

1.5. Sisteme Fuzzy

1.5.1. Considerații generale

[www-12] [www-13] [www-14] [www-20]

Specificul unui sistem fuzzy constă în faptul că poate controla simultan date numerice și cunoștințe lexicale. Reprezintă în fapt o transformare neliniară aplicată vectorului datelor de intrare într-o ieșire scalară.

O mulțime fuzzy (sau vagă) este o mulțime despre care nu se știe prea multe lucruri exacte. Expertul uman trebuie să aibă abilitatea de a obține raționamente eficiente chiar prin exploatarea imprecisului, a informațiilor incomplete și nesigure. Teoria mulțimilor fuzzy ajută la transformarea raționamentelor umane calitative în expresii numerice cantitative.

Se disting în general două forme de cunoaștere:

- Cunoașterea obiectivă, utilizată în toate formulările problemelor ingineresti (de exemplu modele matematice)
- Cunoașterea subiectivă, care reprezintă formularea lingvistică a informației, de regulă imposibil de cuantificat și evaluat numeric prin metode tradiționale (de exemplu reguli, informații de expertiză, cerințe de proiectare)

Avantajul unui sistem fuzzy constă în faptul că există extrem de multe posibilități care conduc la loturi de transformări diferite.

Unii cercetători evită utilizarea sistemelor fuzzy, preferând abordări de modele mai familiare în proiectarea aplicațiilor. Sunt însă unele caracteristici care fac din fuzzy o abordare atrăgătoare. Expresia ADEVĂRAT/ FALS care stă la baza aplicațiilor tradiționale, în cadrul unui sistem fuzzy este înlocuită cu exprimări calitative mult mai nuanțate. În tabelul 1.4 sunt date câteva exemple în acest sens.

Tabelul 1.4

Termen	Utilizare în context
Asemănare	de loc, puțin, foarte
Corelare	mică, medie, mare, perfectă
Eroare	mare, medie, mică, nu prea mare, foarte mare, foarte mică aproape zero
Eșantionare	rată mică, rată mare, rată foarte mare

În fig. 1.30 este redată structura clasică a unui sistem fuzzy compus din cele patru componente de bază: fuzificator, reguli, motor de inferență, defuzificator. O dată ce regulile au fost stabilite, un sistem fuzzy poate fi privit ca o transformare intrare ieșire, exprimabilă cantitativ în forma $y=f(x)$.

Regulile furnizate de către experți sunt exprimate prin propoziții de forma DACĂ ... ATUNCI ... cum ar fi: **DACĂ x_1 este foarte întârziat ȘI x_2 este foarte mic ATUNCI urmărește planificator_varianta_2**. Regula din exemplul anterior reliefează câteva aspecte cum ar fi :

- Transpunerea variabilelor lingvistice în corespondențele lor numerice (ex. *foarte întârziat* poate însemna că activitatea este întârziată cu mai mult de 7 zile)
- Variabilele lingvistice au fiecare o gamă finită de termeni (de exemplu progresul activității poate varia de la extrem de întârziat la extrem de timpuriu cu câteva intervale intermediare)

- Conexiuni logice ale variabilelor lingvistice (de tipul SI, SAU)

Fuzificatorul are rolul de a transpune exprimările numerice în seturi fuzzy, necesare pentru activarea regulilor, care la rândul lor au asociate valorilor lingvistice seturi fuzzy corespunzătoare.

Motorul de inferență aplică o transformare a seturilor de REGULI în seturi fuzzy. Aici se implementează modulul de tratare al regulilor.

În majoritatea aplicațiilor, este necesară transformarea reciprocă din seturi fuzzy în valori numerice, sarcină pe care o are *defuzificatorul*.

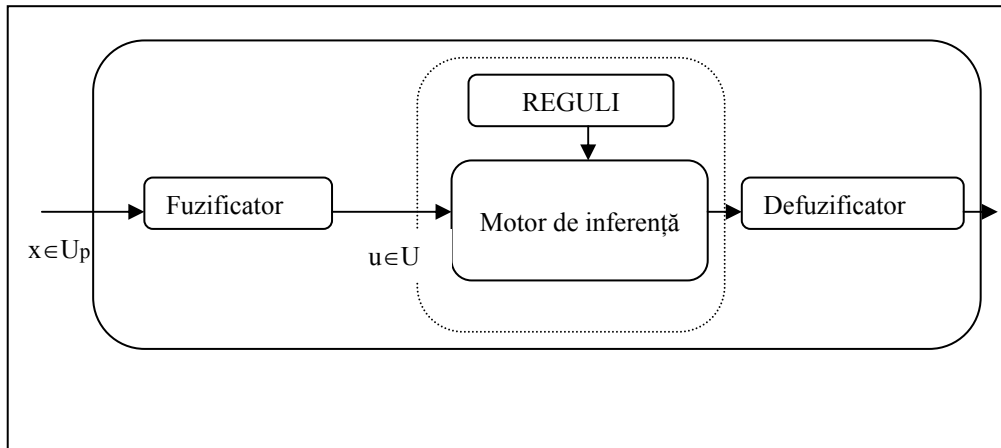


Fig. 1.30. Sistem Expert bazat pe logica Fuzzy

1.5.2. Logica Fuzzy

Logica fuzzy, introdusă de Dr. Lotfi Zadeh [www-20] este un superset al logicii convenționale boolene, logică care a fost extinsă pentru a cuprinde conceptul adevărului parțial – valori ale adevărului cuprinse între “complet adevărat” și “complet fals”.

Sistemele inteligente bazate pe logica bivalentă consideră că un obiect aparține unei mulțimi. Sistemele inteligente bazate pe logica fuzzy consideră că acel obiect poate aparține unei mulțimi, în grade diferite, generându-se o flexibilitate a interpretării situațiilor.

Există o strânsă corelare între logica fuzzy și conceptul de subset, în literatura de specialitate întâlnindu-se cel mai des termenii de “*subset*” și “*set*”, care au aceeași semnificație în context cu cei de “*submulțime*” respectiv “*mulțime*”.

Un subset U al setului S poate fi definit ca un set de perechi ordonate, cu câte un element aparținând setului S iar celălalt element inclus în setul $\{0,1\}$, cu câte o pereche ordonată pentru fiecare element al lui S .

Această regulă stabilește corespondența între elementele setului S și setul $\{0,1\}$. Valoarea 0 este utilizată pentru a reprezenta neapartenența, iar valoarea 1 pentru a reprezenta apartenența. Pentru a stabili dacă propoziția “ x este în U ” este adevărată sau falsă, se caută în perechile ordonate, astfel încât primul element al perechii să fie x . Propoziția este adevărată dacă cel de-al doilea element al perechii este unu, respectiv este falsă dacă acesta este zero.

În mod similar, un subset fuzzy F al setului S poate fi definit ca un set de perechi ordonate, care au fiecare câte un prim element aparținând lui S , iar cel de-al doilea element este o valoare în intervalul $[0,1]$. Există câte o pereche pentru fiecare element al lui S . Valoarea „0” este utilizată pentru a reprezenta neapartenența, valoarea „1” pentru a reprezenta apartenența totală, iar valorile intermediare sunt utilizate

pentru a reprezenta grade de apartenență intermediare. În mod frecvent, apartenența este descrisă de o funcție – *funcția de apartenență* a lui F. Gradul de adevăr în propoziția “x este în F” se determină prin găsirea celui de-al doilea element al perechii ordonate al cărui prim element este X.

Trăsături specifice ale logicii Fuzzy:

- În logica fuzzy raționamentul exact este privit ca un caz limită al raționamentului aproximativ ;
- În logica fuzzy orice situație este exprimată gradual;
- Orice sistem logic poate fi reprezentat în logica fuzzy (fuzzyficare);
- În logica fuzzy cunoașterea este interpretată ca o colecție de restricții fuzzy elastice sau de echivalență pe o colecție de variabile;
- Deducția este privită ca un proces de propagare al restricțiilor elastice .
- Mulțimea fuzzy reprezintă are asociată o funcție caracteristică care ia valori în intervalul [0,1], valorile acesteia descriind gradul de apartenență al unui element la acea mulțime

Variabile lingvistice

În logica fuzzy domeniile corespund conceptelor de *variabile lingvistice*. Unei variabile lingvistice, îi corespunde un domeniu de valori posibile, care sunt definite pe un interval numit *univers de discurs* pentru variabila lingvistică respectivă. O problemă care utilizează variabile lingvistice implică stabilirea mulțimii termenilor lingvistici asociați fiecărei variabile.

