Predicate logic, situation calculus

Jiří Kléma

Department of Computer Science, Czech Technical University in Prague



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

Logic as a knowledge representation tool

- What convenient characteristics has propositional logic shown?
 - declarative nature
 - * knowledge and reasoning naturally separated,
 - * procedural languages miss a general inference mechanism,
 - ability to represent uncertain information
 - * with the aid of disjunction (and negation),
 - * "Tomorrow afternoon I will go to the cinema or do sports."
 - * cannot be taken for granted (see databases, data structures).
 - compositional and context-independent language
 - * sentence semantics is a function of semantics of its parts,
 - * semantics of $A \wedge B$ related to validity of A and B,
 - * unlike natural language: "She saw it then."

Predicate logic

- Propositional logic with limited expressivity
 - problems: no mechanism for generalization and scaling (see wumpus world)
 - * All the rooms that adjoining wumpus' room stench.
 - $*W_{1,2} \Rightarrow (S_{1,1} \land S_{2,2} \land S_{1,3}), W_{2,1} \Rightarrow (S_{1,1} \land S_{2,2} \land S_{3,1}) \text{ etc.}$
 - also relationships between objects, time, syllogisms,
- predicate logic introduces objects and relations among them
 - inspired by the strength of natural language, but preserves unambiguity.
- first-order predicate logic (first-order logic, FOL)
 - intentional restrictions in expressive power:
 - * relations cannot be seen as objects,
 - * general assertions about relations not allowed,
- higher-order logic systems
 - higher expressivity, more complex theory and typically lower efficiency.

FOL – syntax, language elements

- constants refer to particular objects
 - peter, paul, scotland, ai-foundations (Prolog notation),
- predicates represent object properties and relationships among objects
 - man(peter), father(peter,paul), lives(peter,scotland),
- functions as indirect object references
 - father(peter), lives(peter) (functions can replace predicates and vice versa),
- variables refer to sets of objects,
 - -X, Y, Z (Prolog notation: start with capitals),
- quantifiers determine variable interpretation
 - \forall for any substitution of object for variable the sentence holds,
 - \exists we can find an object for which the sentence holds,
 - $\forall X \exists Y \text{ teaches}(X,Y) \text{each course has at least one teacher,}$
- logical operators and parentheses identical with propositional logic.

FOL – syntax

- Syntax definition of correct/well-formed formulae
 - term is each constant, variable or function applied to them
 - * refers to an object, nothing else cannot be term (i.e., predicates are not terms),
 - well-formed formulas (WFFs) are
 - * atomic formulas $p(t_1, \ldots, t_n)$,
 - * where p is a predicate symbol and t_i terms,
 - * formulas $\neg p$ and $p \Rightarrow q$, provided that p and q are WFFs,
 - * formula $\forall Xp$, provided that X is a variable and p WFF.
- Syntax, notes
 - WFF may contain free variables (not quantified)
 - * this may cause difficulties when evaluating WFFs,
 - * sentence is a WFF if all the variables are bound (also closed formula),
 - the logical operators $\vee, \wedge, \Leftrightarrow$ and quantifier \exists are derived.

FOL – semantics, interpretation

- How to determine the truth value of FOL formulas/sentences?
 - given by context, but ...
- can we check validity for all interpretations?
 - we work with large or infinite domains (e.g., real numbers),
 - as a consequence, an infinite number of models may exist.
- Interpretation defines:
 - objects, predicates and functions corresponding to symbols including their meaning
 - * domain $\Delta = \{ \text{Jiří Kléma}, \text{Michal Pěchouček}, STM Y33ZUI, OI A4B33ZUI \},$
 - * constant interpretation: $l_C:C\to\Delta$, jirka refers to Jiří Kléma, zui1 to course STM Y33ZUI, etc.
 - * predicate interpretation (arity n): $l_P: P \to P(\Delta^n)$, teaches refers to teacher-course relationship, teaches/2 = {{jirka,zui1},{michal,zui2}}.

FOL – semantics, interpretation

- Is formula $\forall X(p(X) \lor q(X)) \Rightarrow (\forall Xp(X) \lor \forall Xq(X))$ tautology?
 - to disprove we only need to find an interpretation for which is the formula false,
 - e.g., interpretation $\Delta = \{a,b\}$, $p_D = \{a\}$, $q_D = \{b\}$,
 - the left hand side of the implication holds:
 - * X=a: $p(a) \lor q(a) = T \lor F = T$, X=b: $p(b) \lor q(b) = F \lor T = T$,
 - the right hand side does not hold:
 - * X=b: p(b) = F, $\forall X p(X) = F$ similarly for X=a and $\forall X q(X) = F$.

