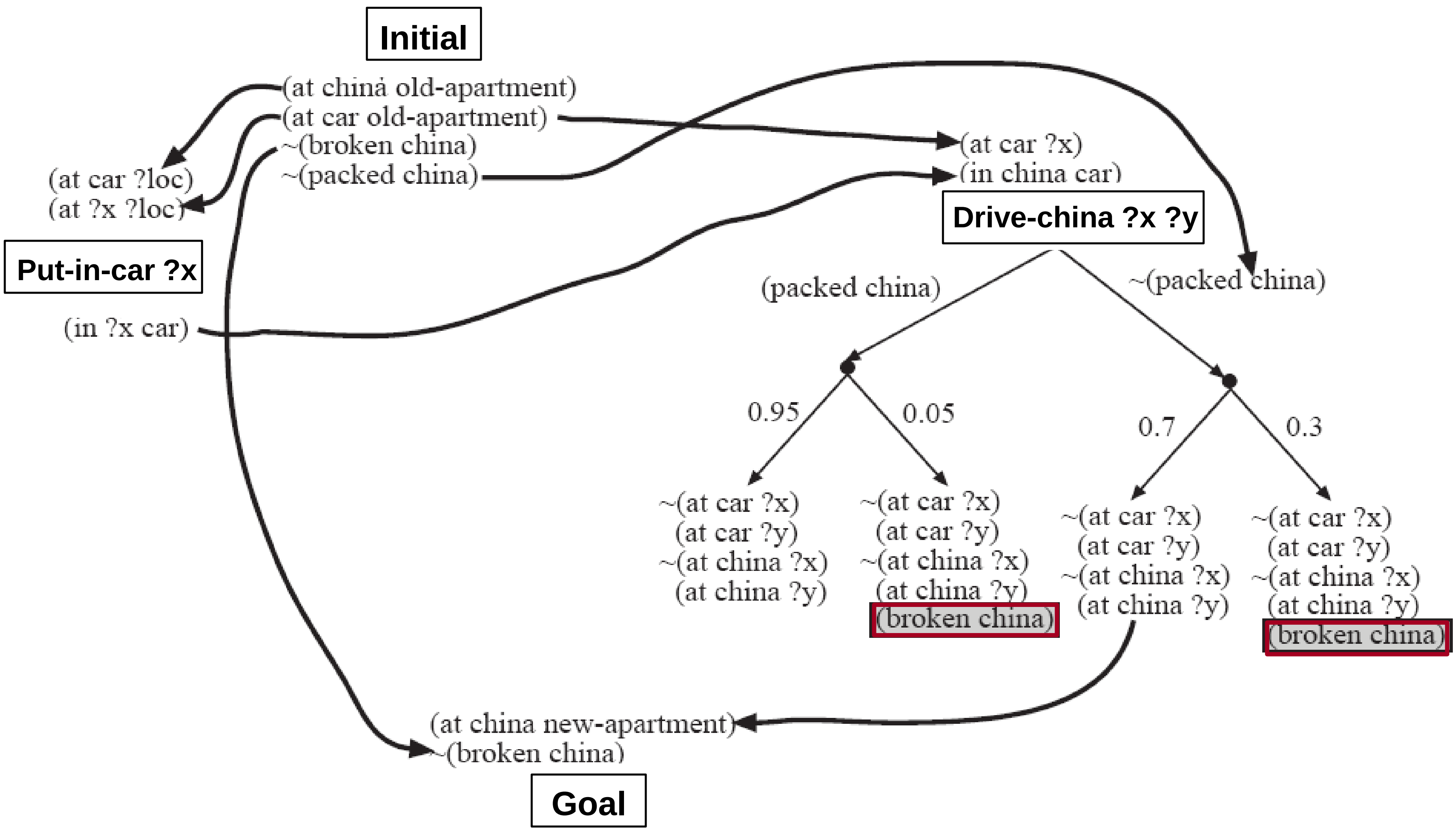
“~”表示否定，同“¬”

A partial-order plan to move china

- An POP-based planner might come up with this plan for deterministic action representation:



A plan that works 70% of the time…



Recall: POP algorithm

POP((A, O, L), agenda, actions)

Initial plan: {*Start*, *Finish*} and preconditions in *Finish* as open conditions