O variabilă lingvistică este definită de un quintuplu format din : x , $T(x)$, σ , U , M , unde:

- x - este denumirea variabilei lingvistice;
- $T(x)$ este mulțimea nevăgă a denumirilor pentru valorile diferite ale variabilei lingvistice, adică un termen al variabilei lingvistice A ;
- σ este operatorul sau cuantificatorul, care se aplică conform regulilor sintactice și care generează termenii variabilei lingvistice;
- U este universul de discurs al aplicației;
- M este operator semantic, fiind utilizat pentru fiecare atașare a unei submulțimi vagi din Universul de discurs, fiecărui termen T al variabilelor lingvistice.

1.5.3. Analiză comparativă între logica Fuzzy și sistemele logice tradiționale

Atât sistemele tradiționale bazate pe logica naturală (bivalentă) cât și sistemele bazate pe logica fuzzy efectuează raționamente logice, care simulează inteligența artificială. Sistemele bazate pe logica fuzzy utilizează date inexacte sau parțial corecte, în timp ce sistemele bazate pe logica naturală nu pot funcționa dacă se modifică modelul procesului, sistemele cu reguli de deducție având dificultăți în funcționare dacă intervin situații contradictorii. Sistemele bazate pe logica fuzzy sunt adecvate pentru recunoașterea de caractere, de forme, etc, pe când sistemele bazate pe logica naturală se pretează mult mai bine pentru analiza datelor financiare, respectiv pentru a naviga în scopul identificării unei strategii potrivite. Câteva diferențe esențiale ale celor două abordări sunt prezentate în Tabelul 1.5.

Tabelul 1.5 Diferențe dintre logica naturală și logica fuzzy

Sisteme inteligente bazate pe logica naturală		
Sisteme inteligente bazate pe logica fuzzy		
adevărul	→	<i>Adevărat ; Fals</i>
	→	Valori lingvistice: <i>Puțin adevărat, Aproape adevărat, Adevărat</i>
cuantificatorii	→	<i>Existențial, Universal</i>
	→	Valori lingvistice: <i>Câteodată, În general, Întotdeauna, Câteva, Mai mulți, Foarte mulți</i>
predicatele	→	<i>Mai mic decât, Mai mare decât</i>
	→	Valori lingvistice: <i>Foarte mic, Puțin mai mic, Aproximativ, Mai mare</i>
modificatori	→	<i>Negație</i>
	→	Valori lingvistice: <i>Puțin, Mai mult, Mult, Foarte mult, Foarte rece, Rece, Moderat, Cald, Foarte cald</i>
probabilități	→	<i>Procentual, Pe un interval</i>
	→	Cuantificatori fuzzy: <i>MaxMin, MinMax</i>
posibilități	→	<i>Procentual, Pe un interval</i>
	→	Valori lingvistice: <i>Imposibil, Puțin posibil, Posibil,</i>

1.5.4. Sisteme Expert Fuzzy

Sistemele expert fuzzy sunt SE care utilizează logica fuzzy în locul logicii tradiționale (bivalente). Cu alte cuvinte, un SE fuzzy reprezintă o colecție de funcții de apartenență și reguli de raționament. Spre deosebire de SE tradiționale, care sunt mașini de raționament simbolic, SE fuzzy sunt orientate mai degrabă spre procesare numerică. Regulile într-un SE fuzzy sunt reprezentate într-o formă asemănătoare cu:

DACĂ x este mic ȘI y este mare ATUNCI $z =$ mediu

unde x și y sunt variabile de intrare; z - este variabilă de ieșire; mic - este o funcție de apartenență (subset fuzzy) definit pentru x ; mare - este o funcție de apartenență definită pentru y , iar mediu - este o funcție de apartenență definită pentru z .

Partea din dreapta lui ATUNCI se numește concluzie sau consecință. Majoritatea aplicațiilor care lucrează cu logica fuzzy permit însă formularea mai multor concluzii pentru fiecare regulă. Totalitatea regulilor este cunoscută ca "bază de reguli" sau "bază de cunoștințe".

Mecanismul prin care regulile și funcțiile de apartenență sunt aplicate datelor de intrare și sunt calculate valorile de ieșire, se numește *inferență* (din engl. "inference" cu semnificația de deducție, raționament)

Sistemele expert fuzzy constau la rândul lor în combinarea a patru subproces: *fuzificare, inferență, compoziție, defuzificare*.

- În procesul de *fuzificare* se aplică funcțiilor de apartenență variabilele de intrare, determinându-se astfel gradul de adevăr al fiecărei premise componente a regulilor. Un grad de adevăr diferit de zero va avea ca efect luarea în considerare a premiselor, deci regula va fi aplicată.

- În procesul de *inferență* sunt calculate valorile de adevăr ale premiselor, în funcție de care vor fi activate regulile corespunzătoare acestora din baza de cunoștințe. Concluziilor regulilor activate le corespunde câte un subset fuzzy atașat ieșirilor sistemului.
- *Compoziția* reprezintă procesul în care toate subseturile fuzzy corespunzătoare câte unei ieșiri sunt combinate între ele, rezultând un singur subset fuzzy pentru ieșire.
- Procesul de *defuzificare* nu apare întotdeauna în aplicații. El constă în transpunerea unui subset fuzzy într-o singură valoare corespunzătoare unei ieșiri.

1.5.4.1 Conectorii și operatorii mulțimilor Fuzzy

Informația fuzzy este tratată cu ajutorul conectorilor lingvistici, corespunzători termenilor lingvistici, prin intermediul cărora se caracterizează mulțimea fuzzy. Tratarea matematică a conexiunilor se bazează pe următorii operatori ai mulțimilor fuzzy:

- **Conectorul ȘI** este asociat cu *intersecția* mulțimilor fuzzy. Considerând mulțimile fuzzy μ_1, μ_2, \dots definite pe mulțimea X , relația corespunzătoare intersecției este:

$$\mu = \mu_1 \cap \mu_2 : X \rightarrow [0,1],$$

fiind evaluată prin operatorul **MIN** (minimum) conform relației:

$$\mu(x) = (\mu_1 \cap \mu_2)(x) = \text{MIN}(\mu_1, \mu_2), \forall x \in X$$

Proprietățile conectorului **ȘI**, respectiv ale operatorului de evaluare **MIN** sunt:

- Comutativitatea $\mu_1 \cap \mu_2 = \mu_2 \cap \mu_1$
- Asociativitate $\mu_1 \cap \mu_2 \cap \mu_3 = (\mu_1 \cap \mu_2) \cap \mu_3 = \dots$

- **Conectorul SAU** este asociat cu *reuniunea* mulțimilor fuzzy. Considerând mulțimile fuzzy μ_1, μ_2, \dots definite pe mulțimea X , relația corespunzătoare reuniunii este:

$$\mu = \mu_1 \cup \mu_2 : X \rightarrow [0,1],$$

fiind evaluată prin operatorul **MAX** (maximum) conform relației:

$$\mu(x) = (\mu_1 \cup \mu_2)(x) = \text{MAX}(\mu_1, \mu_2), \forall x \in X$$

Proprietățile conectorului **SAU**, respectiv ale operatorului de evaluare **MAX** sunt:

- Comutativitatea $\mu_1 \cup \mu_2 = \mu_2 \cup \mu_1$
- Asociativitate $\mu_1 \cup \mu_2 \cup \mu_3 = (\mu_1 \cup \mu_2) \cup \mu_3 = \dots$

- **Operatorul de negare fuzzy (complementare)**

Dacă $\mu : X \rightarrow [0,1]$ este o mulțime fuzzy, se definește complementul fuzzy, având notația $\mu^c, \mu^c : X \rightarrow [0,1]$ și respectiv proprietatea: $\mu^c(x) = 1 - \mu(x), \forall x \in X$

- **Operatorul PRODUS (PROD)**

Operatorul **PROD** reprezintă alternativa evaluării operatorului **MIN**, fiind definit și evaluat conform relației:

$$\mu(x) = \text{PROD}(\mu_1(x), \mu_2(x)) = \mu_1(x) \cdot \mu_2(x), \forall x \in X$$

▪ Operatorul SUMĂ (SUM)

Operatorul SUM reprezintă alternativa evaluării operatorului MAX , fiind definit și evaluat conform relației:

$$\mu(x) = (1-m)[\mu_1(x) + \mu_2(x) + \dots + \mu_m(x)], \forall x \in X$$

Orice operație logică (ȘI, SAU) poate fi transpusă, în logica fuzzy sub forma unei aplicații definite în felul următor: $[0,1] \times [0,1] \rightarrow [0,1]$. Sub această formă se determină valoarea de adevăr a unei formule compuse, în funcție de valorile de adevăr ale subformulelor înălțuite prin operația logică respectivă. În cazul sistemelor bazate pe logica fuzzy operațiile logice (ȘI, SAU) nu mai sunt definite cu ajutorul tabelelor de adevăr.

