

Sistemas de Produção

Trata-se de um modelo de computação que – ao guardar importante semelhança com a maneira humana de processar dados – tem servido de importante paradigma em implementações de IA. Um sistema de produção fornece um modelo com controle adequado para o processo de solução de problemas.

Um sistema de produção comporta 3 componentes

1. Um conjunto de regras de produção que representam o conhecimento do sistema
2. Uma memória de trabalho que registra o estado inicial do sistema, do mundo que o envolve e todos os estados intermediários e eventualmente do final
3. Um ciclo de controle do tipo *reconhece - atua*

Regras de produção Muitas vezes estas regras são chamadas apenas de *produções*. Cada uma das regras é um par de componentes. O primeiro é chamado *condição* e o segundo *ação*. A dupla forma então uma regra cujo formato é *se... então...*. Qualquer semelhança com o onipresente comando *if* de qualquer linguagem de programação NÃO é coincidência.

A condição determina quando esta regra pode se aplicar à solução do problema que se está estudando (ou tentando resolver). A ação, determina um passo na busca dessa solução.

Memória de trabalho Contém uma descrição do estado atual do mundo. Esta descrição é um padrão que é comparado com as condições da base de regras para selecionar quais são as ações adequadas para alterar este mundo (e resolver o problema). Quando a condição de uma regra coincide com o conteúdo da memória de trabalho, a ação equivalente a esta condição PODE ser aplicada sobre o mundo. As regras de produção são desenhadas de maneira a alterar explicitamente o conteúdo da memória de trabalho.

Ciclo reconhece-atua É a estrutura de controle de um sistema de produção. A memória de trabalho é inicializada com a descrição inicial do problema. O estado atual do mesmo é mantido através de determinados padrões que sejam significativos para a solução do problema. Estes padrões são comparados com as condições das regras. As regras para as quais houver “casamento” (entre condição e situação da memória de trabalho) formam um conjunto (chamado de conjunto de conflito). Todas estas ações poderiam ser disparadas, já que todas estão **habilitadas**.

Um subcomponente do sistema de produção, chamado resolutor de conflitos analisa, o conjunto de conflito e escolhe uma produção, que é então disparada. A ação é aplicada sobre a memória de trabalho e esta é modificada.

Após isto, o ciclo se repete e assim segue até que nenhuma condição seja satisfeita, quando então o sistema pára.

O resolutor de conflitos deve ter uma estratégia para selecionar a condição elegida. Essa estratégia pode ser simples (escolher a primeira, escolher uma qualquer...) ou pode ser tão complexa quanto se queira, envolvendo heurísticas complexas para ação da regra. Este é o modo pelo qual um sistema de produção adiciona heurística a um algoritmo de busca.

Um modelo puro de SP não possui mecanismo para se recuperar de becos sem saída que surjam na busca, ele simplesmente continua na busca até que mais nenhuma produção esteja habilitada e então pára. Muitas implementações práticas permitem em tais situações a regressão até um estado anterior da memória de trabalho.

Seja um exemplo (extraído de [Lug02] pág 177) de um sistema de produção para ordenar uma sequência alfabética composta pelas letras **a**, **b** e **c**.

Conjunto de regras

1. $ba \rightarrow ab$
2. $ca \rightarrow ac$
3. $cb \rightarrow bc$

A memória de trabalho começa com **cbaca**. neste exemplo, uma produção é habilitada se a sua condição casar com uma porção da sequência que se encontra na memória de trabalho. Quando uma regra é disparada, a subsequência que casou com a condição da regra é substituída pela sequência que se encontra do lado direito da regra.

Sistemas de produção formam um modelo geral de computação que pode ser programado para fazer tudo o que pode ser feito com um computador. Seu verdadeiro poder, contudo, consiste em ser uma arquitetura para sistemas baseados em conhecimento.

Nº da iteração	memória de trabalho	conjunto de conflito	regra disparada
0	cbaca	1,2,3	1
1	cabca	2	2
2	acbca	2,3	2
3	acbac	1,3	1
4	acabc	2	2
5	aacbc	3	3
6	aabcc	0	Parar

A idéia para o projeto baseado em “produção” para a computação tem sua origem nos artigos do lógico Emil Post (1943). A base desta teoria é

um conjunto de regras para reescrita de sequências de caracteres de várias formas similares ao do exemplo acima. Este modelo é equivalente em poder ao de uma máquina de Turing.

Uma aplicação interessante de regras de produção para modelar a cognição humana é encontrada no trabalho de Newell e Simon no que hoje é a Carnegie Mellon University nos anos 60 e 70. Os programas que eles desenvolveram incluindo o *General Problem Solver* são em grande parte responsáveis pela importância dos sistemas de produção na IA. Nessas pesquisas as pessoas eram monitoradas em várias atividades de solução de problemas (lógica de predicados, xadrez, etc). O protocolo (padrões de comportamento) dessas pessoas era registrado e decomposto em seus itens elementares. Esses componentes eram os *bits* básicos do conhecimento para resolver problemas e eram dispostos em um grafo (chamado grafo de comportamento do problema). Um sistema de produção era usado para conduzir a busca dentro desse grafo.

Note-se que os pesquisadores usaram um sistema de produção não apenas para conduzir a busca, mas sobretudo como um modelo real do comportamento humano durante a solução de problemas. As produções correspondiam às habilidades das pessoas e estavam registradas na memória de longo prazo dessas mesmas pessoas. Essas produções não são afetadas pela execução do sistema (e o aprendizado ?) e são apenas invocadas. A memória de trabalho corresponde à memória de curto prazo nos seres humanos e descreve o estágio atual da solução de uma ocorrência do problema. O conteúdo da memória de trabalho, em geral, não é preservado após um problema ter sido resolvido. Esta pesquisa é descrita no livro *Human Problem Solving*, de 1972 dos dois autores.

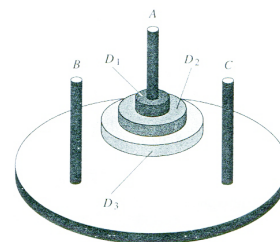
Sistemas especialistas modernos têm perdido um pouco essa aderência original ao comportamento humano, mas o paradigma de sistemas de produção parece adequado para registrar e manter o conhecimento, graças às seguintes características:

- estrutura modular das regras
- separação entre conhecimento e controle
- separação entre memória de trabalho e conhecimento sobre a solução

Para você fazer

De novo, a Torre de Hanoi

O problema da Torre de Hanoi é bastante conhecido. Ele tem uma formulação alternativa, bastante interessante: Os 3 pinos (A, B e C) estão em um círculo. 3 discos (D_1, D_2, D_3 , onde $D_3 > D_2 > D_1$) estão colocados sobre o pino A e devem ser passados para o pino C. A ocupa a posição das 12h, o B a das 20h e o C a das 16h. Define-se que D_1 e D_3 devem sempre se mover em sentido horário e D_2 no sentido anti-horário. Sempre deve-se mover o disco maior que possa ser movido.



Veja-se o esquema sugerido:

Assuma que existem 3 variáveis binárias, de nome V_1, V_2 e V_3 cujos valores são apenas 1 ou 0. A variável V_i será 1 se e somente se o disco D_i é o maior que pode ser movido.

Escreva um sistema de produção que implemente as regras que farão funcionar este algoritmo simples.

Simule a execução do seu sistema com um caso real, mostrando a disposição dos discos após cada etapa.

Este exercício foi adaptado de um similar existente em "Artificial Intelligence" de Nils Nilsson [Nil98]

Resposta

1