

Backward Chaining

Authored by
mohammad looti

June 5, 2026

RECOMMENDED CITATION

mohammad looti (2026). *Backward Chaining*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=38346>

Backward chaining (or backward reasoning) is an inference method that can be described (in lay terms) as working backward from the goal(s). It is used in automated theorem provers, proof assistants and other artificial intelligence applications, but it has also been observed in primates.

In game theory, its application to (simpler) subgames in order to find a solution to the game is called backward induction. In chess, it's called retrograde analysis, and it is used to generate tablebases for chess endgames for computer chess.

Backward chaining is implemented in logic programming by SLD resolution. Both rules are based on the modus ponens inference rule. It is one of the two most commonly used methods of reasoning with inference rules and logical implications - the other is forward chaining. Backward chaining systems usually employ a depth-first search strategy, e.g. Prolog.

How it works

Backward chaining starts with a list of goals (or a hypothesis) and works backwards from the consequent to the antecedent to see if there is data available that will support any of these consequents. An inference engine using backward chaining would search the inference rules until it finds one which has a consequent (Then clause) that matches a desired goal. If the antecedent (If clause) of that rule is not known to be true, then it is added to the list of goals (in order for one's goal to be confirmed one must also provide data that confirms this new rule).

For example, suppose that the goal is to conclude the color of my pet Fritz, given that he croaks and eats flies, and that the rule base contains the following four rules:

If X croaks and eats flies - Then X is a frog
If X chirps and sings - Then X is a canary
If X is a frog - Then X is green
If X is a canary - Then X is yellow

This rule base would be searched and the third and fourth rules would be selected, because their consequents (Then Fritz is green, Then Fritz is yellow) match the goal (to determine Fritz's color). It is not yet known that Fritz is a frog, so both the antecedents (If Fritz is a frog, If Fritz is a canary) are added to the goal list. The rule base is again searched and this time the first two rules are selected, because their consequents (Then X is a frog, Then X is a canary) match the new goals that were just added to the list. The antecedent (If Fritz croaks and eats flies) is known to be true and therefore it can be concluded that Fritz is a frog, and not a canary. The goal of determining Fritz's color is now achieved (Fritz is green if he is a frog, and yellow if he is a canary, but he is a frog since he croaks and eats flies; therefore, Fritz is green).

Note that the goals always match the affirmed versions of the consequents of implications (and not

the negated versions as in modus tollens) and even then, their antecedents are then considered as the new goals (and not the conclusions as in affirming the consequent) which ultimately must match known facts (usually defined as consequents whose antecedents are always true); thus, the inference rule which is used is modus ponens.

Because the list of goals determines which rules are selected and used, this method is called goal-driven, in contrast to data-driven forward-chaining inference. The backward chaining approach is often employed by expert systems.

Programming languages such as Prolog, Knowledge Machine and ECLiPSe support backward chaining within their inference engines.

Use by primates

Kanzi, a bonobo (pygmy chimpanzee, *Pan paniscus*) dramatically illustrated his use of this strategy when he was being taught how to create stone tools by a human expert. Unable to replicate the manipulations of the human expert, Kanzi eventually resorted to smashing stones upon others, and simply selected shards with sharp edges, in order to produce his stone tools.