1.5.4.2 Reprezentarea cunoașterii prin logica Fuzzy

Încercările convenționale de reprezentare a cunoașterii prin rețele semantice, cadre, calculul predicatelor și PROLOG se bazează pe logica bivalentă. Un impediment serios al acestor tehnici este inabilitatea de a reprezenta incertitudinea și imprecizia. Reprezentarea convențională nu oferă un model adecvat pentru metodele de raționament , care sunt mai degrabă aproximative decât exacte. Majoritatea metodelor raționamentului uman și toate metodele raționamentului comun se înscriu în această categorie .

Logica Fuzzy, care poate fi privită ca o extensie a sistemelor logice clasice , oferă un cadru conceptual, care abordează problema reprezentării cunoașterii, într-un mediu al incertitudinii și impreciziei

Esența reprezentării în logica Fuzzy se bazează pe semanticile test-scor, în cadrul cărora o propoziție este interpretată ca un sistem de restricții elastice, iar raționamentul este privit ca o propagare a restricțiilor elastice .

1.5.4.3 Aplicație a inferenței Fuzzy în Management

[Pre-97] [Pro-99g] [Pro-01b],[Pro-01d]

Există un număr mare de situații în care apartenența unei anumite valori la o mulțime dată nu poate fi caracterizată prin funcții caracteristice. Astfel, luând ca exemplu anii de recuperare a investiției într-un proiect, acesta poate fi împărțit calitativ prin 3 noțiuni: durata de recuperare – DR mică, DR medie, DR mare. (Fig.1.31)

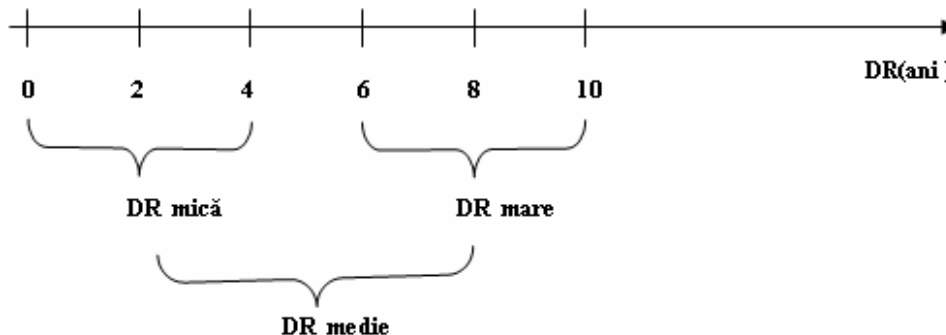


Fig. 1.31. Împărțirea calitativă a domeniului Durată de recuperare (de la 0 la 10 ani) prin 3 noțiuni „mică”, „medie” și „mare”

Apar următoarele situații: DR=3 ani, pe de-o parte mai este încă DR mică, pe de altă parte este și DR medie. Deci DR = 3 ani aparține atât mulțimii MICĂ cât și mulțimii MEDIE. Ce se poate spune comparativ despre duratele de recuperare DR₁=3 ani și 11 luni și DR₂=5 ani? Amândouă valori aparțin mulțimii MEDIE, dar în măsuri diferite.

Termenii MICĂ, MEDIE, MARE, sunt *variabilele lingvistice*. Intervalele anilor la care se referă sunt *mulțimile vagi*. Aparența unei anumite DR la o mulțime se caracterizează prin *funcția de apartenență*, care gradează apartenența valorilor la mulțime. S-a convenit ca domeniul (mulțimea care gradează apartenența) să fie intervalul [0,1]. Devine astfel posibilă reprezentarea din figură Fig.1.32, care evidențiază 3 forme diferite de funcții de apartenență:

- trapezoidală
- triunghiulară simetrică
- triunghiulară asimetrică

Împărțirea pe “ani” este subiectivă, un alt “expert” ar putea caracteriza aceleași mulțimi vagi prin alte funcții de apartenență.

Valoarea $\mu^{DR}(x)$ reprezintă *gradul de apartenență* a lui x la mulțimea DR.

Evaluarea investiției într-un proiect bazată pe logica fuzzy, utilizează funcțiile de apartenență corespunzătoare acoperirii domeniului de variație a diverselor mărimi, cu valori lingvistice atât pentru mărimile de intrare cât și pentru mărimile de ieșire.

Domeniul total de variație al unei mărimi este numit în general *univers de discurs*. De exemplu pentru DR, universul de discurs (Fig.1.31) este de la 0 la 10 ani.

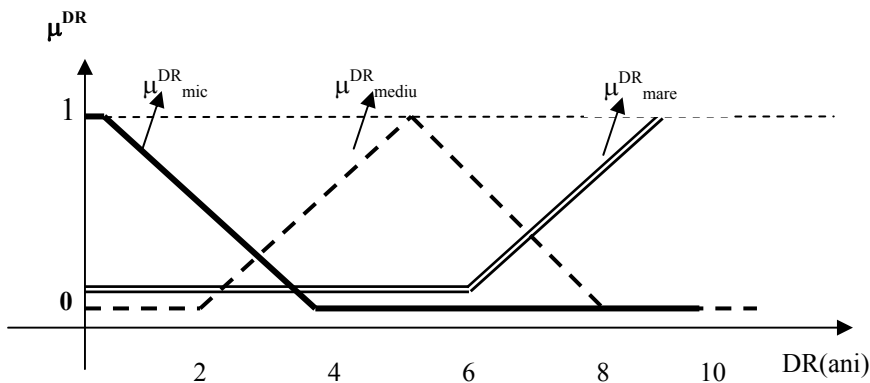


Fig. 1.32. Funcții de transfer

Aplicație: Analiza Riscului Proiectelor de Investiție

În cele ce urmează se consideră o aplicație a abordărilor fuzzy într-o problemă de Analiză a Riscului Proiectelor de Investiții, în care se urmărește Rata de Randament (RR) posibil a unui proiect de investiții, având ca date de intrare Durata de Recuperare (DR) estimată a investiției în proiect. Universul de discurs al celor 2 variabile DR și RR poate fi acoperit în acest caz, prin 5 valori lingvistice, fig.1.33.

1. FR – foarte redusă
2. R – redusă
3. m – medie
4. M – mare
5. FM – foarte mare

Cazul inferenței o intrare – o ieșire și mai multe reguli

Se consideră următoarea bază de reguli:

- R1: DACĂ (DR =FR) ATUNCI (RR=FM)
- R2: DACĂ (DR= R) ATUNCI (RR=M)
- R3: DACĂ (DR=m) ATUNCI (RR=m)
- R4: DACĂ (DR=M) ATUNCI (RR=R)
- R5: DACĂ (DR=FM) ATUNCI (RR=FR)

Durata de recuperare se consideră **DR=3ani și 6 luni**, al cărui univers de discurs este acoperit cu 2 termeni lingvistici **R și m**, având funcțiile de apartenență μ^{DR}_R și μ^{DR}_m . În acest caz se vor activa regulile R2 și R3.

