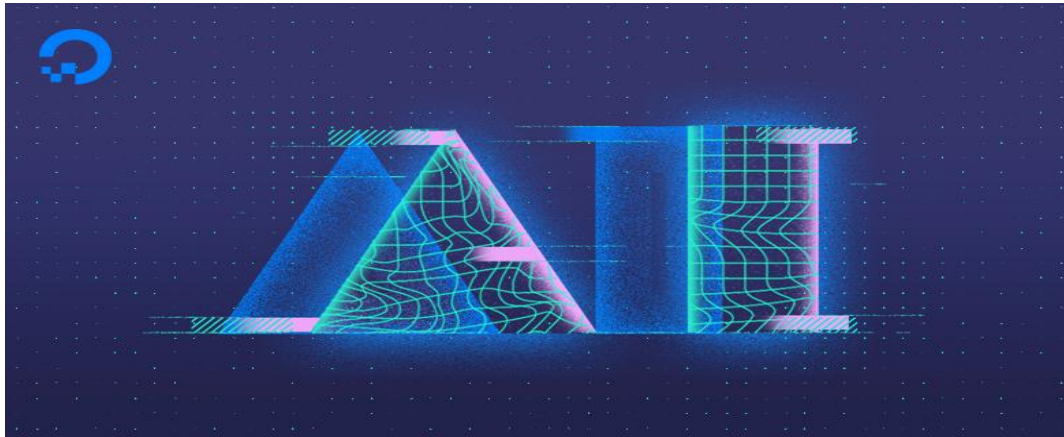


Unit 7: Production System



Presented By : Tekendra Nath Yogi

Tekendranath@gmail.com

College Of Applied Business And Technology

Contd...

- **Outline:**
 - 7.1 Strong Methods vs Weak Methods
 - 7.2 Advantages of Production Systems
 - 7.3 Production Systems and inference methods
 - 7.3.1 Conflict resolution strategies
 - 7.3.2 Forward chaining
 - 7.3.3 Backward chaining

Production Systems

- The production System is a model of computation that has proved particularly important in AI, Both for implementing search algorithms and for modeling human problem solving.
- A production system provides pattern-directed control of a problem solving process and consists of:
 - a set of production rules,
 - a working memory, and
 - a recognize- act control cycle.

Contd...

- A schematic drawing of a production system is as shown in figure below:

