

Modelul regulilor de productie

Modelul regulilor de productie este deosebit de important in inteligenta artificiala deoarece acest model a jucat un rol semnificativ in evolutia sistemelor bazate pe cunostinte, de la stadiul de produse ale laboratoarelor de cercetare la acela al produselor comerciale, cu aplicabilitate directa.

1. Reprezentarea cunostintelor sub forma regulilor de productie

Intr-un model de reprezentare bazat pe reguli de productie, cunostintele despre problema sint reprezentate prin doua tipuri de entitati: reguli si fapte. *Regulile* sint cunostintele reprezentate de implicatii si exprima cunostinte generale despre domeniul problemei de rezolvat. *Faptele* sint asertiuni unitare si reprezinta cunostintele specifice care descriu un caz particular, i.e. o instanta a problemei de rezolvat.

Regulile de productie sint formate din doua componente: *partea stinga a regulii*, prescurtat din limba engleza LHS, numita si antecedent, premisa, conditie sau situatie, si *partea dreapta a regulii*, prescurtat din limba engleza RHS, numita si consecinta, concluzie, actiune sau raspuns. Legatura logica intre partea stinga si partea dreapta a regulii este implicatia, in sensul ca adevarul partii stingi determina adevarul partii drepte a regulii, deci $LHS \rightarrow RHS$. Pornind de la structura de control decizionala "if-then" comuna limbajelor de programare, cele mai multe limbaje bazate pe reguli folosesc cuvintele cheie **if then** (uneori **when do**), in romaneste **daca atunci**, pentru a marca partea stinga, respectiv partea dreapta a regulii.

Exemple:

R: **daca** Coco zboara
 si Coco are pene
 atunci Coco este pasare.

Semnificatia unei reguli de productie este urmatoarea: daca premisa poate fi satisfacuta intr-un context dat, atunci consecinta poate fi satisfacuta in acel context.

Fiecare regula reprezinta o unitate de cunostinte despre un anumit domeniu de expertiza. O multime de reguli poate corespunde unui lant de inferente care duc de la faptele cunoscute initial la concluzii necunoscute, concluzii care reprezinta rezolvarea unei probleme.

Faptele, cealalta forma de reprezentare a cunostintelor in modelul regulilor de productie, au o forma care permite identificarea ipotezelor de satisfacut din reguli. Spre deosebire de reguli, care sint imperative si dinamice, faptele sint statice si inactive. Urmatoarele enunturi sint exemple de fapte posibile intr-o baza de cunostinte:

Coco zboara.

Coco are pene.

2. Sisteme bazate pe reguli de productie

2.1. Structura unui sistem bazat pe reguli

Un sistem care utilizeaza modelul regulilor de productie pentru reprezentarea cunostintelor se numeste *sistem bazat pe reguli*, pe scurt SBR [Hayes-Roth,1985]. In general, un sistem bazat pe reguli de productie este format din urmatoarele trei componente principale:

- Baza de cunostinte (BC)
- Memoria de lucru (ML)
- Interpretorul de reguli sau Motorul de inferenta (MI)

Structura generala a unui sistem bazat pe reguli este prezentata in Figura 1.