- R2: DACĂ (DR= R) ATUNCI (RR=M)
- R3: DACĂ (DR=m) ATUNCI (RR=m)

▪ **Etapale inferenței**

1. Fuzificarea valorii ferme DR = 3 ani și 6 luni
2. Analiza regulilor activate: R2 și R3
3. Stabilirea gradelor de apartenență H_2, H_3 , la TL ce aparțin lui μ^{DR}_R și μ^{DR}_m
4. Stabilirea funcției de apartenență pentru TL din concluzia fiecărei reguli μ^{RR}_M și μ^{RR}_m
5. Stabilirea funcției de apartenență pentru ieșire $\mu^{RR}_{rez}(RR) = \text{MAX} (\text{MIN} (H_i, \mu^{RR}_i(RR)))$

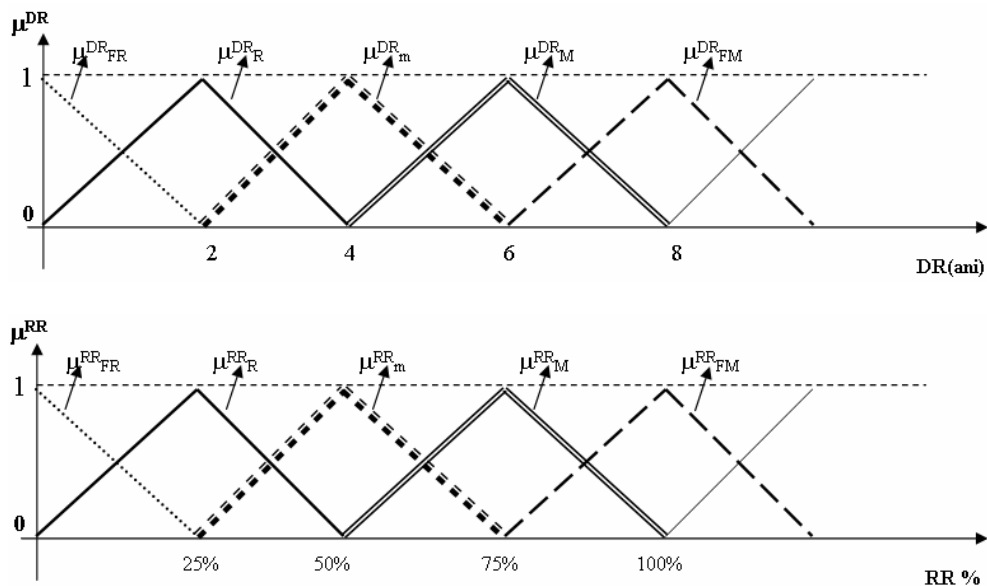


Fig.1.33. Funcțiile de apartenență pentru variabilele de intrare-ieșire

DR – durata de recuperare a investiției
 RR- rata de randament necesar

1. Fuzificarea valorii ferme

$$DR = 3 \text{ ani și } 6 \text{ luni} \longrightarrow DR^* = \begin{pmatrix} \mu^{DR}_{FR}(t) \\ \mu^{DR}_R(t) \\ \mu^{DR}_m(t) \\ \mu^{DR}_M(t) \\ \mu^{DR}_{FM}(t) \end{pmatrix} = \begin{pmatrix} 0,0 \\ 0,25 \\ 0,75 \\ 0,0 \\ 0,0 \end{pmatrix}$$

R2: DACĂ (DR= R) ATUNCI (RR=M)
R3: DACĂ (DR=m) ATUNCI (RR=m)

3. Stabilirea gradelor de apartenență H_2, H_3 , la TL: μ^{DR}_R și μ^{DR}_m , (Fig. 1.33)

4. Stabilirea funcției de apartenență pentru TL din concluzia fiecărei reguli μ^{RR}_M și μ^{RR}_m , (Fig. 1.33)

5. Stabilirea funcției de apartenență pentru ieșire

$$\mu^{RR}_{rez.}(RR) = \text{MAX} (\text{MIN} (H_i, \mu^{RR}_i(RR))) \text{ (Fig. 1.33)}$$

Inferența se bazează pe operații cu mulțimi vagi exprimate prin operații efectuate asupra funcțiilor de apartenență, aplicându-se operatorii combinați MAX(MIN), adică hașura maximă obținută prin reunirea minimelor hașurate (concluziile fiecărei reguli). (§ 1.5.5.1). Inferența se soldează cu determinarea caracteristicii MAX(MIN) hașurate.

Defuzificarea constă în a asocia o valoare concretă variabilei RR, care să corespundă ariei hașurate. Metoda din exemplu este cea a centrului de greutate, adică se consideră că RR trebuie să ia valoarea corespunzătoare abscisei centrului de greutate al ariei hașurate.

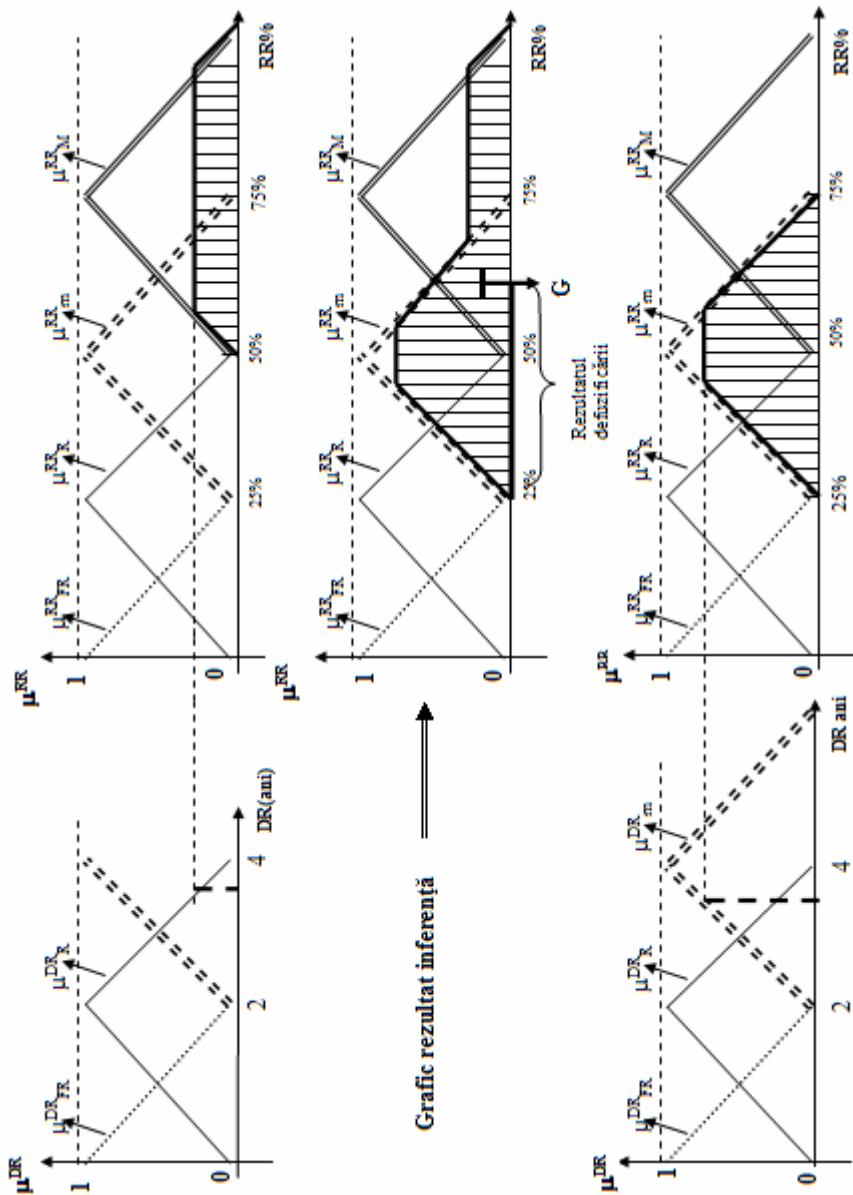


Fig.1.34. Rezultatul inferenței

1.6. Implementarea tehnicilor bazate pe logica fuzzy în modulul fuzzy toolbox al mediului de programare MATLAB 7.4.0

O posibilitate de implementare a tehnicilor bazate pe logica fuzzy este oferită de mediul de programare MATLAB 7.4.0. Vizualizările modului "Fuzzy toolbox" sunt foarte accesibile utilizatorilor, facilitând introducerea, organizarea și prezentarea informațiilor sistemului bazat pe logica fuzzy, prin proceduri de editare a intrărilor și ieșirilor, de configurare a funcțiilor de apartenență, respectiv de editare a bazei de cunoștințe. În continuare este prezentat modul de utilizare a modului Fuzzy toolbox, implementând aplicația rezolvată mai sus.

După lansarea programului MATLAB 7.4.0, în fereastra de comandă „Command Window” (fig.1.35) se tastează cuvântul <<fuzzy>> și se apasă tasta **Enter**. Sistemul afișează Editorul de tip FIS, care procesează informația corespunzătoare Sistemelor bazate pe inferența Fuzzy. În partea superioară se

afișează în mod grafic diagrama sistemului, care urmează a fi creat, având intrarea și ieșirea etichetate. (fig.1.36)

Sub diagramă este un câmp de tip text care afișează denumirea fișierului, având extensia FIS.