1. If **agenda** is empty, then **return** (A, O, L) 结束条件
2. Pick (Q, A_{need}) from **agenda** （子）目标
3. **Choose** an action A_{add} that adds effect Q 动作选择
   - If no such action exists, **fail**
   - Add the link $A_{add} \xrightarrow{Q} A_{need}$ to **L** and the ordering A_{add}<A_{need} to **O**
   - If A_{add} is new, add it to **A** 规划扩充
4. Remove (Q, A_{need}) from **agenda**. If A_{add} is new, for each of its preconditions P add (P, A_{add}) to **agenda** 更新（子）目标
5. For every action A_t in A that threatens any causal link $A_p \longrightarrow A_c$ in **L**
   - **Choose** to add A_t<A_p or A_c<A_t to **O**
   - If neither choice is consistent, **fail**
6. POP((A, O, L), **agenda**, actions)

保护因果连接：
- 降级(Demotion): $\mathbf{A_t < A_p}$
- 升级(Promotion): $\mathbf{A_c < A_t}$

Modifications to the UCPOP algorithm

- Allow more than one causal link for each condition in the plan
- Confront a threat by decreasing the probability of an outcome in which the threatened condition is negated
- Terminate when sufficient probability reached (may still have threats)

Finding plans for moving china

- **Goals:**

  (at china new-apartment)

  ~(broken china)

- **Initial states:**

  (at china old-apartment)

  (at car old-apartment)

  ~(broken china)

  ~(packed china)

- **Actions:**

  (at car ?loc)
  (at ?x ?loc)

  Put-in-car ?x

  (in ?x car)

  (at car ?x)
  (in china car)

  Drive-china ?x ?y

  (packed china)

  ~(at car ?x)
  (at car ?y)
  ~(at china ?x)
  (at china ?y)
  (broken china) 0.05

  ~(packed china)

  ~(at car ?x)
  (at car ?y)
  ~(at china ?x)
  (at china ?y)
  (broken china) 0.3

  ~(packed china)

  Pack-china

  (packed china)

- **Plans:**

  Start → {Pack-china, Put-in-car china} → Drive-china → *Finish* (Prob: 0.95)

  Start → Put-in-car china → Drive-china → *Finish* (Prob: 0.70)

  … …

概率无法衡量后果的严重性

DRIPS: Decision-theoretic Refinement Planner

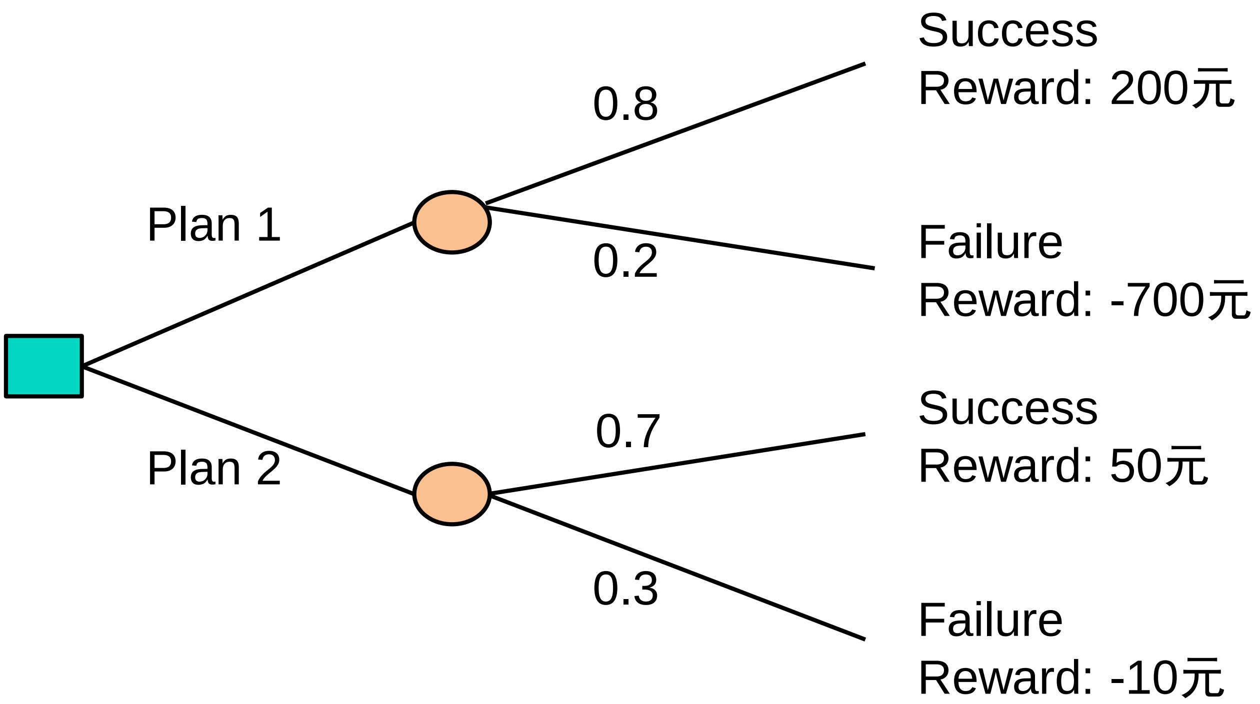
- Consider plan *utility*, taking into account action costs and benefits of different states
- Search for a plan with *Maximum Expected Utility* (*MEU*), not just above a threshold
- A skeletal planner, makes use of utility ranges of alternative plans in order to search efficiently