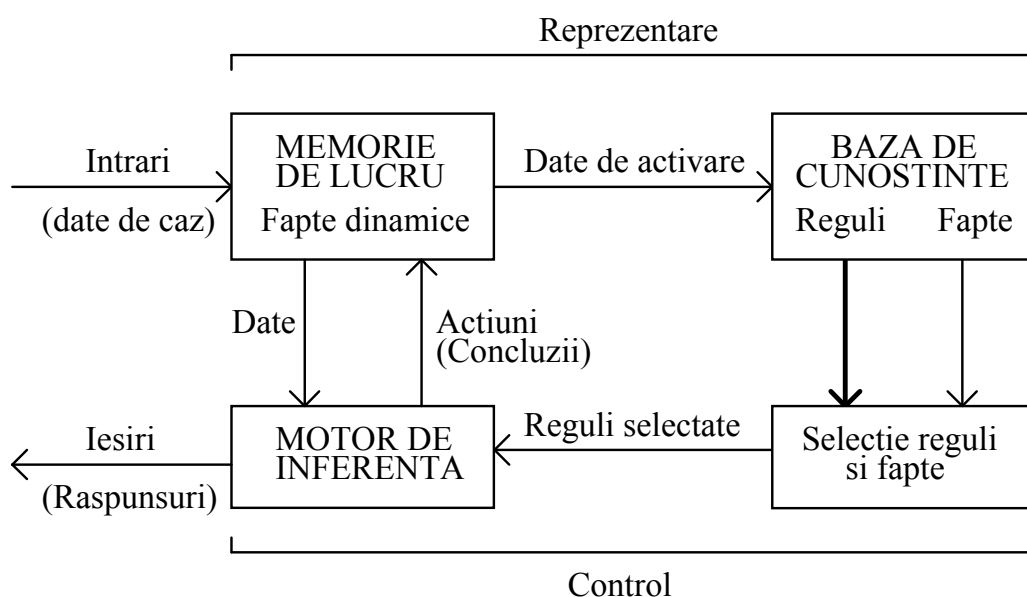


Figura 1. Organizarea unui sistem bazat pe reguli

Baza de cunostinte contine cunostintele domeniului problemei de rezolvat, exprimate sub forma de reguli, si cunostinte specifice instantelor problemei, exprimate sub forma de fapte.

Memoria de lucru contine informatia de stare a rezolvarii problemei, informatie reprezentata prin asertiuni temporare. Aceste asertiuni temporare sint atit datele initiale ale problemei de rezolvat, cit si toate faptele inferate pe parcursul rezolvarii problemei. De obicei, datele din memoria de lucru respecta conventiile sintactice ale faptelor din baza de cunostinte, ceea ce face ca aceste asertiuni temporare sa se numeasca si *fapte dinamice*, iar memoria de lucru sa se numeasca *memorie dinamica*.

Motorul de inferenta reprezinta componenta de control si executie a unui sistem bazat pe reguli. Motorul de inferenta inspecteaza partea stinga a fiecarei reguli din baza de cunostinte pina cind gaseste o regula care identifica continutul memoriei de lucru, dupa care executa (aplica) aceasta regula. Executia regulii de productie determina schimbarea continutului memoriei de lucru pe baza partii drepte a regulii care a identificat. Acest proces continua pina cind in memoria de lucru se acumuleaza faptele care reprezinta solutia problemei sau pina cind nu se mai poate aplica nici o regula. Rezolvarea unei probleme folosind un sistem bazat pe reguli consta deci in realizarea unei serii de inferente printr-un proces de

inlantuire recursiva a regulilor, pina la gasirea solutiei sau pina la intilnirea unei situatii de blocare (nu intotdeauna se ajunge intr-una din aceste doua stari, sistemul putind cicla la infinit in anumite cazuri).

In general, motorul de inferenta accepta intrebari de la utilizator si raspunde folosind informatia dinamica din memoria de lucru (datele de caz) si cunostintele statice din baza de cunostinte (reguli). Cunostintele din baza de cunostinte sint folosite pentru a deriva concluzii despre cazul sau situatia curenta prezentata de utilizator.

2.2. Ciclu de inferenta al unui sistem bazat pe reguli

Functionarea unui sistem bazat pe reguli este formata din executia repetata a unui ciclu de operatii care realizeaza, in esenta, aplicarea unei reguli, deci un pas de inferenta al reprezentarii. Acest ciclu de operatii se numeste *ciclu de inferenta* al sistemului expert bazat pe reguli de productie si se compune din urmatoarele trei etape: identificare, selectie si executie.

(1) *Identificare*. In timpul etapei de identificare se compara continutul memoriei de lucru cu baza de cunostinte si regulile care identifica sint grupate de motorul de inferenta in multimea de conflicte. *Multimea de conflicte* reprezinta multimea regulilor aplicabile intr-un anumit context, context specificat de memoria de lucru.

(2) *Selectia*. Etapa de selectie consta in selectarea unei reguli din multimea de conflicte, pe baza unui criteriu de selectie. Aceasta etapa se mai numeste si *rezolvarea conflictelor*.

(3) *Executie*. In timpul etapei de executie se aplica regula prin executia partii drepte a regulii. Daca partea dreapta a regulii este o concluzie, se adauga faptele din concluzie in memoria de lucru, iar daca partea dreapta este o actiune, se executa aceasta actiune. Actiunea poate indica diverse operatii, cum ar fi: adauga fapte in memoria de lucru, sterge fapte, modifica fapte existente, tipareste mesaje, pune intrebari, opreste procesul de inferenta.

Ciclu de inferenta al unui sistem bazat pe reguli poate fi exprimat, la nivel general, prin urmatorul algoritm.