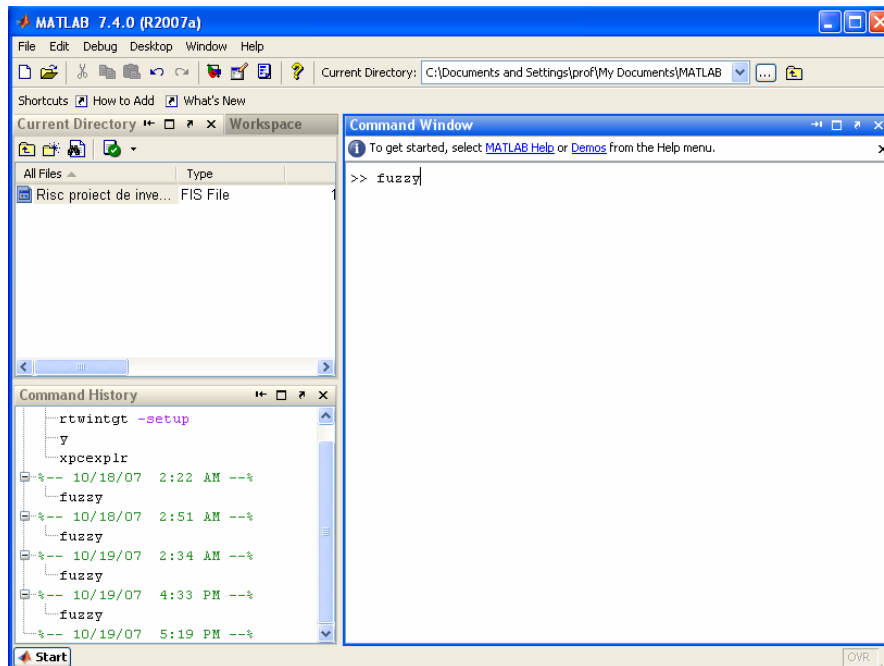


Fig.1.35. Fereastra de comandă

În partea inferioară stângă a ferestrei se află o serie de meniuri derulante, care-i permit utilizatorului să specifice operatorii care urmează să fie aplicați în cadrul procesului. În partea inferioară dreapta sunt afișate câmpuri, care furnizează informații despre ariabila selectată în partea superioară a interfeței.

Variabilele de intrare sunt afișate în caseta din partea stângă a sistemului (ex. Durata_recuperare), la mijloc este afișată caseta regulilor de inferență, respectiv în partea dreaptă este afișată caseta variabilelor de ieșire (ex. Rata_randament).

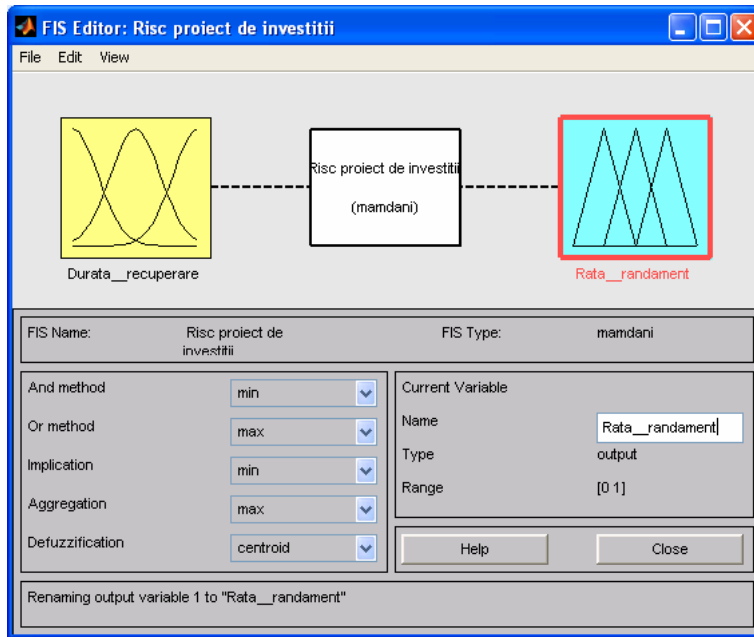


Fig.1.36. Editorul „FIS”

Utilizatorul are posibilitatea definirii mai multor variabile de intrare, respectiv a mai multor variabile de ieșire.

Procedura de introducere a variabilelor de intrare (ieșire)

- Din meniul **Edit** se selectează **Add Variable**, respective **Input** (pentru adăugarea variabilelor de intrare) sau **Output** (pentru adăugarea variabilelor de ieșire) (Fig.1.37)

Procedura de ștergere a variabilelor de intrare (ieșire)

- Dacă se dorește ștergerea unei variabile de intrare (ieșire), se selectează caseta grafică a variabilei respective, iar din meniul **Edit** se selectează **Remove Selected Variable**, sau de la tastatura se tastează **Ctrl+X**.

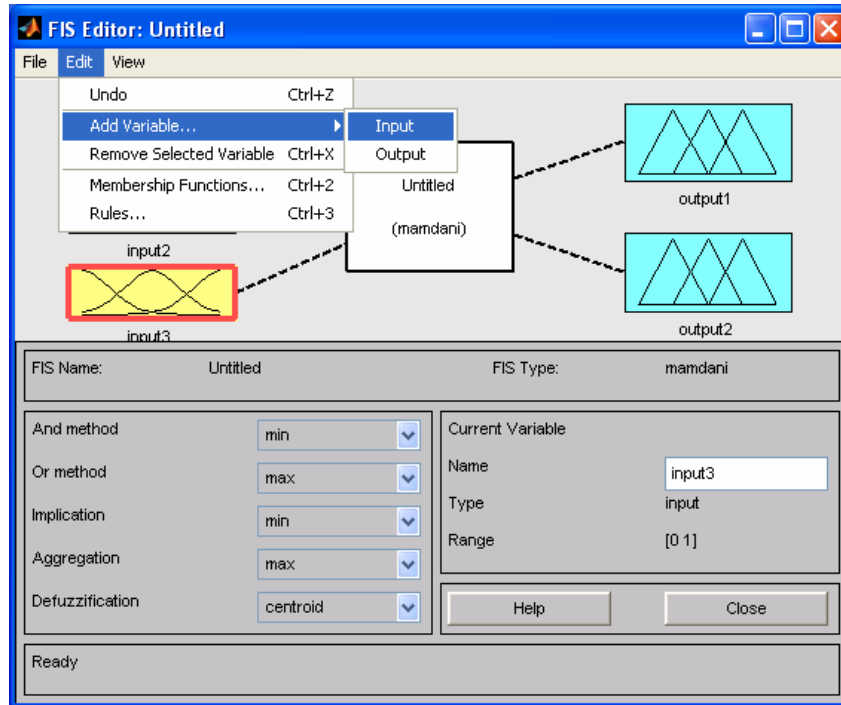


Fig.1.37. Adăugarea variabilelor

În urma introducerii variabilelor de intrare și de ieșire, se stabilesc denumirile acestora.

Procedura de denumire a variabilelor de intrare (ieșire)

- Se selectează caseta grafică a unei variabile și se introduce numele acesteia în partea dreaptă – jos a Editorului de tip FIS, în caseta **Name**.

După definirea numelui variabilelor de intrare și ieșire se definesc funcțiile de apartenență și universal de discurs pentru fiecare variabilă în parte.

Procedura de definire a funcțiilor de apartenență și a universului de discurs

- Se selectează variabila de intrare (ieșire), care urmează să fie configurată
- Din meniul **Edit** se selectează **Membership Functions**,
sau
- Se face dublu clic cu mouse-ul pe caseta grafică variabilei respective. (Fig.1.38)
MATLAB 7.4.0 afișează o nouă interfață grafică denumită **Membership Function**.

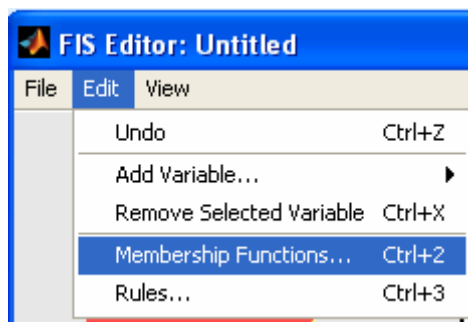


Fig.1.38. Adăugarea funcțiilor de apartenență

Editorul funcțiilor de apartenență (Membership Function Editor)

Editorul **Membership Function Editor** (fig.1.39) se utilizează pentru a crea, anula sau modifica funcțiile de apartenență ale sistemului fuzzy.

