Logic-based artificial intelligence

Jiří Kléma

Department of Computer Science,
Czech Technical University in Prague

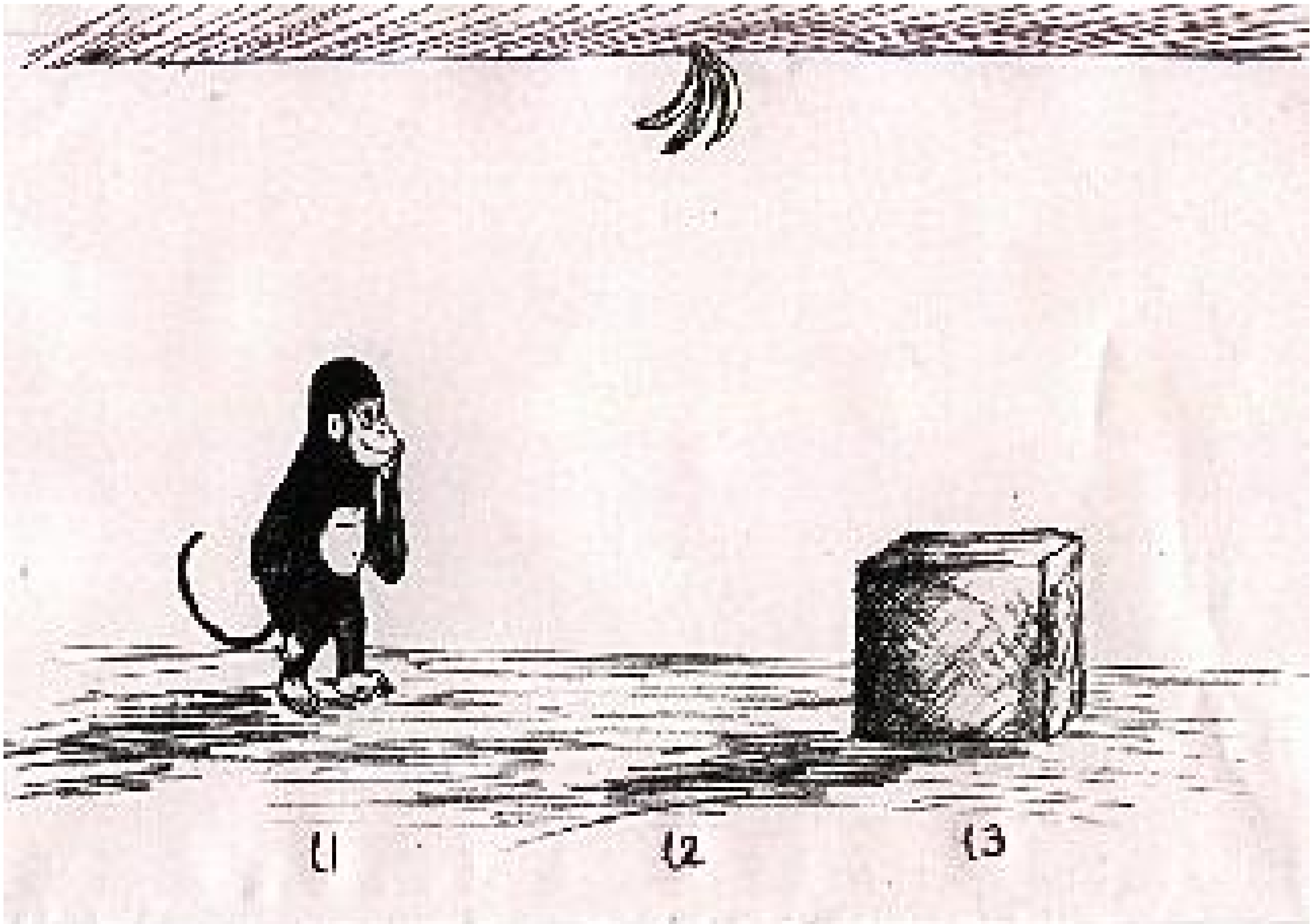
Intelligent Data Analysis
RESEARCH GROUP

http://cw.felk.cvut.cz/doku.php/courses/a4b33zui/start

Agenda

- Logic-based agents
  - declarative nature,
  - ability to represent uncertain information.
- Running example – monkey and banana.
- The most simple but not extendable logical formalization
  - first-order logic, resolution.
- A modular logical formalization
  - requires change tracking,
  - **situation calculus**.
- What is the difference from the real-world planning systems?