剪枝

- Prune alternatives whose utility ranges are completely below the range of some *dominated plan*

Example of decision theory

- People are utility maximizers



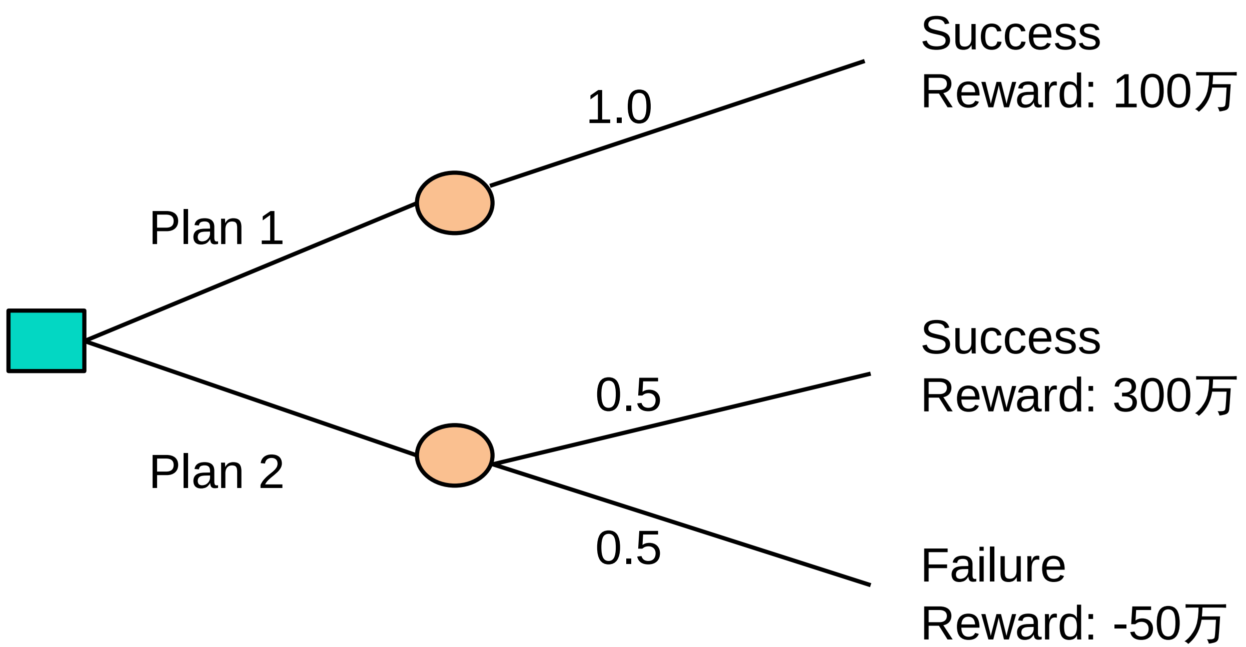
EU(Plan1)=
200*0.8-700*0.2
=20

EU(Plan2)=
50*0.7-10*0.3
=32 ✓

Decision node: Agent plays

Chance node: Nature plays

