Search Tree in Prolog

Samy Aittahar

ULiege - INFO0049

Tuesday 20th February, 2018

Me in a nutshell

- PhD Student in computer science.
- Main research : Transfer learning for deep reinforcement learning.
- Personal page : `http://www.montefiore.ulg.ac.be/~saittahar/`.
- Contact me at `saittahar@ulg.ac.be` or come to my office (I-136).

Key urls

- Sessions repository : `https://github.com/epochstamp/INFO0049-1`
- Submission platform : `https://submit.montefiore.ulg.ac.be/teacher/viewprojects/INFO0049-1`

Outline

- Today : Presentation of search tree in Prolog + exercises (no expected deliverable).
- Later : Presentation of (up to 6) projects with expected deliverable.
  - Only source code with clear documentation is needed.
  - The timeframe for each project is one month, according to deadline displayed in the submission platform.
  - After the deadline, you are expected to present your work to the class. The teacher may be present.

Facts and search trees

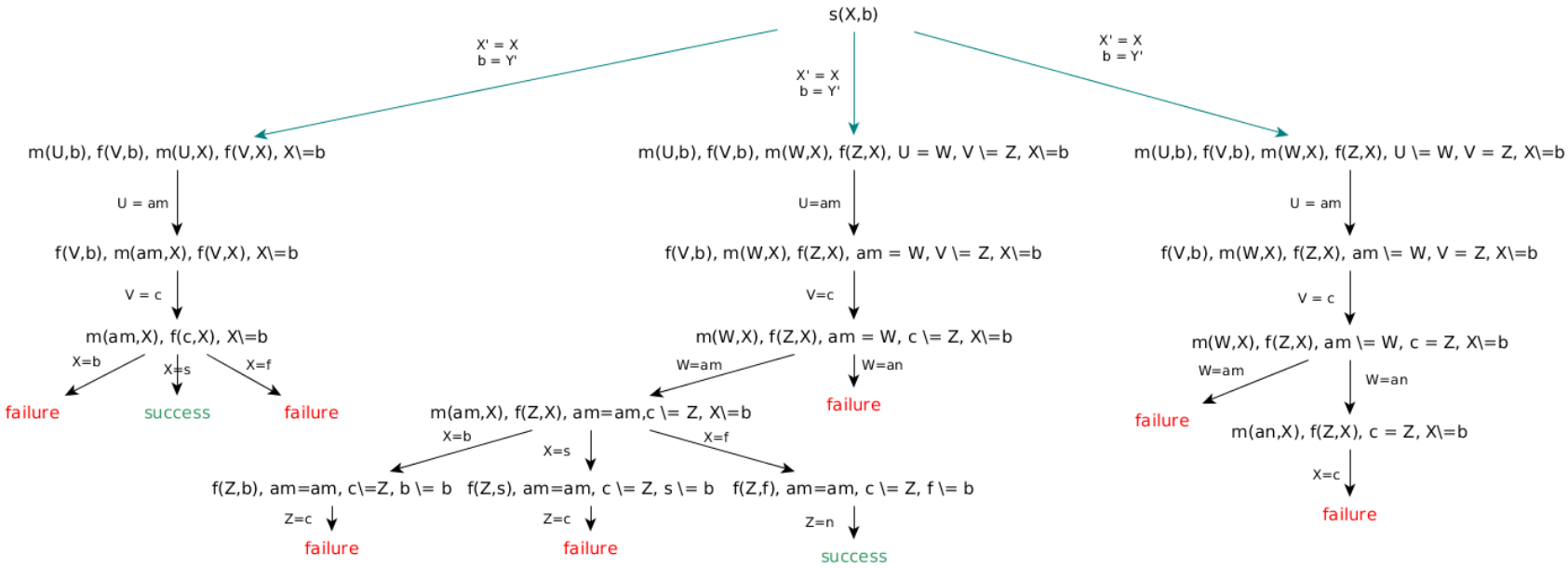
- A fact is a logical statement, represented by a first-order logical predicate ;
- Depth-first searching algorithm through a given database of facts and predicates;
  - Goes backward when either unification fails or succeed ;
- Pro : enumerates all possible solutions ;
- /!\ Even if logically equivalent, order of facts often matters for computational efficiency !