Motivation example – monkey and banana



Vladimir Lifschitz: Planning course, The University of Texas at Austin.

Motivation example – monkey and banana

- Problem description
  - a monkey is in a room, a banana hangs from the ceiling,
  - the banana is beyond the monkey's reach,
  - the monkey is able to walk, move and climb objects, grasp banana,
  - the room is just the right height so that the monkey can move a box, climb it and grasp the banana,
  - the goal is to generate this plan (i.e., a sequence of simple actions) automatically.
- Key characteristics
  - a deterministic task,
  - a general description available
    - all the necessary knowledge is provided,
    - we need to **represent** it in some **language**,
    - and perform certain **reasoning/inference**.
  - a planning task.

The first most simple logic-based formalization

- Language $\rightarrow$ first-order logic (FOL).
- The state of the task represented in a single predicate

$$world(monkey\_position, monkey\_onBox, box\_position, has\_banana).$$

- The actions represented by logical formulas/rules
  - the monkey walks

$$\forall P1, P2, V, B, H \;\; world(P1, V, B, H) \rightarrow world(P2, V, B, H).$$

  - the monkey pushes the box

$$\forall P1, P2, B, H \;\; world(P1, down, P1, H) \rightarrow world(P2, down, P2, H).$$

  - the monkey climbs the box

$$\forall P, V, B, H \;\; world(P, down, P, H) \rightarrow world(P, up, P, H).$$

  - the monkey grasps the banana

$$world(at\_ban, up, at\_ban, no) \rightarrow world(at\_ban, up, at\_ban, yes).$$

Reasoning as the next necessary step

- The initial and goal states

$$world(at\_window, down, at\_corner, no).$$

$$\exists P, V, B \;\; world(P, V, B, yes).$$

- Reasoning checks whether the goal formula/state follows from KB
  - the knowledge base (KB) contains the inference rules and the initial state,
  - KB must **entail** the goal formula/state.
- Reasoning should have a few essential characteristics
  - $\alpha \models \beta$ ...sentence α **entails** sentence β,
  - KB $\vdash_i \alpha$ ... α is **derivable** from KB (a sentence as well!) by procedure i,
  - **soundness** – i is sound iff: KB $\vdash_i \alpha \Rightarrow$ KB $\models \alpha$,
  - **completeness** – i is complete iff: KB $\models \alpha \Rightarrow$ KB $\vdash_i \alpha$.

The inference procedure

- The candidate inference procedures
  - resolution, deductive inference, model checking.
- **Resolution** will be used in this lecture
  - proof by contradiction
    - good control over search space, goal-directed search,
    - KB $\models \phi$ iff KB $\cup \{\neg\phi\} \models \Box$,
    - contradiction equals to an empty clause, denoted by $\Box$,
  - sound and refutation complete
    - the given goal confirms or refutes, but does not have to finish,
    - does not generate a list of true statements (confirmation only)!
  - exponential complexity
    - complete conjunctive normal form (CNF) minimizes the number of applicable inference rules,
  - easy to automate,
  - has some "restricted" more efficient variations.

The resolution steps

- Negate the goal formula.
- Translate KB into CNF
  - eliminate implications,
  - move negations into atomic formulae, reduce their scope,
  - skolemization – eliminate existential quantifiers,
  - bind each quantifier to a unique variable,
  - move universal quantifiers to the left of formula – **prenex** form,
  - distribute disjunctions inwards over conjunctions,
  - drop the prefix = all universal quantifiers,
- Generate the resolution proof tree
  - resolution rule:

$$\frac{(l_1 \vee \cdots \vee l_k \qquad m_1 \vee \cdots \vee m_n)_{unify(l_i, \neg m_j)=\theta}}{(l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee l_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n)_\theta}$$

  - contradiction/empty clause terminates the proof.