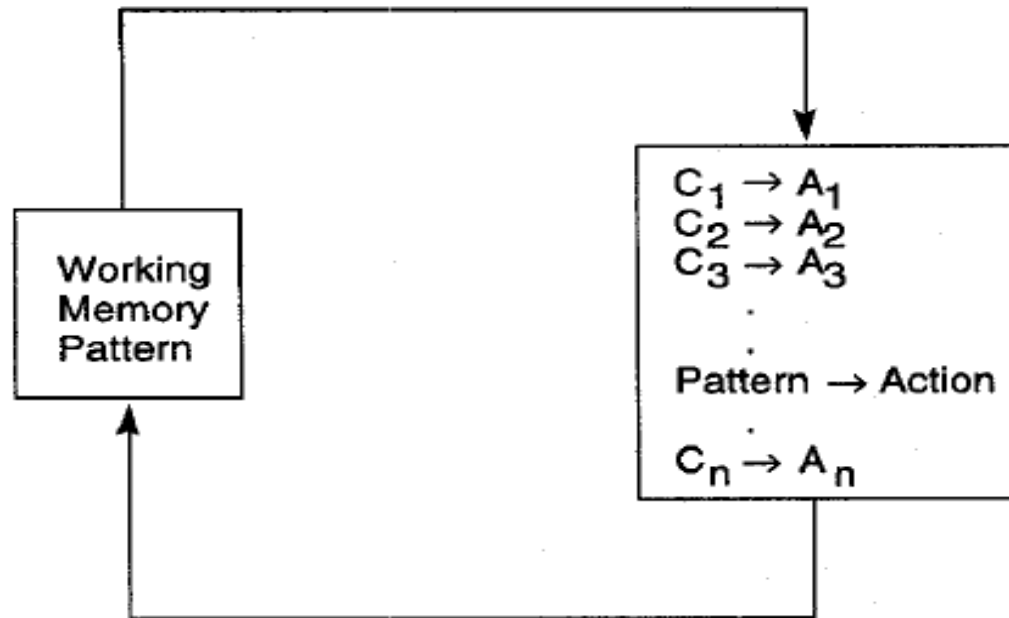


Fig: A production system

- Control loops until working memory pattern no longer matches the condition of any productions.

Contd...

- The set of production rules:
 - These are often simply called productions. A production is a condition-action pair and defines a single chunk of problem-solving knowledge.
 - The **condition** part of the rule is a pattern that determines when that rule may be applied to a problem instance.
 - The **action** part defines the associated problem-solving step.
 - E.g., If it is raining then open the umbrella.

Contd...

- Working memory:
 - Working memory contains a description of the current state of the world(facts/ data) in a reasoning process.
 - This description is a pattern that is matched against the condition part of a production to select appropriate problem solving actions.
 - When the condition part of a rule is matched by the contents of working memory, the action associated with that condition may then be performed.
 - The actions of production rules are specifically designed to alter the contents of working memory.

Contd...

- **The recognize- act cycle:** The control structure for a production system is simple:
 - Working memory is initialized with the beginning problem description as a set of patterns in working memory.
 - These patterns are matched against the conditions of the production rules, this produces a subset of the production rules, called the conflict set, whose conditions match the patterns in working memory.
 - One of the productions in the conflict set is then selected(conflict resolution) and the production is fired.
 - To fire a rule, its action is performed, changing the contents of working memory.
 - After the selected production rule is fired, the control cycle repeats with the modified working memory.
 - This process terminates when the contents of working memory do not match any rule conditions.

Contd...

- **Example of production system execution:**
 - **Problem:** sorting a string composed of the letters a, b, and c.
 - **Working memory:** cbaca
 - **Production set:**
 1. **ba** → **ab**
 2. **ca** → **ac**
 3. **cb** → **bc**
 - In this example, a production is enabled if its condition matches a portion of the string in working memory.
 - When a rule is fired, the substring that matched the rule condition is replaced by the string on the right hand side of the rule.

Contd...

Iteration #	Working memory	Conflict set	Rule fired
0	cbaca	1, 2, 3	1
1	cabca	2	2
2	acbca	3, 2	2
3	acbac	1, 3	1
4	acabc	2	2
5	aacbc	3	3
6	aabcc	\emptyset	Halt

Fig : Trace of simple production system

Advantages of Production Systems for AI

- The production system offers a general framework for implementing search.
- Because of its simplicity, modifiability, and flexibility in applying problem solving knowledge, the production system has proved to be an important tool for the construction of expert systems and other AI applications.
- The major advantages of production systems for artificial intelligence include:
 - Separation of knowledge and control
 - A natural mapping onto state space search.
 - Modularity of production rules.
 - Pattern directed control.
 - Opportunities for heuristic control of search
 - Tracing and explanation.
 - Language independence.
 - A plausible model of human problem solving

Contd...

- **Separation of knowledge and Control:**
 - The production system is an elegant model of separation of knowledge and control in a computer program.
 - Control is provided by the recognize-act cycle of the production system loop, and the problem –solving knowledge is encoded in the rules themselves.
 - The advantages of this separation include ease of modifying the rule base without requiring a change in the code for program control and , conversely, the ability to alter the code for program control without changing the set of production rules.

Contd...

- **A Natural mapping onto state space search:**
 - The components of a production system map naturally into the constructs of state space search. The successive states of working memory form the nodes of a state space graph.
 - The production rules are the set of possible transitions between states, with conflict resolution implementing the selection of a branch in the state space. These rules simplify the implementation, debugging, and documentation of search algorithms

Contd...