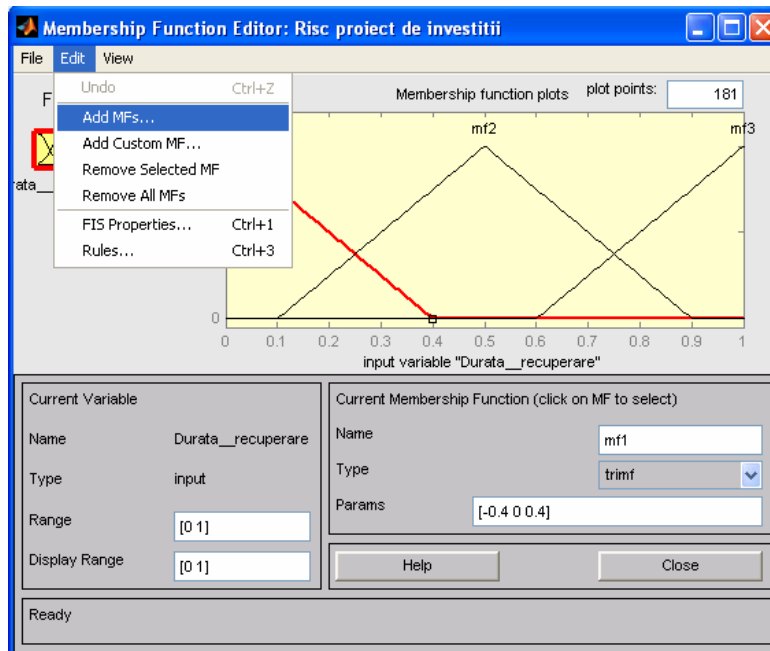


Fig. 1.39. Editorul „ Membership Function”

- Din meniul **Edit** al interfeței grafice **Membership Function Editor** se selectează **Add MFs....**
- În caseta de dialog **Membership Functions** se derulează lista **Number of MFs** și se selectează numărul de funcții de apartenență, pe care le are variabila respectivă.
- *Opțional:* În caseta **Membership Functions** se poate stabili forma generală a funcțiilor de apartenență, care poate fi triunghiulară (**trimf**), trapezoidală (**trapmf**), clopot (**gbellmf**), gaussiană (**gaussmf**), etc. Tipul formei stabilite se aplică tuturor funcțiilor de apartenență, utilizatorul urmând să diferențieze în pașii care urmează definirea formelor funcțiilor de apartenență (dacă este cazul).

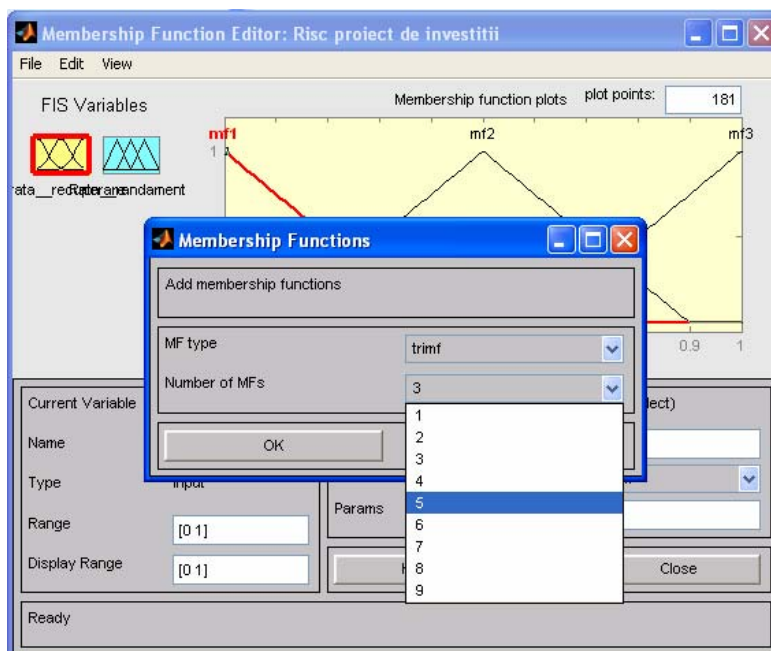


Fig.1.40. Incrementarea numărului de funcții de apartenență

- Universul de discurs se definește în caseta **Range** din parte stângă- jos a editorului **Membership Function**.
- În caseta **Display Range** (plasată sub **Range**) utilizatorul poate opta pentru afișarea întregului univers de discurs (caz în care se introduc aceleași cifre ca și în caseta **Range**), sau poate opta pentru afișarea unei secvențe din cadrul universului de discurs (caz în care se introduce cifre din interiorul intervalului afișat în caseta **Range**).
- Parametrii care definesc geometria funcției de apartenență se configurează în caseta **Params** din partea dreaptă – jos a editorului **Membership Function**

Luând exemplul aplicației “Risc proiect de investiții” (§.1.5.4.3.),(Fig.1.40) au fost selectate 5 funcții de apartenență, care au fost denumite cu următoarele variabile lingvistice: *Foarte_redușă*, *Redușă*, *Medie*, *Mare*, și *Foarte_mare*. Pentru definirea fiecărei variabile lingvistice, se selectează câte o funcție de apartenență, urmând a se atribui denumirea în caseta **Name** din partea dreaptă jos.

Pentru redefinirea formei fiecărei funcții de apartenență se selectează câte o funcție, după care i se selectează profilul geometric din lista derulantă “**Type**” din partea dreaptă – jos a editorului **Membership Function**. (Fig.1.41)

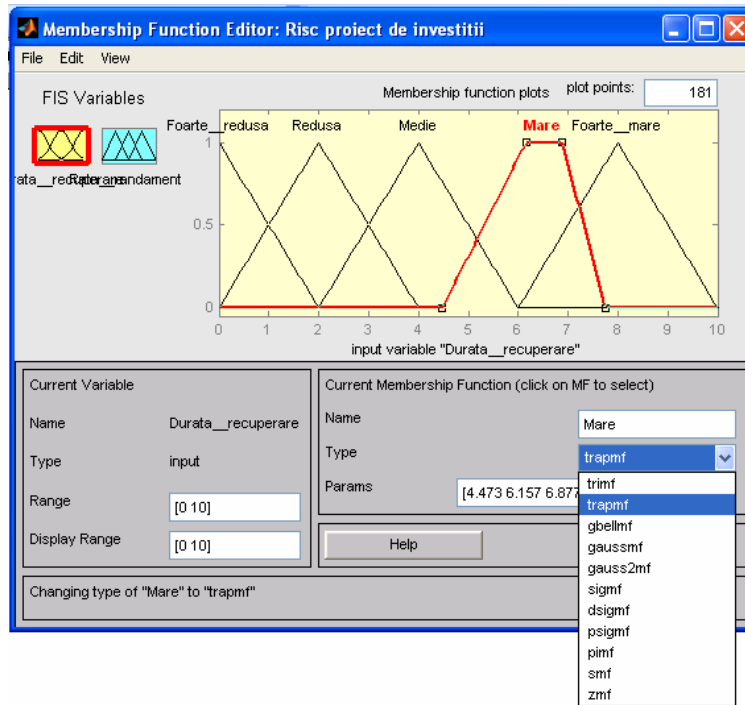


Fig. 1.41. Schimbarea formei funcției de apartenență

Forma unei funcții de apartenență poate fi de asemenea ajustată prin stabilirea parametrilor, (în caseta **Params** din partea dreaptă – jos a editorului **Membership Function**) care definesc geometria fiecărei funcții în cadrul universului de discurs. De exemplu, pentru funcția de apartenență “**Mare**” din fig.1.42, s-au configurat următorii parametrii geometrici: parametrul **4** din cadrul universului de discurs definește limita din partea stângă a bazei triunghiului, parametrul **6** definește vârful triunghiului și parametrul **8** definește limita din partea dreaptă a bazei triunghiului.

În cazul în care funcția de apartenență este un trapez, în caseta **Params** se configurează 4 cifre, având semnificația definirii punctelor geometriei trapezului în ordinea următoare: limita stângă a bazei mari, limita stângă a bazei mici, limita dreaptă a bazei mici, respectiv limita dreaptă a bazei mari.

Parametrii definitorii ai profilului geometric selectat pentru o funcție de apartenență pot fi ajustați și cu ajutorul mouse-ului.