Example of decision theory



EU(Plan1)=100*1

EU(Plan2)=
300*0.5-50*0.5
=125

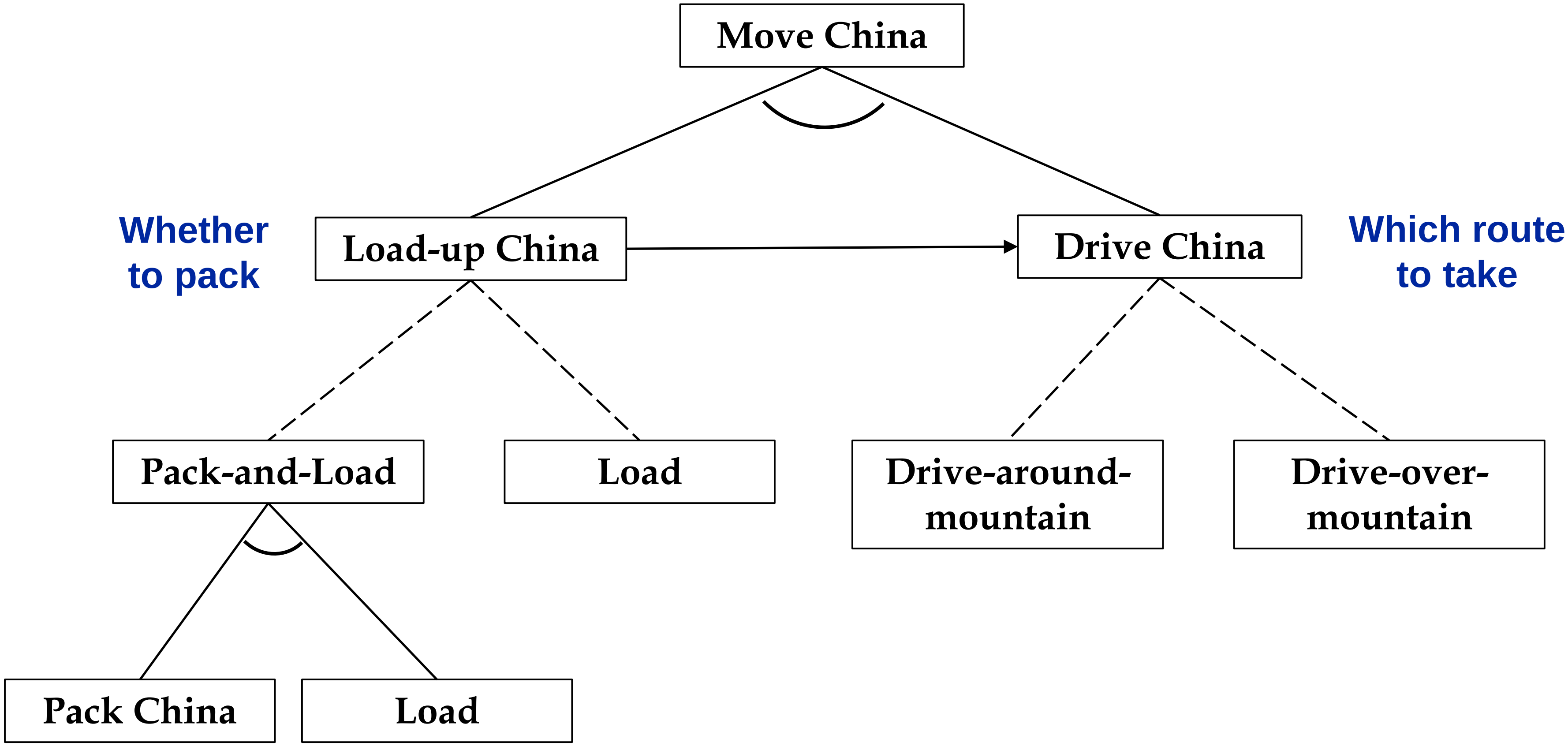
货币不一定等价于效用、

- Money is not necessarily equal to preferences …
  - Utility theory does not assume money is people’s utility function
  - People assign utility to money. Different people have different utility functions