- **Modularity of production rules:**
 - An important aspect of the production system model is the **lack of any syntactic interactions between production rules**. Rules may only effect the firing of other rules by changing the pattern in working memory.
 - This syntactic independence supports the incremental development of expert systems by successively adding, deleting, or changing the knowledge (rules) of the system.

Contd...

- **Pattern directed control:**
 - The problems addressed by AI programs require particular flexibility in program execution.
 - This goal is served by the fact that the rules in a production system may fire in any sequence.
 - The descriptions of a problem that make up the current state of the world determine the conflict set and, consequently, the particular search path and solution.

Contd...

- **Tracing and Explanation:**
 - The modularity of rules and the iterative nature of their execution make it easier to trace execution of a production system.
 - Because each rule corresponds to a single chunk of problem-solving knowledge, the rule content should provide a meaningful explanation of the system's current state and action

Contd...

- **Language Independence:**
 - The production system control model is independent of the representation chosen for rules and working memory, as long as that representation supports pattern matching.

Contd...

- **A Plausible Model of human problem solving:**
 - Modeling human problem solving was among the first uses of production systems.
 - E.g.,
 - IF the ‘traffic light’ is green
 - THEN the action is go

 - IF the ‘traffic light’ is red
 - THEN the action is stop

Disadvantages of Production Systems

- There are three main shortcomings:
 - Opaque relations between rules.
 - Ineffective search strategy.
 - Inability to learn.

Contd...

- **Opaque relations between rules:**
 - Although the individual production rules tend to be relatively simple and self-documented, their logical interactions within the large set of rules may be opaque.
 - productions systems make it difficult to observe how individual rules serve the overall strategy.

Contd...

- **Ineffective search strategy:**
 - The rule interpreter applies an exhaustive search through all the production rules during each cycle.
 - Production systems with a large set of rules (over 100 rules) can be slow, and thus large rule-based systems can be unsuitable for real-time applications.

Contd...

- **Inability to learn:**
 - In general, rule-based systems do not have an ability to learn from the experience.
 - Unlike a human, who knows when to ‘break the rules’, a production system cannot automatically modify its knowledge base, or adjust existing rules or add new ones.
 - The knowledge engineer is still responsible for revising and maintaining the system.

Conflict Resolution

- The set of rules that have their conditions satisfied by working memory elements forms a conflict set.
- A conflict is said to occur if more than one rules are found to fire in one cycle.
- In such a scenario, the control structure must determine which rule to fire from this conflict set of active rules.
- This process of selection is called as conflict resolution so, conflict resolution normally selects a single rule to fire.
- There are three conflict resolution mechanism:
 - Refraction
 - Recency and
 - specificity

Contd...

- Refraction:
 - Refraction specifies that once a rule has fired, it may not fire again until the working memory elements that match its conditions have been modified.
 - This discourage looping.

Contd...

- Recency:
 - The recency strategy prefers rules whose conditions match with the patterns most recently added to working memory.
 - This focuses the search on a single line of reasoning.

Contd...

- **Specificity:**
 - This strategy assumes that a more specific problem-solving rule is preferable to a general rule.
 - A rule is more specific than another if it has more conditions,
 - This implies that it will match fewer potential working memory patterns.

Production Systems and inference systems

- In a **production system**, the domain knowledge is represented by a set of IF-THEN production rules and data is represented by a set of facts about the current situation.
- The **rule interpreter** compares each rule stored in the **rule base** with facts contained in the **working memory**.
- When the IF (condition) part of the rule matches a fact, the rule is fired and its THEN (action) part is executed.
- The fired rule may change the set of facts in the working memory by adding a new fact.
- The matching of the rule's IF parts to the facts produces inference chains. The inference chain indicates how a production system applies the rules to reach a conclusion.

Contd...

- There are two principal ways in which rules are executed.
 - Forward chaining: **Forward chaining** starts from the start state (Known facts or data) and proceeds towards the goal state (conclusion) and
 - Backward chaining: On the other hand **backward chaining** starts from the goal state and proceeds towards start state.