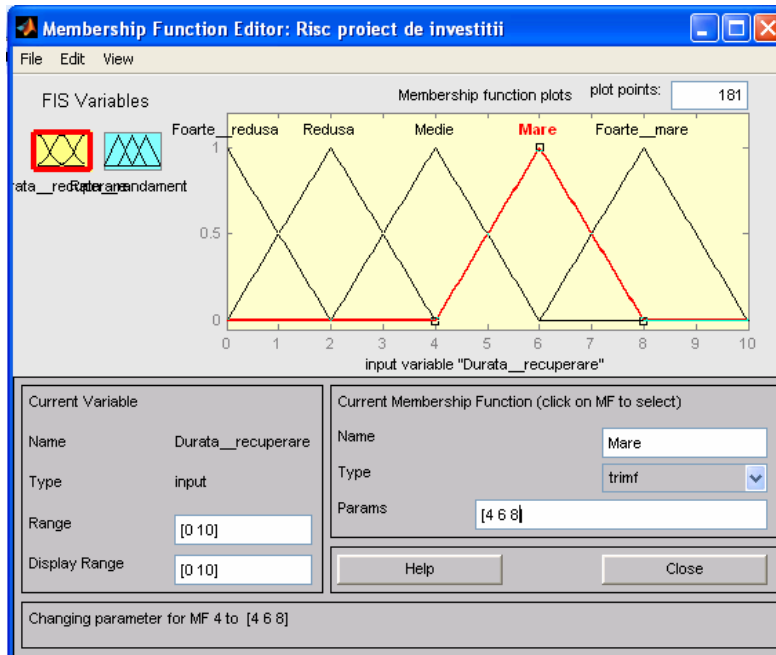


Fig. 1.42. Configurarea parametrilor pentru funcțiile de apartenență

Procedura de ajustare cu mouse-ul a profilului geometric pentru o funcție de apartenență:

- Se selectează funcția de apartenență (fig. 1.43)
- Se selectează cu mouse-ul vârfurile profilului geometric al funcției de apartenență și se draghează în stânga sau dreapta universului de discurs. Parametrii din caseta **Params** din partea dreaptă – jos a editorului **Membership Function** se reconfigurează automat.

Procedura de definire a universului de discurs se repetă pentru toate variabilele de intrare, respectiv de ieșire.

Procedura de definire a funcțiilor de apartenență se repetă pentru toate funcțiile de apartenență ale unei variabile de intrare (ieșiere), respectiv pentru toate variabilele de intrare (ieșiere). Fig. 1.44

Orice acțiune poate fi anulată din meniul **Edit** prin selectarea comenzii **Undo**. De asemenea orice funcție de apartenență poate fi stearsă.

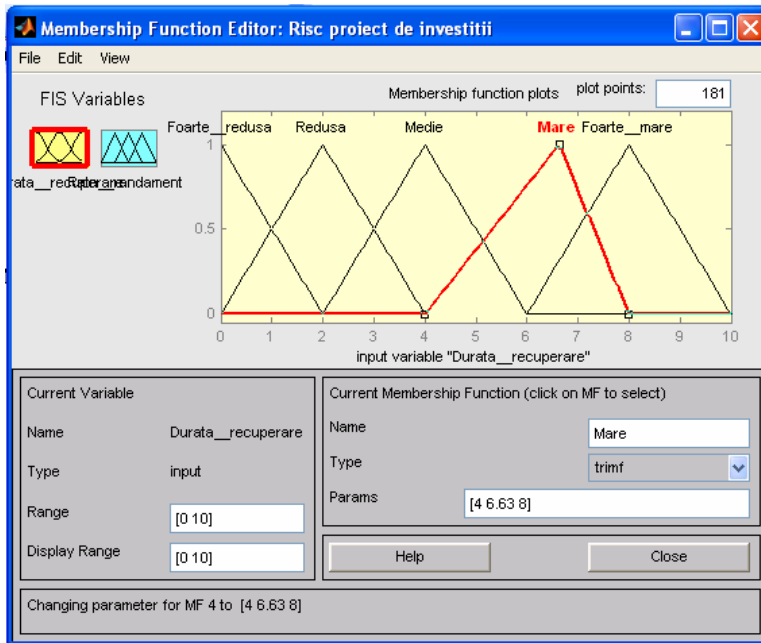


Fig. 1.43. Ajustarea profilului funcției de apartenență

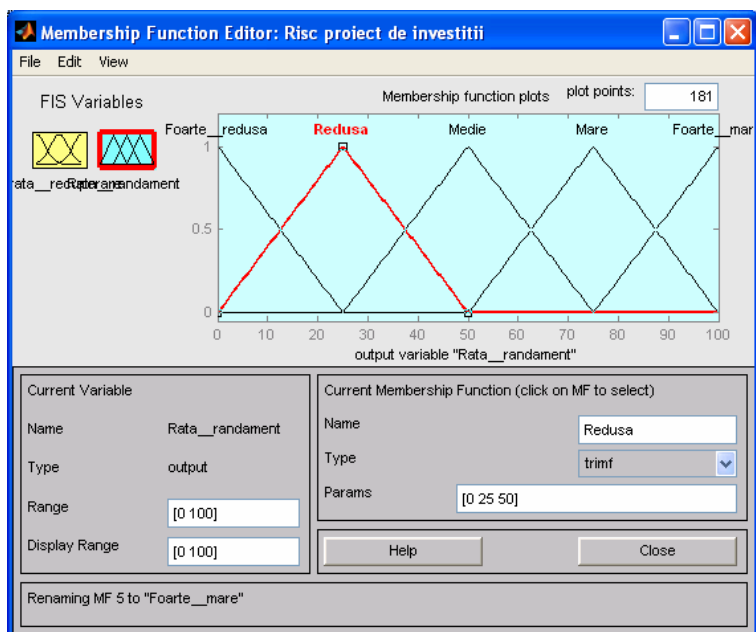


Fig. 1.44. Definire funcțiilor de apartenență pentru variabila de ieșire

Procedura de ștergere a funcțiilor de apartenență

- Se selectează funcția de apartenență, care trebuie ștersă, iar din meniul **Edit** se selectează **Remove Selected MF** (Fig.1.39)

După încheierea procesului de definire a variabilelor de intrare (ieșire) urmează editarea bazei de reguli pentru sistemul Fuzzy.

Procedura de afișare a editorului bazei de cunoștințe:

- Se deschide una din ferestrele de editare **FIS Editor** sau **Memberships Function Editor**

- Din meniul **Edit** se selectează **Rules...** sau se tastează **Ctrl+3**
- Programul afișează fereastra de editare **Rule Editor**, (Fig.1.44). Partea superioară a editorului de reguli este destinată afișării bazei de reguli, pe măsura editării acestora. În partea inferioară a editorului sunt afișate variabilele de intrare (în partea stângă) și variabilele de ieșire (în parte dreaptă). Variabilele de intrare sunt plasate sub incidența condiției **If**, iar variabilele de ieșire sunt plasate sub incidența concluziei **Then**.

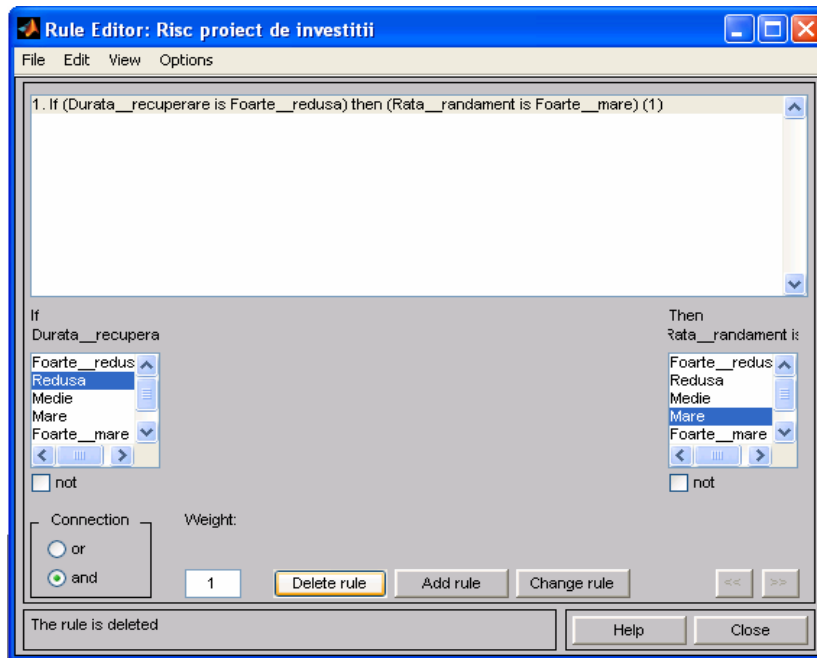


Fig. 1.45. Editorul „Rule”

În cazul existenței mai multor variabile de intrare sau ieșire utilizatorul are posibilitatea selectării conectorilor logici de combinare **or** (sau) sau **and** (și). De asemenea există posibilitatea selectării conectorului logic de negare **not** atât pentru variabilele de intrare cât și pentru variabilele de ieșire.