Abstract plan for moving china

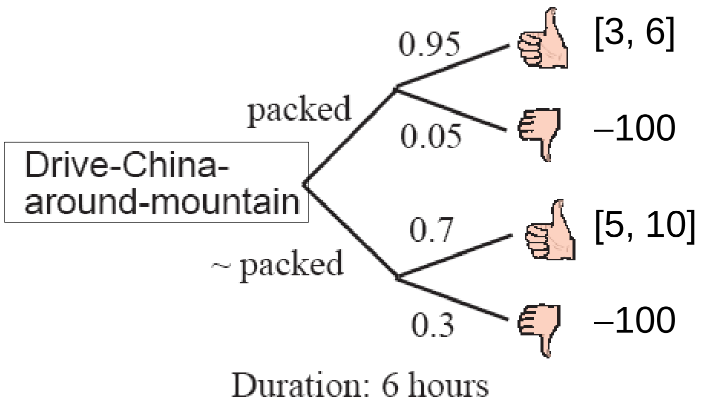
HTN

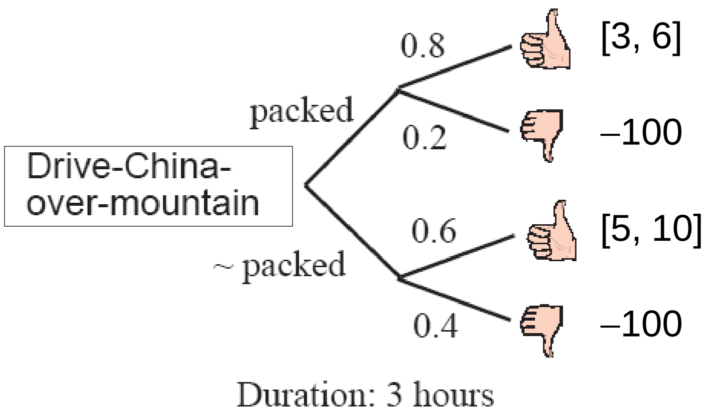
- Use an *abstraction-decomposition network* to describe all possible specializations of an abstract plan to move china

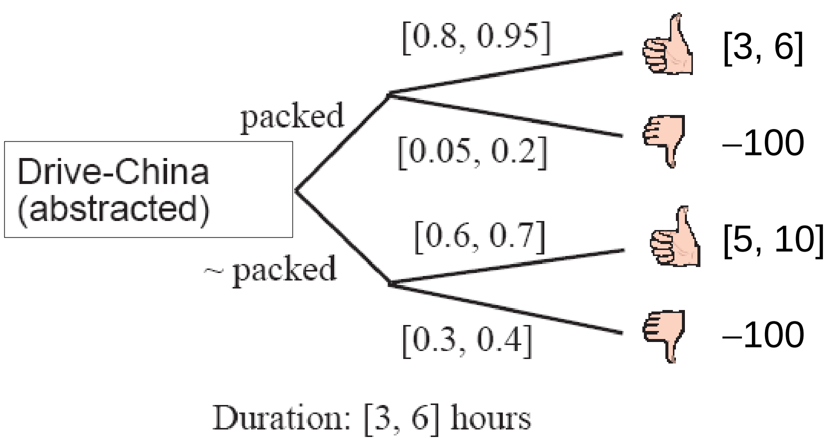


Utilities of action outcomes

- Use utility to represent the costs / benefits of action *outcomes*







Refining abstract action Load-up-china

- Use utility ranges to compute the *dominance* between *alternative* plans

Plan A: Pack and load china

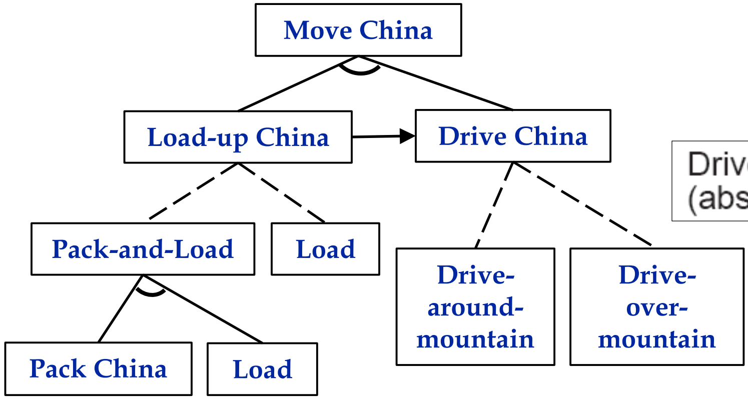
Outcome	Duration	Utility	Probability
Not broken	[3 6]	[3 6]	[0.8 0.95]
Broken	[3 6]	–100	[0.05 0.2]