Inference in FOL – unification

- How to substitute for free variables or terms when unifying terms?
 - KB1: Jirka knows Filip. Vaclav is known by everybody. Everybody knows his/hers mother.

Monika knows someone.

- question: Who knows Jirka?
- return all the correct substitutions for the free variable considering KB,
- $\ \mathsf{KB2:} \ knows(jirka, filip), \ \forall Yknows(Y, vaclav), \ \forall Zknows(Z, mother(Z)), \\ \exists Wknows(monika, W),$
- KB3: knows(jirka, filip), knows(Y, vaclav), knows(Z, mother(Z)), knows(monika, s1),
- query: ?knows(jirka,X)
- substitution: $\theta_1 = \{X|filip\}$, $\theta_2 = \{jirka|Y,X|vaclav\}$, $\theta_3 = \{jirka|Z,X|mother(jirka)\}$,
- we search for the most general substitution to keep the terms unifiable.

Inference in FOL – unification

- Substitution and unification needed for any inference
 - see generalized modus ponens

$$\frac{A, A \Rightarrow B}{B} \rightarrow \frac{p'_1, \dots, p'_n, (p_1 \land p_2 \land \dots \land p_n \Rightarrow q)}{Subst(\theta, q)}$$
$$Subst(\theta, p'_i) = Subst(\theta, p_i)$$

— When two persons know each other, they greet each other.

$$* \forall X, Y(knows(X,Y) \Rightarrow greets(X,Y)).,$$

- MP application on KB (KB $\vdash \alpha$)
 - *greets(jirka, filip), greets(Y, vaclav),
 - * greets(U, mother(U)), greets(monika, s1).

Resolution in FOL – conversion into CNF (1)

- More difficult than in propositional logic variables and quantifiers,
- E.g.: All Romans who know Marcus either hate Caesar or think that anyone who hates anyone is crazy.

$$\forall X \big[(roman(X) \land knows(X, markus)) \Rightarrow \\ (hates(X, caesar) \lor \forall Y \exists Z (hates(Y, Z) \Rightarrow thinkscrazy(X, Y))) \big]$$

- general steps of the conversion:
 - 1. eliminate implications

$$\forall X \big[\neg (roman(X) \land knows(X, markus)) \lor (hates(X, caesar) \lor \forall Y \neg (\exists Zhates(Y, Z)) \lor thinkscrazy(X, Y))) \big]$$

2. move negations into atomic formulae, reduce their scope

$$\forall X \big[\neg roman(X) \lor \neg knows(X, markus) \lor hates(X, caesar) \lor \\ \lor \forall Y \forall Z \big(\neg hates(Y, Z) \lor thinkscrazy(X, Y) \big) \big]$$

Resolution in FOL – conversion into CNF (2)

- 4. skolemization eliminate existential quantifiers
 - no occurrence, let us try: $\forall X \exists Y father(Y,X) \rightarrow \forall X father(s1(X),X)$
 - -s1 is in the given context a Skolem function (assigns a father to each instantiation of X),
- 5. bind each quantifier to a unique variable

$$\forall Xroman(X) \lor \forall Xgreek(X) \rightarrow \forall Xroman(X) \lor \forall Ygreek(Y)$$

6. move universal quantifiers to the left of formula - prenex form,

$$\forall X, Y, Z \big[\neg roman(X) \lor \neg knows(X, markus) \lor \\ hates(X, caesar) \lor \neg hates(Y, Z) \lor thinkscrazy(X, Y) \big]$$

- 7. distribute disjunctions inwards over conjunctions
 - apply distributive laws $a \lor (b \land c) \rightarrow (a \lor b) \land (a \lor c)$
- 8. drop the prefix = all universal quantifiers

$$\neg roman(X) \lor \neg knows(X, markus) \lor \\ hates(X, caesar) \lor \neg hates(Y, Z) \lor thinkscrazy(X, Y)$$

Motivation example 3 – Does Zuzana lay eggs?