The resolution outcome

- The list of **clauses** = disjunctions of of atomic formulas and their negations:

  $C_{walk}: \neg world(P1, V, B, H) \vee world(P2, V, B, H).$

  $C_{push}: \neg world(P1, down, P1, H) \vee world(P2, down, P2, H).$

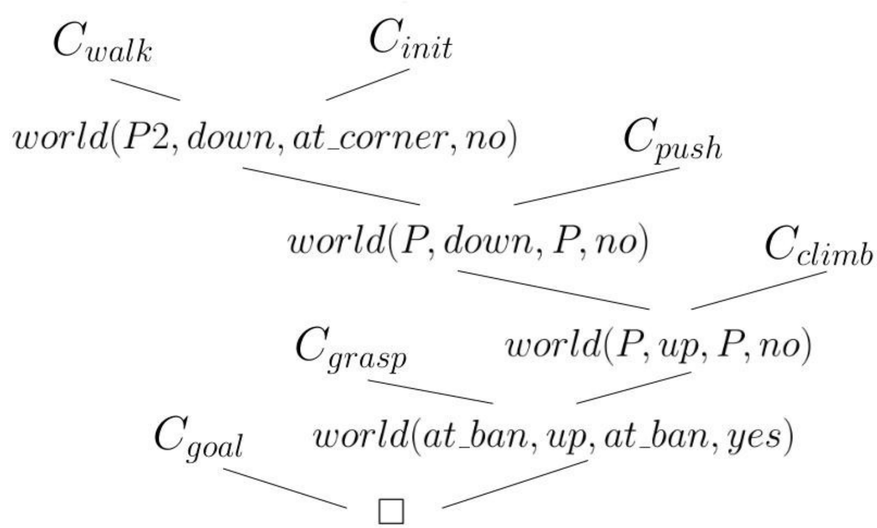
  $C_{climb}: \neg world(P, down, P, H) \vee world(P, up, P, H).$

  $C_{grasp}: \neg world(at\_ban, up, at\_ban, no) \vee world(at\_ban, up, at\_ban, yes).$

  $C_{init}: world(at\_window, down, at\_corner, no).$

  $C_{goal}: \neg world(P, V, B, yes).$

- The resolution tree:



The first solution – feedback

- Positives
  - we proved that the monkey can reach the banana,
  - the specific rules tailored to the actions maximize efficiency,
    * help to implicitly prune the search space.
- Negatives
  - confirmation only, no plan directly available,
  - the knowledge base is not extendable nor easy to maintain
    * it is not modular, knowledge hardwired in rules.
- Common answer to the negatives
  - employ more distinct predicates,
  - maintain them with context-free rules,
  - **situation calculus** is a way to represent change in FOL.