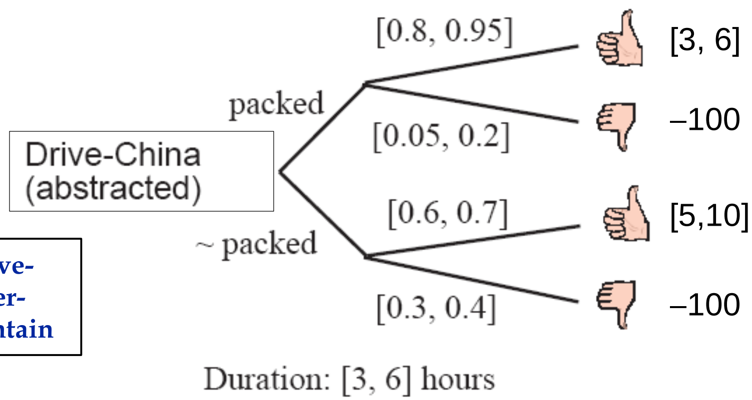
Plan utility:
[-20+2.4, -5+5.7]

Plan B: Load china, don't pack

Outcome	Duration	Utility	Probability
Not broken	[3 6]	[5 10]	[0.6 0.7]
Broken	[3 6]	–100	[0.3 0.4]

Plan utility:
[-40+3, -30+7]





Other approaches

MaxPlan (Inspired by SATPLAN):

- Given a boolean formula with variables (either *choice* variables or *chance* variables), find an assignment to the choice variables that maximizes the probability that the formula is true
- We have control to *choice* variables:

  e.g. which actions to use
- We cannot control *chance* variables:

  e.g. the weather, the outcome of each action …
- Then use standard algorithm to compute and *maximize* probability of *success*

Other approaches

MDPs and POMDPs: 大规模、维数爆炸

- MDP-based approaches can find optimal solutions and deal elegantly with costs
- However,
  - They use rather compact action representation
  - Need to control potential state space explosion
  - Complexity issue and tractability

Recent trends in planning research

Trends:

与RL结合,

- Combining *planning* with *learning* technique
  - Bridging the Gap between AI Planning and Reinforcement Learning (PRL) Workshop @ *ICAPS 2020-2023*
  - Bridging the Gap between AI Planning and Reinforcement Learning (PRL) Workshop @ *IJCAI 2022*

可解释 AI planning.

- *Explainable* automated *planning* and decision making
  - Explainable AI Planning (XAIP) Workshop @ *ICAPS 2018-2022*
  - Explainable AI (XAI) Workshop @ *IJCAI 2017-2020*, *2022*
  - Explainable AI Workshop 2021 Organized by *NIST*
  - Explainable and Transparent AI and MAS (EXTRAAMAS) Workshop @ *AAMAS 2019-2023*

小结：关于规划

偏序规划（POP）
图规划（GraphPlan）
约束满足规划（SATPlan）
时序规划（TLPlan）

Classical planning

Hierarchical extension

层次任务网络规划（HTN）

Probabilistic extension

概率规划

Probabilistic planning

Multi-agent extension

多智能体规划

Multi-Agent planning

Multi-agent planning

结合自身特长，处理其他 Agents 未解决的目标

- *Collective* effort of multiple agents to combine their knowledge, information and capabilities so as to develop solution(s) to the planning problem that each agent could not have solved as well alone

合作情况下

- Assume that agents are *cooperative*: each agent must formulate plans for what it will do that take into account (sufficiently well) the plans of other agents

除了自己目标，考虑其他 agent 或集体目标

What aspects are multi-agent?

- Multi-agent planning could refer to just the *results* of the planning process:

  A centralized planning process formulates plans that specify how each of the multiple agents should do

  - Centralized planning for distributed plans 分发任务

- Multi-agent planning could refer to the *process* of formulating plans:

  Multiple agents participate in the construction of a single joint plan

  - Distributed planning for centralized plans 部分规划合并为单一规划

What aspects are multi-agent?