Contd...

- **Forward chaining:**
 - Sometimes called the data driven approach.
 - Working from the facts(data) to conclusion(goal).
 - To chain forward, match data in working memory against 'conditions' of rules in the rule-base. When one of them fires, this is liable to produce more data. So the cycle continues.
 - Forward chaining is the best choice if:
 - All the facts are provided with the problem statement; or:
 - There are many possible goals, and a smaller number of patterns of data; or There isn't any sensible way to guess what the goal is at the beginning of the consultation.

Contd...

- Forward Chaining example:
 - Initial content of working memory(Known fact) is Start
 - Production Rule set:

1. $p \wedge q \rightarrow \text{goal}$
2. $r \wedge s \rightarrow p$
3. $w \wedge r \rightarrow p$
4. $t \wedge u \rightarrow q$
5. $v \rightarrow s$
6. $\text{start} \rightarrow v \wedge r \wedge q$

- **Problem** : Prove If p and q true Then goal is true
i.e., prove $p \wedge q \rightarrow \text{goal}$

Contd...

- The following table shows the sequence of rules firing and the stages of working memory in the execution . Execution halts when a goal is reached.

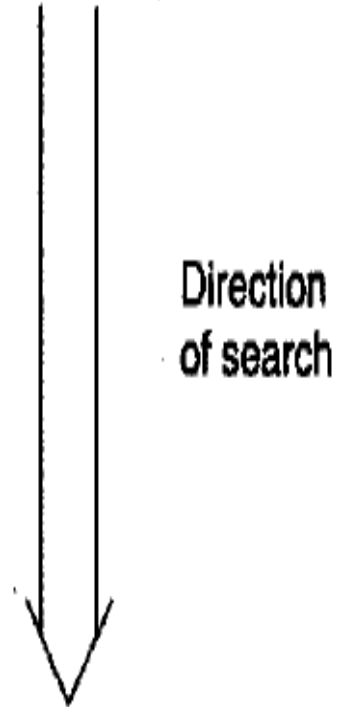
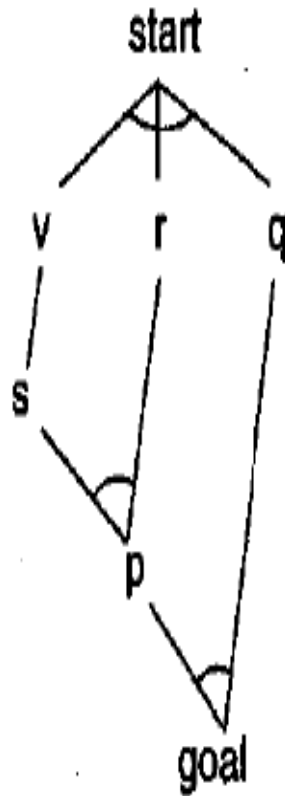
A Trace of execution is shown in the following table:

Iteration #	Working memory	Conflict set	Rule fired
0	start	6	6
1	start, v, r, q	6, 5	5
2	start, v, r, q, s	6, 5, 2	2
3	start, v, r, q, s, p	6, 5, 2, 1	1
4	start, v, r, q, s, p, goal	6, 5, 2, 1	halt

- Conflict resolution strategy used here is:** choose rule that has fired least recently(or not at all.

Contd...

Space searched by execution:



Contd...

- **Backward chaining:**

- It is also known as goal driven search. A goal driven search begins with a goal and works backward to establish its truth.
- To implement this in a production system, the goal is placed in working memory and matched against the ACTIONS of the production rules. These ACTIONS are matched just as the CONDITIONS of the productions were matched in the data- driven reasoning.
- When the ACTION of a rule is matched, the CONDITIONS are added to working memory and become the new sub-goals of the search.
- The new sub-goals are then matched to the ACTIONS of other production rules. This process continues until a fact given in the problem's initial description is found.
- The search stops when the CONDITONs of all the productions fired in this backward fashion are found to be true.
- These CONDITIONS and the chain of rule firings leading to the original goal form a proof of its truth through successive inferences.