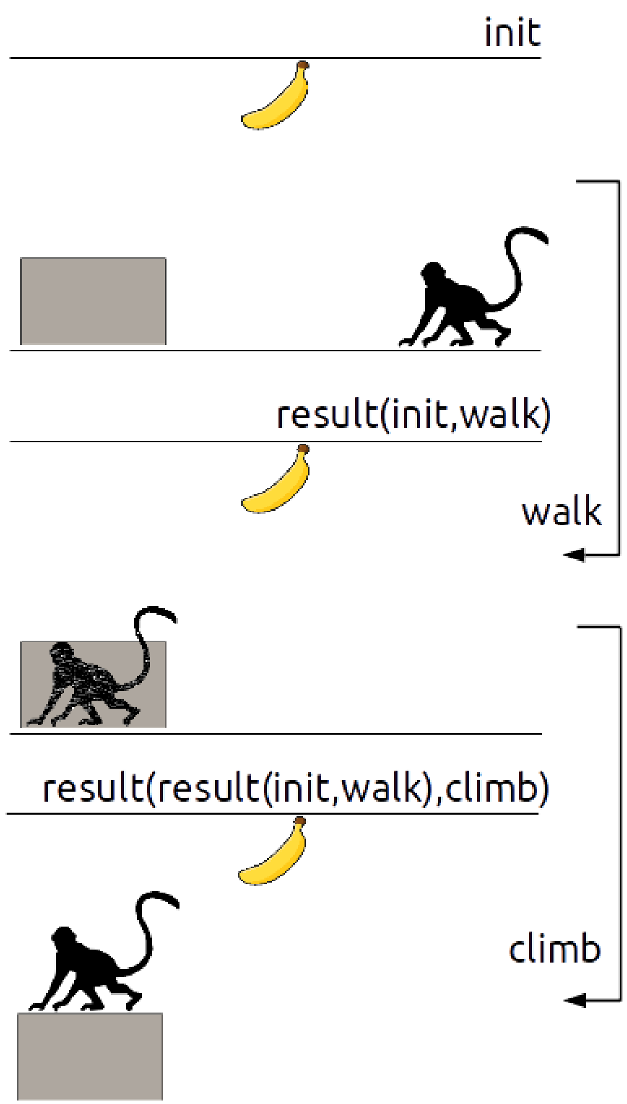
Keeping track of change

- facts hold in particular **situations**, rather than eternally,
- **situation calculus** is one way to represent change in FOL
  - predicates either rigid (eternal) or fluent (changing)
    - with or without possibility to change during time,
  - adds a situation argument to each fluent predicate
    - e.g. $agent(monkey, at\_ban, now)$, term now denotes a situation,
  - rigid predicates e.g. $moves(monkey)$, $moveable(box)$,
  - situations connected by the $result$ function
    - s is a situation, $result(s, a)$ is a situation too,
    - $result(s, a)$ reached by doing action a in situation s.
- two main fluent predicates replace the previous $world$ predicate

$$agent(agent\_name, agent\_position, stands\_on, situation)$$

$$object(object\_name, object\_position, who\_stands, situation)$$

Keeping track of change



agent(monkey, right, ground, init).
object(box, left, none, init).

agent(monkey, left, ground, result(init,walk)).
object(box, left, none, result(init,walk)).

agent(monkey, left, box, result(result(init,walk),climb)).
object(box, left, monkey, result(result(init,walk),climb)).

Description and application of actions

- "effect" axiom – defines changes corresponding to the outcome of action

$$\forall X, Y, V, Z\ (agent(X, V, ground, Z) \wedge walks(X) \rightarrow agent(X, V, ground, result(Z, walk))).$$

- "frame" axiom – defines all that **remains the same**

$$\forall X, Y, W, Z\ (object(X, V, Y, Z) \rightarrow object(X, V, Y, result(Z, walk))).$$

- **frame problem**: how to cope with the unchanged facts smartly

  (a) representational

  avoid the frame axioms in the local world,

  F fluent predicates, A actions $\rightarrow \mathcal{O}(FA)$ frame axioms,

  (b) inferential

  avoid copying of unchanged to keep the state information complete,

Description and application of actions

- **qualification problem**
  - precise description of real actions requires infinite care,
  - what if box is slippery or nailed to the ground or ...?
- **ramification problem** = too many action consequences
  - real actions have many secondary/hidden consequences,
  - agent moves with all that it holds,
  - when the box is dusty, the dust moves with it,
  - gloves needed to climb get worn out.

Description and application of actions

- "successor-state" axioms diminish the representational frame problem,
- each axiom attached to one predicate (instead of an action)

$$\text{P holds after execution of action} \Leftarrow [\text{action caused } P \;\vee\; P \text{ held before and action did not touch } P]$$

- for standing on an object
  (a new predicate $on(agent, object, situation)$ considered)

$$\forall A, S\ [on(X1, X2, result(S, A)) \leftarrow (A = climb \wedge agent(X1, Y, S) \wedge object(X2, Y, S) \wedge climbable(Y)) \vee on(X1, X2, S)].$$

- we obtain F axioms
  - the total number of literals is $\mathcal{O}(AE)$
    (E is the number of effects per action),
- alternative notation for $on()$ with frame and effect axioms?

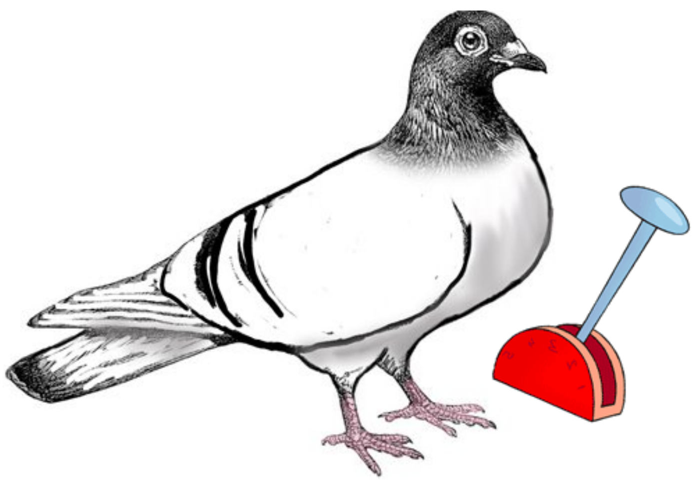
Frame problem and (human) brain

- Sherlock Holmes and the dog that did not bark

```
Is there any point to which you would wish to draw my attention?
To the curious incident of the dog in the night-time.
The dog did nothing in the night-time.
That was the curious incident, remarked Sherlock Holmes.
```

- What can human brain and memory do?
  - even human memory does not store everything,
  - but often it pretends so,
  - people do not realize the simplification tricks,
  - focus on action and its results,
  - frame gets reconstructed,
  - that is why people have difficulties to notice the events, that *did not happen*.