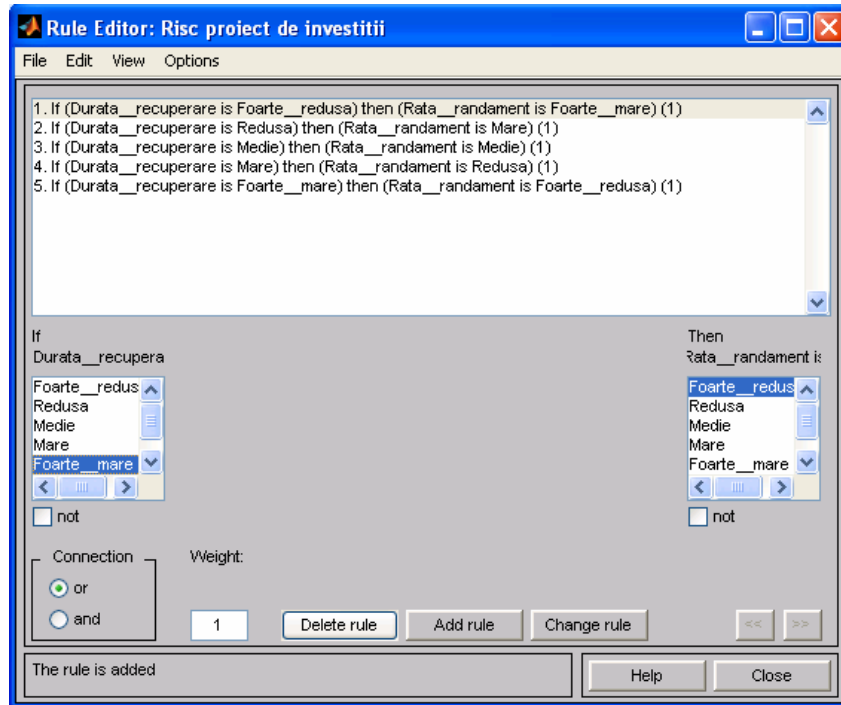


Fig. 1.46. Editarea bazei de cunoștințe

Procedura de editare a bazei de cunoștințe:

- Din lista derulantă corespunzătoare variabilei de **intrare** se selectează denumirea funcției de apartenență corespunzătoare regulii, care se editează.
- Din lista derulantă corespunzătoare variabilei de **ieșire** se selectează denumirea funcției de apartenență corespunzătoare regulii, care se editează.
- Se selectează butonul **Add rule** și regula se editează automat în fereastra superioară a editorului de reguli
- Procesul se repetă până la finalul editării bazei de cunoștințe. (Fig.1.46)

Observație: În cazul în care o regulă are o pondere mai mare decât celelalte reguli în cadrul procesului de inferență, se va specifica cifra ponderii în caseta **Weight**. Ponderea este afișată între paranteze în partea dreaptă a fiecărei reguli.

Procedura de ștergere sau modificare a bazei de cunoștințe

- Se selectează regula din baza de cunoștințe, care trebuie ștearsă sau modificată
- Se selectează butonul **Delete rule** (pentru ștergere), respective **Change rule** (pentru modificare). În cazul modificării, regula se va rescrie conform procedurii de editare.

În urma editării, baza de cunoștințe este conectată automat sistemului, putându-se vizualiza grafic inferența bazată pe logica fuzzy.

Procedura de vizualizare a inferenței fuzzy:

Opțiunea 1.

- Din meniul **View** al oricărei interfețe de editare se selectează **Rules** sau **Ctrl+5**, (Fig.1.47)
- Sistemul vizualizează regulile în sistemul reprezentat prin funcțiile de apartenență (Fig.1.48)

Opțiunea 2.

- Din meniul **View** al oricărei interfețe de editare se selectează **Surface** sau **Ctrl+6**, (Fig.1.47)

- Sistemul vizualizează în sistemul tridimensional suprafețele generate de inferența fuzzy în cazul în care există mai multe variabile de intrare și/sau de ieșire.

Observație: În cazul exemplului prezentat, există o singură variabilă de intrare și o singură variabilă de ieșire, vizualizarea fiind reprezentată în sistemul de axe bidimensional (Fig.1.49).

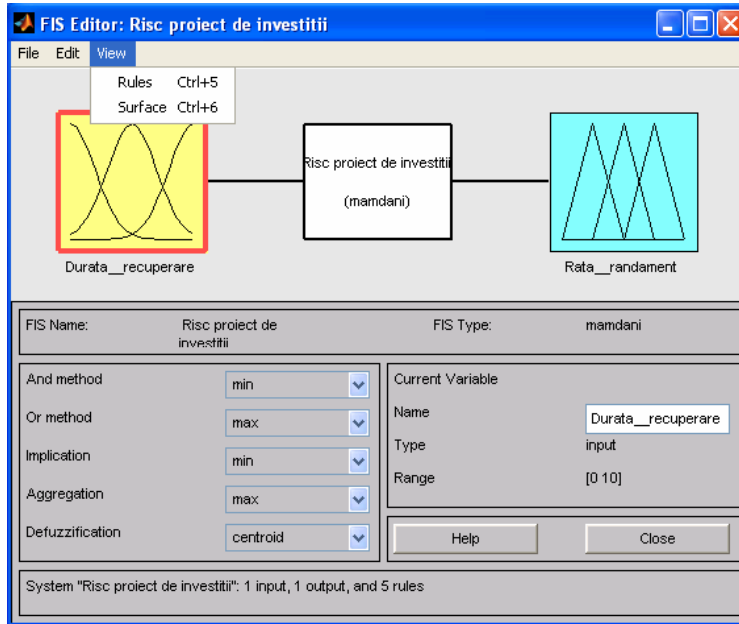


Fig. 1.47. Selectarea vizualizării regulilor

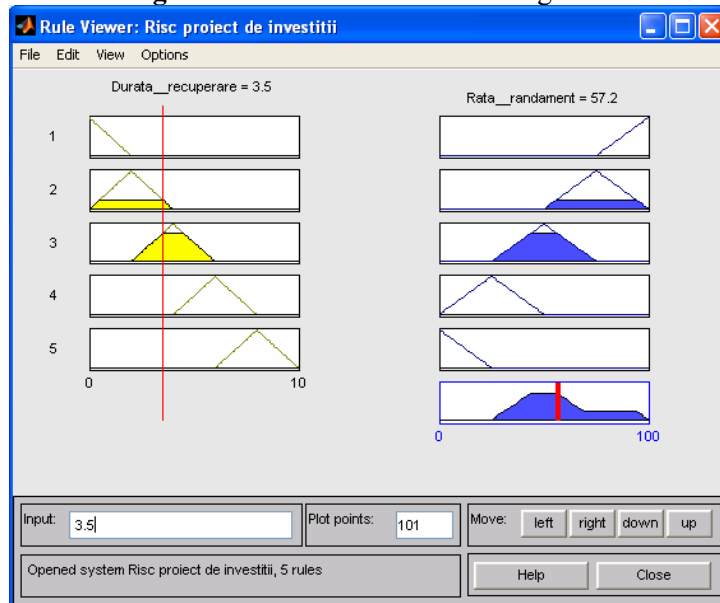


Fig. 1.48. Vizualizarea regulilor

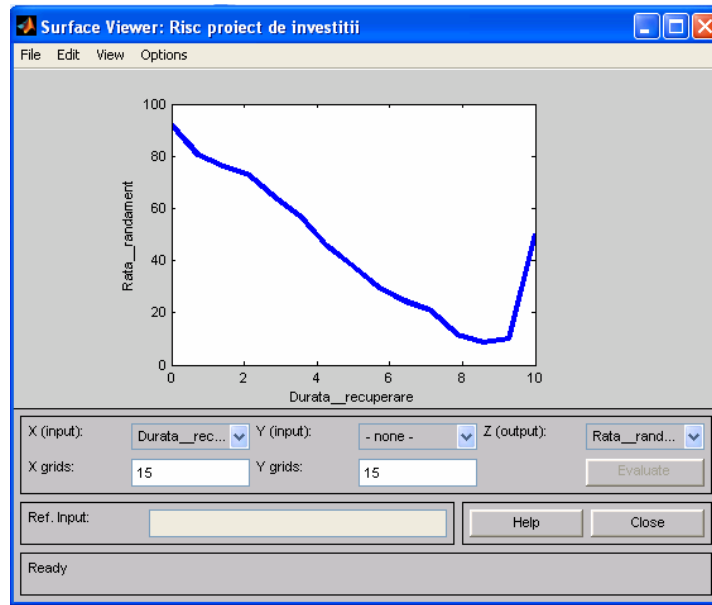


Fig. 1.49. Vizualizarea suprafețelor