Contd...

- **Backward chaining Example:**

- Initial content of **working memory** is **goal**

- initially given fact is **start**

- **Production Rule set:**

1. $p \wedge q \rightarrow \text{goal}$
2. $r \wedge s \rightarrow p$
3. $w \wedge r \rightarrow p$
4. $t \wedge u \rightarrow q$
5. $v \rightarrow s$
6. $\text{start} \rightarrow v \wedge r \wedge q$

- **Problem** : Prove If p and q true Then goal is true

i.e., prove

$$p \wedge q \rightarrow \text{goal}$$

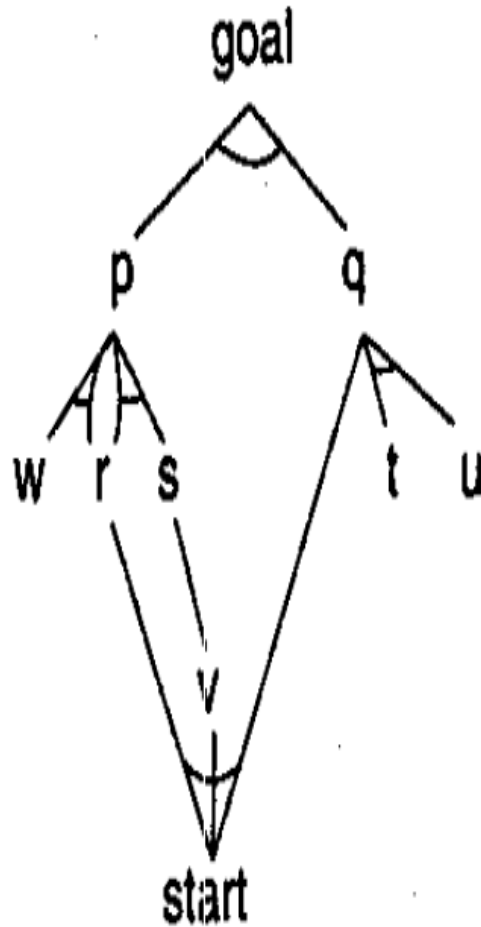
Contd...

Trace of execution:

Iteration #	Working memory	Conflict set	Rule fired
0	goal	1	1
1	goal, p, q	1, 2, 3, 4	2
2	goal, p, q, r, s	1, 2, 3, 4, 5	3
3	goal, p, q, r, s, w	1, 2, 3, 4, 5	4
4	goal, p, q, r, s, w, t, u	1, 2, 3, 4, 5	5
5	goal, p, q, r, s, w, t, u, v	1, 2, 3, 4, 5, 6	6
6	goal, p, q, r, s, w, t, u, v, start	1, 2, 3, 4, 5, 6	halt

Contd...

searched by execution:



Contd...

- Backward chaining is the best choice if:
 - The goal is given in the problem statement, or can sensibly be guessed at the beginning of the consultation; or
 - The system has been built so that it sometimes asks for pieces of data rather than expecting all the facts to be presented to it.
 - e.g. "please now do the gram test on the patient's blood, and tell me the result“.
 - This is because (especially in the medical domain) the test may be expensive, or unpleasant, or dangerous for the human participant so one would want to avoid doing such a test unless there was a good reason for it.
- If it is important that the system should seem to behave like a human expert, backward chaining is the best choice.

Homework

- What is AI? Differentiate between strong AI and Weak AI.
- What is weak methods and strong methods? Differentiate between weak methods and weak methods.
- What is productions and production systems? Explain different advantages and disadvantages of production systems.
- What is production system? Explain its different architectural components.
- Why conflict resolution is required in production systems? Explain different conflict resolution strategy used in the production systems.
- How inferencing is done in production system? Differentiate between forward chaining and backward chaining.

Thank You !