- Both the planning process and results are multi-agent:
  - 局部经验
  - Each agent applies its local expertise and awareness to develop its local plan
  - 交流，共享知识，使整体更有效律
  - Agents use communication and/or shared knowledge to shape their local plans and conform better to others' plans, in order to work together effectively
  - ➢ Distributed planning for distributed plans

Levels of working together

	As a process	As a result
最基本 Coordination 协调	没有集体利益，只是考虑进来 Decision-making takes others into account	Individuals are not thwarted by others
Cooperation 合作	Individual decisions further the collective welfare	Individuals appear to be “working together”
Collaboration /Teamwork 团队任务.	Individuals work toward shared goals 共享目标	Common goals are achieved well and/or efficiently

Coordinating multiple agents 契约

- Social conventions（协定）
- Flocking behavior in nature
  - Cohesion（凝聚）: get closer to the average position of neighbors
  - Separation（分离）: avoid getting too close to any one neighbor
  - Alignment（对齐）: get closer to the average heading of neighbors
- Agent communication
- Agent modeling
  - Build mind model of other agents 建立其他 agent 心智模型
  - Intention/goal recognition 行为意图识别

Agent modeling

- Representing and reasoning about *other* agents
  - Other agents' capabilities, skills, responsibilities
  - Other agents' mental states, plans, goals, actions
  - Other teams' goals, plans, states

    …
- *Explicitly* model other agents
  - *Implicit* modeling if all coordination with other agents compiled in without actual representation and reasoning about other agents

Modeling other agents: A critical technique

- In dynamic, complex, multi-agent environments:
  - Agents respond unpredictably to the environment
  - Interactions with other agents not always feasible

    E.g. communications failures, opponent behavior, etc
  - State of other agents not known at design time
- Agents must model other agents at *execution time*
  - To detect failures in their execution (leading to diagnosis, recovery)
  - To coordinate with others, render assistance
  - To visualize, ascertain progress
  - To compete against other agents

  ...

Plan recognition（规划/意图识别）

- *Key* agent modeling technique
  - Infer other agent's current plans and goals based on the available *observations*
  - Other agents may be unable to *communicate* directly for a variety of reasons
- Major challenge: *Ambiguity* 歧义.
  - There may be multiple plans to explain the observed actions

Resources

- UCPOP Planner:

  https://www.swmath.org/software/20687
- GraphPlan Planner:

  https://en.wikipedia.org/wiki/Graphplan (External links)
- UMCP Planner:

  http://www.cs.umd.edu/projects/plus/umcp
- SHOP/SHOP2 Planner:

  http://www.cs.umd.edu/projects/shop
- Planning Domain Definition Language (PDDL):

  https://planning.wiki/guide/whatis/pddl

References

- R. Fikes and N. Nilsson. STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. *Artificial Intelligence*, 2(3-4), 1971
- D. Weld. An Introduction to Least-Commitment Planning. *AI Magazine*, 15(4), 1994 (POP)
- J. S. Penberthy and D. Weld. UCPOP: A Sound, Complete, Partial-Order Planner for ADL. *Proceedings of KR*, 1992
- A. Blum and M. Furst. Fast Planning Through Planning Graph Analysis. *Artificial Intelligence*, 90(1-2), 1997 (GraphPlan)
- K. Erol, J. Hendler and D. S. Nau. UMCP: A Sound and Complete Procedure for Hierarchical Task-Network Planning. *Proceedings of AIPS*,1994 (HTN)
- D. S. Nau, Y. Cao, A. Lotem and H. Munoz-Avila. SHOP: Simple Hierarchical Ordered Planner. *Proceedings of IJCAI*, 1999

References

- J. Hoffmann. FF: The Fast-Forward Planning System. *AI Magazine*, 22(3), 2001
- N. Kushmerick, S. Hanks and D. Weld. An algorithm for probabilistic planning. *Artificial Intelligence*, 76(1-2), 1995 (BURIDAN)
- P. Haddawy and M. Suwandi. Decision-Theoretic Refinement Planning Using Inheritance Abstraction. *Proceedings of AIPS*, 1994 (DRIPS)
- S. M. Majercik and M. L. Littman. MaxPlan: A new approach to probabilistic planning. *Proceedings of AIPS*, 1998
- J. Blythe. Decision-Theoretic Planning. *AI Magazine*, 20(2), 1999
- E. H. Durfee. Distributed Problem Solving and Planning. In M. Luck, V. Marik, O. Stepankova, and R. Trappl (eds.), *Multiagent Systems and Applications* (LNAI 2086). Springer, 2001

End.