Frame problem and (human) brain



The second solution – a detailed overwiew

- rigid facts, do not change, agent/object characteristics

  $moveable(box).\ climbable(box).$

  $walks(monkey).\ moves(monkey).\ climbs(monkey).$

- define the environment

  $\forall X, Y, Z\ (agent(X, at\_ban, V, Z) \land V \neq ground \rightarrow has(X, banana, Z)).$

- the effect axioms (r=result,g=ground,n=none)

  $\forall X, Y, V, V1, Z\ (agent(X, V, g, Z) \land walks(X) \rightarrow$

  $\quad agent(X, V1, g, r(Z, walk(X, V, V1)))).$

  $\forall X, Y, V, Z\ (agent(X, V, g, Z) \land object(Y, V, n, Z) \land climbs(X) \land climbable(Y) \rightarrow$

  $\quad agent(X, V, Y, r(Z, climb(X, Y)) \land object(Y, V, X, r(Z, climb(X, Y)).$

  $\forall X, Y, V, V1, Z\ (agent(X, V, g, Z)\ \land\ object(Y, V, n, Z)\ \land\ moves(X) \land$

  $\quad moveable(Y) \rightarrow agent(X, V1, g, r(Z, move(X, Y, V, V1))) \land$

  $\quad object(Y, V1, n, r(Z, move(X, Y, V, V1)))).$

The second solution – a detailed overwiew

- the frame axioms

$$\forall X, Y, U, V, W, W1, Z\ (object(X, V, Y, Z) \rightarrow$$
$$object(X, V, Y, r(Z, walk(U, W, W1)))).$$