:: If I tell you that:

- (S1) Platypus and echidna are the only mammals that lay eggs.
- (S2) Only birds and mammals are warm-blooded.
- (S3) Zuzana, my armadillo, is warm-blooded and has no feathers.
- (S4) Every bird has feathers.

:: and ask: (D) Does Zuzana lay eggs?

Motivation example 3 – Does Zuzana lay eggs?

:: If I tell you that:

- (S1) Platypus and echidna are the only mammals that lay eggs.
- (S2) Only birds and mammals are warm-blooded.
- (S3) Zuzana, my armadillo, is warm-blooded and has no feathers.
- (S4) Every bird has feathers.
- :: and ask: (D) Does Zuzana lay eggs?
- :: Inference in natural language:
 - Zuzana has no feathers and thus it is not a bird.
 - Zuzana is warm-blooded and it is not a bird, it must be a mammal.
 - Zuzana is mammal and armadillo, not platypus/echidna, it cannot lay eggs.
- :: How to implement automatically? Strings difficult for inference ...

Does Zuzana lay eggs? – resolution proof (1)

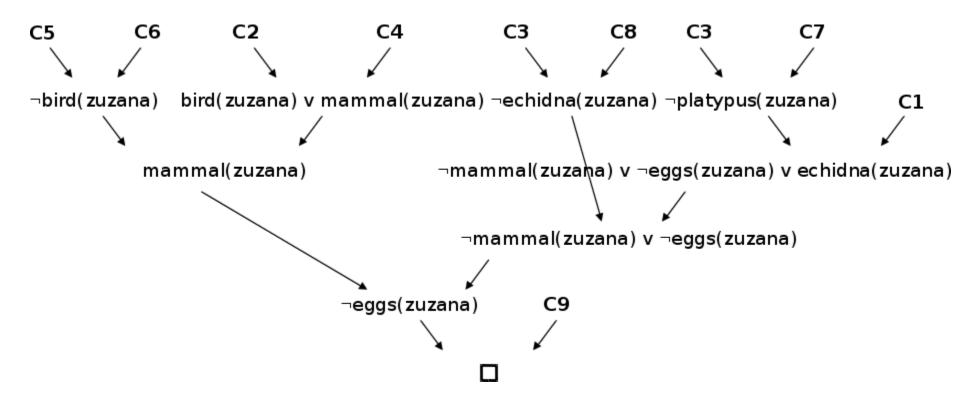
- (S1) $\forall X(mammal(X) \land eggs(X) \Rightarrow echidna(X) \lor platypus(X))$
- (S2) $\forall X(warm_blooded(X) \Rightarrow bird(X) \lor mammal(X))$
- $\bullet \textbf{ (S3)} \ armadillo(zuzana) \land warm_blooded(zuzana) \land \neg feathers(zuzana) \\$
- (S4) $\forall X(bird(X) \Rightarrow feathers(X))$
- (F) $\forall X(armadillo(X) \Rightarrow \neg(platypus(X) \lor echidna(X)))$
- \bullet (D) eggs(zuzana)

Does Zuzana lay eggs? – resolution proof (1)

- (S1) $\forall X(mammal(X) \land eggs(X) \Rightarrow echidna(X) \lor platypus(X))$
- (S2) $\forall X(warm_blooded(X) \Rightarrow bird(X) \lor mammal(X))$
- $\bullet \textbf{ (S3)} \ armadillo(zuzana) \land warm_blooded(zuzana) \land \neg feathers(zuzana) \\$
- (S4) $\forall X(bird(X) \Rightarrow feathers(X))$
- (F) $\forall X(armadillo(X) \Rightarrow \neg(platypus(X) \lor echidna(X)))$
- lacksquare (D) eggs(zuzana)
- (C1) $\neg mammal(X) \lor \neg eggs(X) \lor echidna(X) \lor platypus(X)$
- $\bullet \text{ (C2) } \neg warm_blooded(X) \lor bird(X) \lor mammal(X)$
- (C3,4,5) $armadillo(zuzana) \land warm_blooded(zuzana) \land \neg feathers(zuzana)$
- (C6) $\neg bird(X) \lor feathers(X)$
- $\bullet \textbf{ (C7,8)} \ (\neg armadillo(X) \lor \neg platypus(X)) \land (\neg armadillo(X) \lor \neg echidna(X)) \\$
- ullet (C9) eggs(zuzana)

Does Zuzana lay eggs? – resolution proof (2)

:: Resolution proof tree:



:: The assertion that Zuzana lays eggs contradicts the theory.