Example 1 - Facts

```
m(am,b).
m(am,s).
m(am,f).
m(an, c).
f(c,b).
f(c,s).
f(n,f).
f(e, c).

s(X,Y) :- m(U,Y), f(V,Y), m(U,X), f(V,X), X \= Y.
s(X,Y) :- m(U,Y), f(V,Y), m(W,X), f(Z,X), U = W,
          V \= Z, X \= Y.
s(X,Y) :- m(U,Y), f(V,Y), m(W,X), f(Z,X), U \= W,
          V = Z, X \= Y.
```

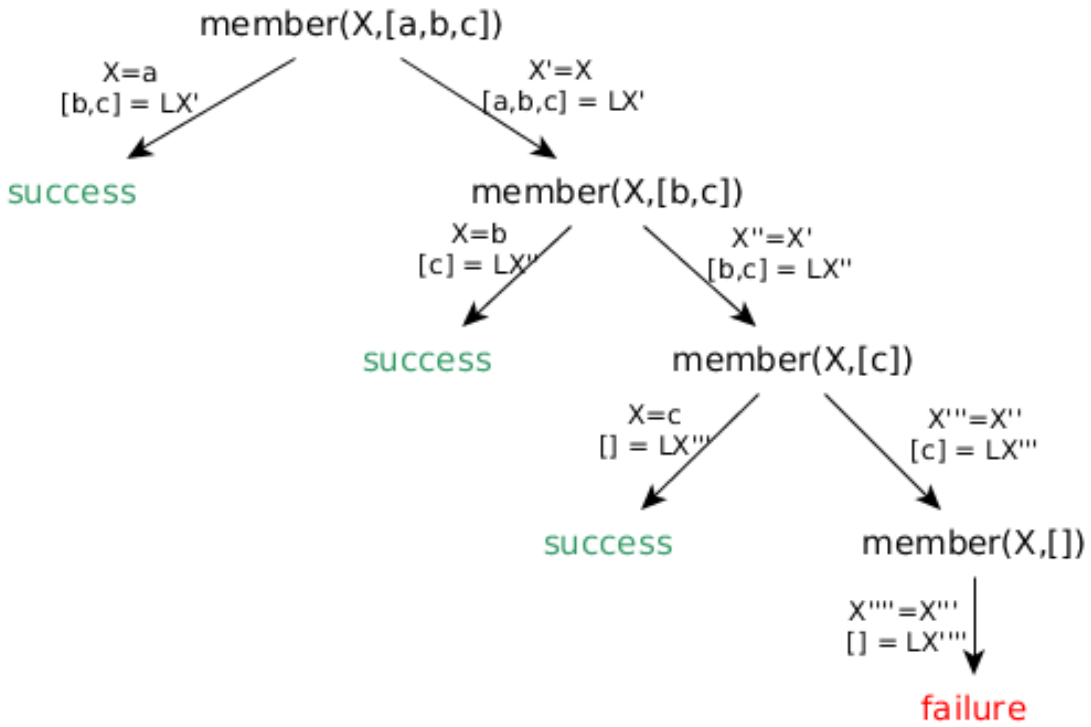
Search Tree for Example 1



Example 2 - Lists

```
member(X,[X|_]).
member(X,[_|LX]) :- member(X,LX).
```

Search Tree for Example 2

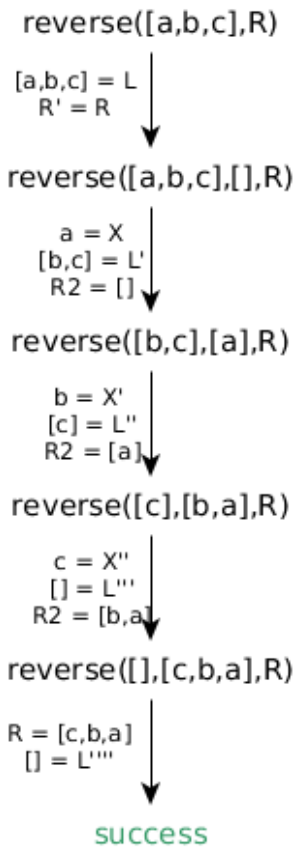


Example 3 - Accumulator list

```
reverse(L,R) :- reverse(L,[],R).

reverse([],R,R).
reverse([X|L],R2,R) :- reverse(L,[X|R2],R).
```

Search Tree for Example 3



Search Tree for Example 3 (bis)

```
reverse(L,[a,b,c]).
```

???

Training time !

Reference : *Leon Sterling and Ehud Shapiro. 1986. The Art of Prolog: Advanced Programming Techniques. MIT Press, Cambridge, MA, USA.*