- the init state

$$agent(monkey, at\_window, g, init).$$

$$object(box, at\_corner, n, init).$$

- the goal state

$$\exists X, Z\ has(X, banana, Z).$$

The second solution – the resolution tree

The second solution – feedback

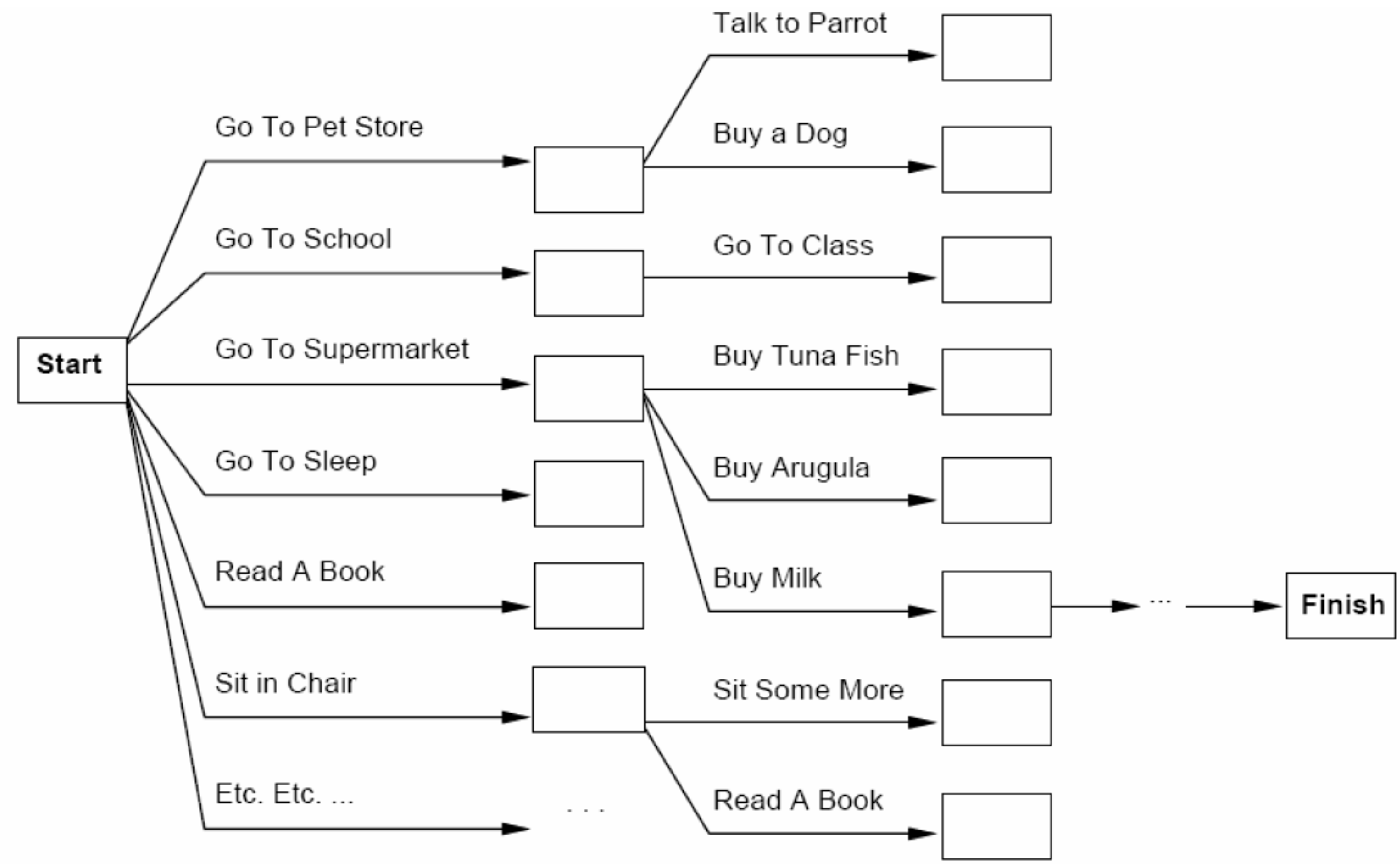
- Positives
  - we proved that the monkey can reach the banana again,
  - the plan can be deduced from the situation argument
    * resolution proof, remains confirmative, but ...
  - the knowledge base is modular and extendable
    * consider a task extension with multiple cooperating agents
      · dog can move objects but cannot climb them, cat just the opposite,
    * and multiple different objects in the room
      · toilet can be climbed but cannot be moved, picture just the opposite,
- Negatives
  - efficiency is worse than in the first solution
    * the same asymptotic bounds, but worse parameters
      (the number of predicates, the solution depth etc.),
    * runs around 10 times slower.

Efficiency of logical planning

- Key properties of knowledge representation languages
  - expressiveness – a scale of objects and relationships we can capture,
  - suitability for automated reasoning (soundness, completeness, efficiency).
- We often sacrifice expressiveness to improve efficiency.
- A simple example in **propositional** logic:
  - **Horn clauses** with **forward** or **backward chaining**
  - Horn clauses may have one positive literal at most
    - $(P \vee \neg Q \vee \neg R) \Leftrightarrow (Q \wedge R) \Rightarrow P$,
    - we cannot formalize e.g. $(\neg P \vee Q \vee R) \Leftrightarrow P \Rightarrow (Q \vee R)$,
    - i.e., only simple facts and implications with the only consequent.
  - Reward is inference in $O(n)$, where n is the number of clauses in KB
    - forward or backward chaining represent the inference methods,
    - both approaches are sound and complete,
    - namely the complexity of backward chaining often sublinear.

Why specialized planners

- Find a suitable sequence of actions (a path) in graph below



Russell, Norvig: Artificial Intelligence: A Modern Approach.

Summary

- We learned planning as FOL deduction
  - situation calculus helps to represent change, temporal problems,
  - asks for no principal changes in FOL,
  - not the major planning approach,
  - however illustrative
    - complex AI tasks can in theory be solved with a very general knowledge representation language and inference procedure,
    - favourable wrt soundness and completeness,
    - the above-mentioned generality leads to inefficiency,
  - real systems balance between efficiency, expressivity and completeness.
- Interactions with the other courses
  - prerequisite: B0B01LGR (FOL, resolution),
  - corequisite: B4M36LUP (first-order theorem proving),
  - corequisite: B4M36PUI (planning in detail, heuristic solutions).

Recommended reading, lecture resources

:: Reading

- Russel, Norvig: **AI: A Modern Approach**, Third edition
  - Part III Knowledge, reasoning and planning,
  - `http://aima.cs.berkeley.edu`.