:: Zuzana does not lay eggs.

FOL in wumpus world

- FOL is expressive enough to describe wumpus world and for inference in it,
- environment description
 - adjacency of cave rooms

$$\forall X, Y, A, B \left[adjacent([X, Y], [A, B]) \Leftrightarrow [A, B] \in \{[X + 1, Y], [X - 1, Y], [X, Y + 1], [X, Y - 1]\} \right]$$

- diagnostic rules
 - * infer cause from effect for breeze:

$$\forall S \ (breeze(S) \Rightarrow \exists R \ (adjacent(R,S) \land pit(R))), \\ \forall S \ (\neg breeze(S) \Rightarrow \neg \exists R \ (adjacent(R,S) \land pit(R))), \\$$

- causal rules
 - * infer effect from cause for cave rooms:

$$\forall R (pit(R) \Rightarrow \forall S (adjacent(R, S) \Rightarrow breeze(S))), \ \forall S [\forall R (adjacent(R, S) \Rightarrow \neg pit(R)) \Rightarrow \neg breeze(S)],$$

— the above written diagnostic and causal rules equivalent to formula:

$$\forall S (breeze(S) \Leftrightarrow \exists R (adjacent(R, S) \land pit(R))).$$

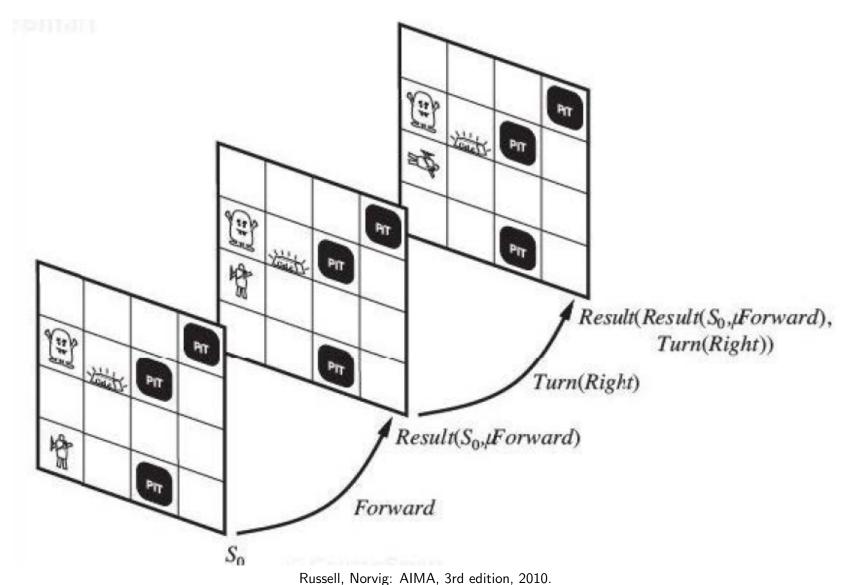
FOL in wumpus world

- it turns out more difficult to represent perception, reflexes and environment modifications
 - perception (T denotes time, resp. the number of step) $\forall T, S, B \ (sensors([S, B, glitter], T) \Rightarrow glitter(T))$
 - $\text{ reflex} \\ \forall T \left(glitter(T) \right) \Rightarrow best_action(grab, T) \right)$

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. holds(gold, now), term now denotes a situation,
 - rigid predicates e.g. pit(R), adjacent(R, S),
 - situations connected by the result function,
 - * s is a situation, result(a, s) is a situation too,
 - * result(a, s) reached by doing action a in situation s.