Algoritm: Functionarea unui sistem bazat pe reguli

1. $ML \leftarrow$ date de caz
 2. **repetă**
 - 2.1. Executa identificare intre ML si BC (partea stinga sau partea dreapta a regulilor si fapte) si creeaza multimea de conflicte a regulilor aplicabile
 - 2.2. Selecteaza o regula dupa un criteriu de selectie
 - 2.3. Aplica regula prin executia partii drepte a regulii
- pina** nu mai sint reguli aplicabile **sau**
memoria de lucru satisface conditia de stare scop **sau**
o cantitate predefinita de efort a fost epuizata

sfirsit.

Observatie: Conditia de stare scop a memoriei de lucru este atinsa in momentul in care memoria de lucru contine solutia problemei.

Etapa de identificare are ca scop determinarea acelor reguli care pot fi satisfacute pe baza continutului curent al memoriei de lucru, prin identificarea partii stingi a regulilor cu faptele din memoria de lucru.

Etapa de selectie este cea care stabileste de fapt forma de cautare utilizata de sistemul bazat pe reguli, deci strategia de control. Intr-un sistem bazat pe reguli strategia de control are doua componente: *stabilirea criteriului de selectie a regulilor* din multimea de conflicte si *directia de aplicare a regulilor*: inlantuirea inainte sau inlantuirea inapoi a regulilor.

Rezolvarea unei probleme de catre un sistem bazat pe reguli este de fapt un proces de cautare a solutiei in care operatorii sint reprezentati de regulile sistemului. In consecinta, toate tehnicile de cautare pot fi aplicate in cazul sistemelor bazate pe reguli. Din punct de vedere al directiei de aplicare a regulilor, inlantuirea inainte a regulilor corespunde unei reprezentari a solutiei problemei prin spatiul starilor, iar inlantuirea inapoi a regulilor corespunde unei reprezentari prin grafuri SI/SAU a solutiei problemei. Rezolvarea conflictelor se refera la ordinea de selectie si preferarea unui operator fata de alti operatori aplicabili intr-un context dat.

2.3. Criterii de selectie a regulilor

Rezultatul etapei de identificare este multimea de conflicte, deci multimea tuturor regulilor care au identificat cu descrierea starii curente a rezolvarii problemei, descriere continuta in memoria de lucru. Rezolvarea conflictelor din etapa de selectie are rolul alegerii uneia sau mai multor reguli care vor fi aplicate. Exista diverse criterii de selectie, prezentate in continuare.

(a) *Selectia primei reguli aplicabile*: considerind ordinea fizica a regulilor din baza de cunostinte, se alege prima regula aplicabila, deci prima care a identificat, si se aplica acea regula.

(b) *Alegerea unei reguli din multimea de conflicte*: in acest caz se foloseste o tehnica explicita de rezolvare a conflictelor pentru a decide asupra regulii de aplicat.

2.4. Directia de aplicare a regulilor

A doua componenta a strategiei de control a unui sistem bazat pe reguli este directia de aplicare a regulilor. Regulile pot fi aplicate utilizind *inlantuirea inainte* prin identificarea partii stingi a regulii cu memoria de lucru, sau utilizind *inlantuirea inapoi* prin identificarea partii drepte a regulii cu memoria de lucru si incercarea satisfacerii partii stingi a regulii. Cele doua abordari sint prezentate sintetic in Figura 2.

INLANTUIRE INAINTE	INLANTUIRE INAPOI
daca A atunci B daca B atunci C A (data) <hr/> C (concluzie)	determina C daca B atunci C daca A atunci B <hr/> (daca A atunci C, implicit) Este A adevarata? (data)

Figura 2. Inlantuirea executiei regulilor de productie

Exemplu. Fie urmatorul set de reguli de productie:

- R1: **daca A si B atunci C**
- R2: **daca C atunci D**
- R3: **daca E atunci B**
- R4: **daca A si E atunci C**