Keeping track of change



Russell, Norvig: AliviA, 3rd edition, 2010

Description and application of actions

- "effect" axiom defines changes corresponding to the outcome of action
 - $\forall S \ withGold(S) \Rightarrow holds(gold, result(grab, S))$
- "frame" axiom defines all that remains the same
 - $\forall S \ hasArrow(S) \Rightarrow hasArrow(result(grab, S))$
- frame problem: how to cope with the unchanged facts smartly
 - (a) representational avoid the frame axioms in the local world, $F \mbox{ fluent predicates, } A \mbox{ actions} \rightarrow \mathcal{O}(FA) \mbox{ frame axioms,}$
 - (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 gold is slippery or nailed to the ground or . . . ?

ramification problem

- real actions have many secondary/hidden consequences,
- agent moves with all that he holds,
- when the gold is dusty, the dust moves with it,
- gloves needed to grab 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)
 - P holds after execution of action \Leftarrow [action caused P

 $\vee P$ held before and action did not touch P

for gold handling

$$\forall A, S \ holds(gold, result(A, S)) \Leftarrow \\ [(A = grab \land withGold(S)) \lor (holds(gold, S) \land A \neq release)]$$

- we obtain F axioms
 - the total number of literals is $\mathcal{O}(AE)$ (E is the number of effects per action),
- alternative notation for holds 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.

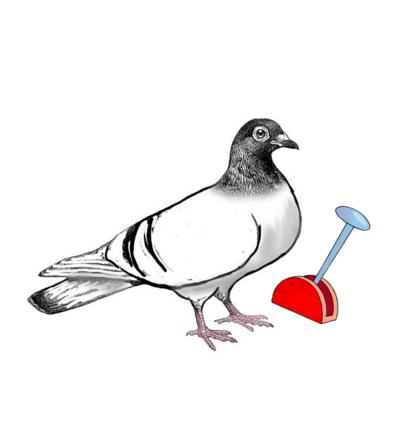
The dog did nothing in the night-time.

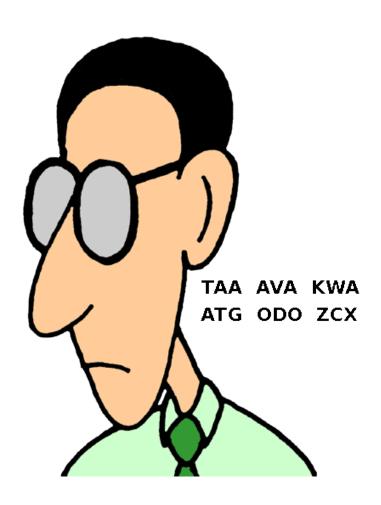
That was the curious incident, remarked Sherlock Holmes.

- 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

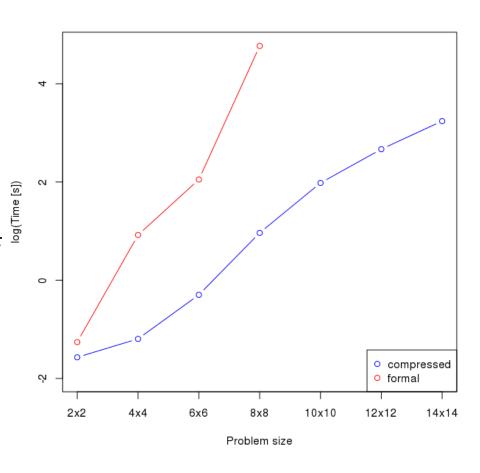




Wumpus: formal and efficient solution, comparison

formal solution

- modular knowledge,
- extendable,
- easy to maintain,
- efficient solution
 - minimizes the number of predicates,
 - the ultimate case is 1,
 - implicit search space pruning,
- demo.



Making plans

initial facts in the knowledge base

```
-position(agent, [1, 1], s_0), position(gold, [1, 2], s_0),
```

query:

- $-Ask(KB,\exists S\ holds(gold,S)),$
- i.e., in which situation will the agent hold gold?

answer:

- $-\{S/result(grab, result(go_forward, s_0))\}$,
- i.e., go forward and then grab the gold,

assumptions:

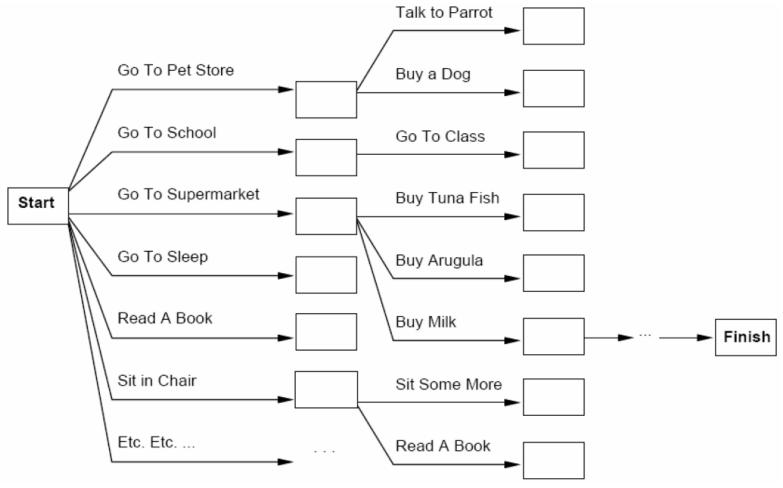
- agent is interested only in plans starting in s_0 ,
- $-s_0$ is the only situation described in KB.

Making plans: a better way

- lacktriangle make plans in the form of sequence of actions $[a_1,a_2,\ldots,a_n]$,
- lacktriangleq planResult(P,S) is a result of execution of plan P in state S
 - projection task situation after application of sequence of actions? $position(agent, [1,1], s_0) \wedge position(gold, [1,2], s_0) \wedge \neg holds(O, s_0) \\ position(gold, [1,1], result([go([1,1], [1,2]), grab(gold), go([1,2], [1,1])], s_0))$
 - planning task which sequence of actions leads to the desired situations?
- query $Ask(KB, \exists P \ holds(gold, planResult(P, s_0)))$
 - has the solution $\{P/[go_forward, grab]\}$,
- definition of planResult in terms of result
 - $\forall S \ planResult([], S) = S$,
 - $\forall A, P, S \ planResult([A|P], S) = planResult(P, result(A, S)),$
- motivation for planning as a discipline
 - specialized planners more efficient than general reasoning/search.

Why specialized planners

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



Russell, Norvig: Artificial Intelligence: A Modern Approach.

Situation calculus: example

- Let us have a logical KB defining distances between Czech cities:
 - $-\ dist(praha, brno, 209)$, dist(brno, zlin, 96),
- agent state determined by its position and the amount of gasoline in the tank
 - -position(City,State), tank(Gas,State),
 - let fuel consumption be 6l/100km,
- define the predicate journeyLength(P,D)
 - e.g. journeyLength([praha,brno,zlin],305),
- define an effect axiom which specifies consequences of action takeJourney(P)
 - consider the change of position and the amount of gasoline in the tank,
- do we need any frame axioms?

Situation calculus: example

recursive definition of journeyLength(P,D):

$$\forall X \ journeyLength([X], 0)$$

$$\forall X, Y, L, D_1, D_2 \ dist(X, Y, D_1) \land journeyLength([Y|L], D_2) \Rightarrow$$

$$journeyLength([X, Y|L], D_1 + D_2)$$

lacktriangle the effects of action takeJourney(P) arrive only when its assumptions are met

$$\forall X, Y, B, D, P, S \ position(X, S) \land journeyLength(P, D) \land \\ tank(B, S) \land (B > D * 6/100) \land first(P, X) \land last(P, Y) \Rightarrow \\ position(Y, result(takeJourney(P), S)) \land \\ tank(B - D * 6/100, result(takeJourney(P), S))$$

- lacktriangle action takeJourney(P) changes both the fluent predicates
 - frame axioms not needed, needed e.g. for a hypothetical predicate: roadworthyCar(Roadworthy, State).

Summary

- In a dynamic world it is more difficult to derive and maintain knowledge
 - general bottlenecks frame and ramification problems,
 - even human memory reconstructs the frame only roughly,
 - representation of fluent worlds in FOL \rightarrow situation calculus,
- another KR systems
 - production systems,
- practical utilization
 - e.g. semantic web,
- where you can learn more?
 - A4M33RZN Advanced methods of knowledge representation,
 - A4M33AU Automated reasoning.

Recommended reading, lecture resources

:: Reading

- Russel, Norvig: Al: A Modern Approach, Logical Agents, chapter 7
 - representation for intelligent agents,
 - available in pdf http://aima.cs.berkeley.edu/newchap07.pdf.
- Mařík a kol. Umělá inteligence 1
 - kapitola Reprezentace znalostí
 - * základní formáty, logika, sémantické sítě, rámce,
 - kapitola Řešení úloh a dokazování vět
 - * predikátová logika a důkazní prostředky,
- Mařík a kol. Umělá inteligence 2
 - kapitola Znalostní inženýrství
 - * praktická, znalostní systémy v konkrétních aplikacích.