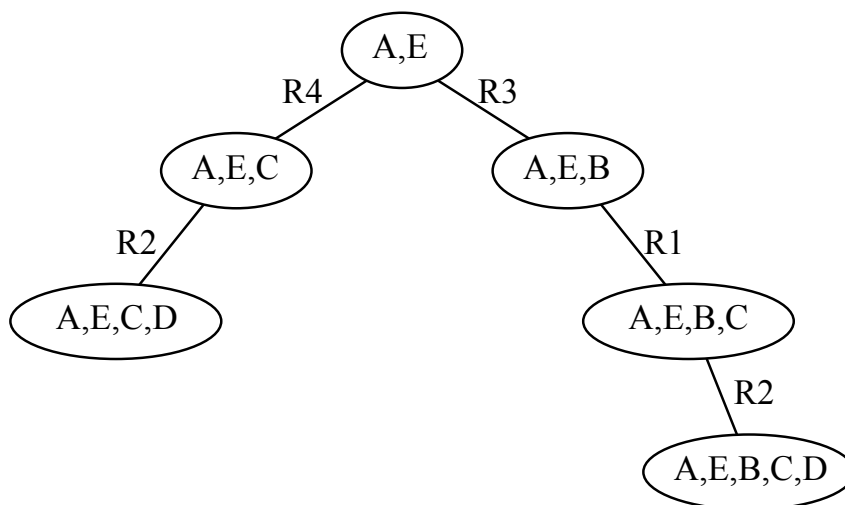
Memoria de lucru contine initial faptele A si E care reprezinta datele de caz ale instantei problemei de rezolvat, si starea scop D, D reprezentind solutia cautata.

Aplicind inlantuirea inainte a regulilor se poate gasi secventa de reguli R3, R1, R2, prezentata in Figura 3 (a), care produce in memoria de lucru stare scop D cautata. Considerind toate regulile aplicabile la un moment dat, exemplul genereaza arborele de cautare in spatiul starilor din Figura 3 (b). Se observa ca starea initiala, definita prin continutul A, E al memoriei de lucru, creeaza multimea de conflicte {R4, R3}. Fiecare aplicare de regula, pe o cale de cautare, va adauga noi fapte la memoria de lucru astfel incit, in final, aceasta contine fie secventa A, E, C, D fie A, E, B, C, D. Din spatiul de cautare al solutiei s-au eliminat aplicari de reguli care nu modifica continutul memoriei de lucru, de exemplu nu s-a figurat regula R4 care s-ar fi putut aplica secventei A, E, C dar care nu ar fi produs nici o schimbare.

Continut initial al memoriei de lucru: A,E
Stare scop: D

$$A,E \xrightarrow{R3} A,E,B \xrightarrow{R1} A,E,B,C \xrightarrow{R2} A,E,B,C,D$$

(a)



(b)

Figura 3. Inlantuirea inainte a regulilor de productie

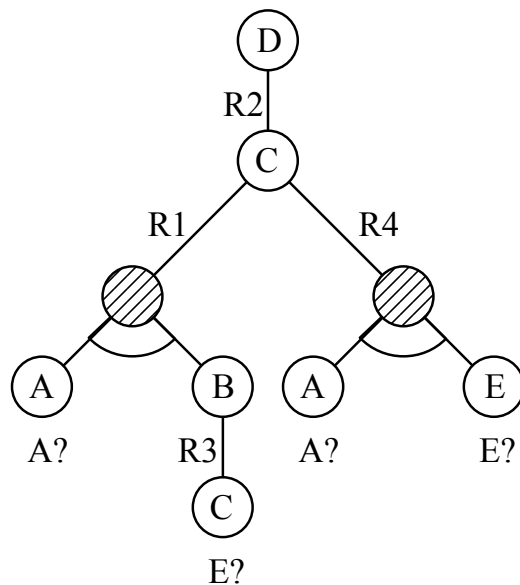
Aplicind inlantuirea inapoi a regulilor de productie, se incearca satisfacerea scopului D prin aplicarea unei reguli care mentioneaza in concluzie D. O astfel de regula este R2. Pentru ca R2 sa poata fi executata trebuie ca memoria de lucru sa contina faptele care satisfac premisa regulii. Pentru aceasta trebuie indeplinite scopurile din premisa, deci cautate regulile care refera aceste scopuri in concluzie. In cazul in care nu exista astfel de reguli, fie faptele sint deja in memoria de lucru, si regula poate fi direct aplicata, fie, in caz contrar, se solicita utilizatorului informatii despre adevarul acestor fapte. O posibila secventa de satisfacere a scopului D prin aplicarea inlantuirii inapoi a regulilor este prezentata in Figura 4 (a). Daca se considera toate posibilitatile de satisfacere a scopului D se obtine arborele SI/SAU din Figura 4 (b).

Continut initial al memoriei de lucru: A,E

Stare scop: D

$D? \xrightarrow{R2} C? \xrightarrow{R1} A?, B? \xrightarrow{R3} A?, E?$

(a)



(b)

Figura 4. Inlantuirea inapoi a regulilor de productie

Observatie: Acest material a fost extras din cartea:

A. Florea, A. Boangiu. "Elemente de Inteligenta Artificiala", http://turing.cs.pub.ro/blia_08/