**Introducere**

**Logica *fuzzy***  a fost definită în anul 1965 de către prof. [Lotfi Zadeh](http://ro.wikipedia.org/w/index.php?title=Lotfi_Zadeh&action=edit&redlink=1) de la Universitatea Californiei din Berkeley. Spre deosebire de logica clasică, în care se lucrează cu două valori de adevăr exacte (notate de ex. 0 pentru fals și 1 pentru adevărat), logica *fuzzy* folosește o plajă continuă de valori logice cuprinse în intervalul [0, 1], unde 0 indică falsitatea completă, iar 1 indică adevărul complet. Astfel, dacă în logica clasică un obiect poate aparține (1) sau nu (0) unei mulțimi date, logica *fuzzy* lucrează cu gradul de apartenență al obiectului la [mulțime](http://ro.wikipedia.org/wiki/Mul%C8%9Bime), acesta putând lua valori între 0 și 1.

Logica *fuzzy* oferă instrumentele necesare pentru reprezentarea în sistemele inteligente a unor concepte imprecise cum sunt „mare”, „fierbinte”, „ieftin” ș.a., concepte numite variabile lingvistice sau variabile *fuzzy*. Pentru reprezentarea acestora se folosesc seturile *fuzzy*, care captează din punct de vedere cantitativ interpretarea calitativă a termenilor.

Bazate pe logica *fuzzy*, sistemele *fuzzy* sunt considerate un caz particular al [sistemelor expert](http://ro.wikipedia.org/wiki/Sistem_expert) (motiv pentru care mai sunt denumite și sisteme expert *fuzzy*), sisteme care oferă o metodă flexibilă pentru tratarea incertitudinii.

Japonia este țara cu cele mai multe sisteme *fuzzy* implementate, mai ales în domeniile urmăririi producției și a vânzărilor. De asemenea, multe sisteme *fuzzy* au fost înglobate în unele dintre bunurile de larg consum: mașini de spălat rufe, cuptoare cu microunde, aparate foto, aparate de aer condiționat, etc.

Calculul fuzzy se caracterizează prin faptul că permite manipularea conceptelor vagi care nu pot fi modelate prin concepte matematice exacte (numere, mulţimi sau funcţii clasice). Probleme în care intervin concepte vagi apar în teoria controlului când sistemele au caracter neliniar iar stările lor nu pot fi descrise în mod exact, ci doar prin enunţuri care au un grad de ambiguitate.

Trasaturi specifice ale logicii *Fuzzy*:

* In logica fuzzy rationamentul exact este privit ca un caz limita a rationamentului aproximativ;
* In logica fuzzy orice situatie este exprimata gradual;
* Orice sistem logic poate fi reprezentat in logica fuzzy (fuzzyficare);
* In logica fuzzy cunoasterea este interpretata ca o colectie de restrictii fuzzy elastice sau de echivalenta pe o colectie de variabile;
* Deductia este privita ca un process de propagare al restrictiilor elastice;
* Multimea fuzzy are asociata o functie caracteristica care ia valori in intervalul [0,1], valorile acesteia descriind gradul de apartenenta al unui element la acea multime.

**Referinte stiintifice**

Extras din lucrarea: **“Automatic Transmission Shift Schedule Control Using Fuzzy Logic”-** Yamaguchi, H., Narita, Y., Takahashi, H., and Katou, Y- SAE Technical Paper 930674, 1993, doi:10.4271/930674.

Folosirea transmisiei automate la masini este in continua crestere datorita usurintei de operare. Cerintele cele mai importante ale transmisiei automate sunt: schimabare lina a vitezei si un program de schimbare a vitezelor care se potriveste cu intentiile conducatorului auto. In aceasta lucrare se vorbeste despre o cutie de viteze automata care seteaza programul de schimbare a vitezelor in functie de viteza vehiculului si deschiderea clapetei de acceleratie, oferind o imbunatatire in ceea ce priveste calitatea schimbarii vitezelor fata de sistemul cu cutie manuala de viteze.

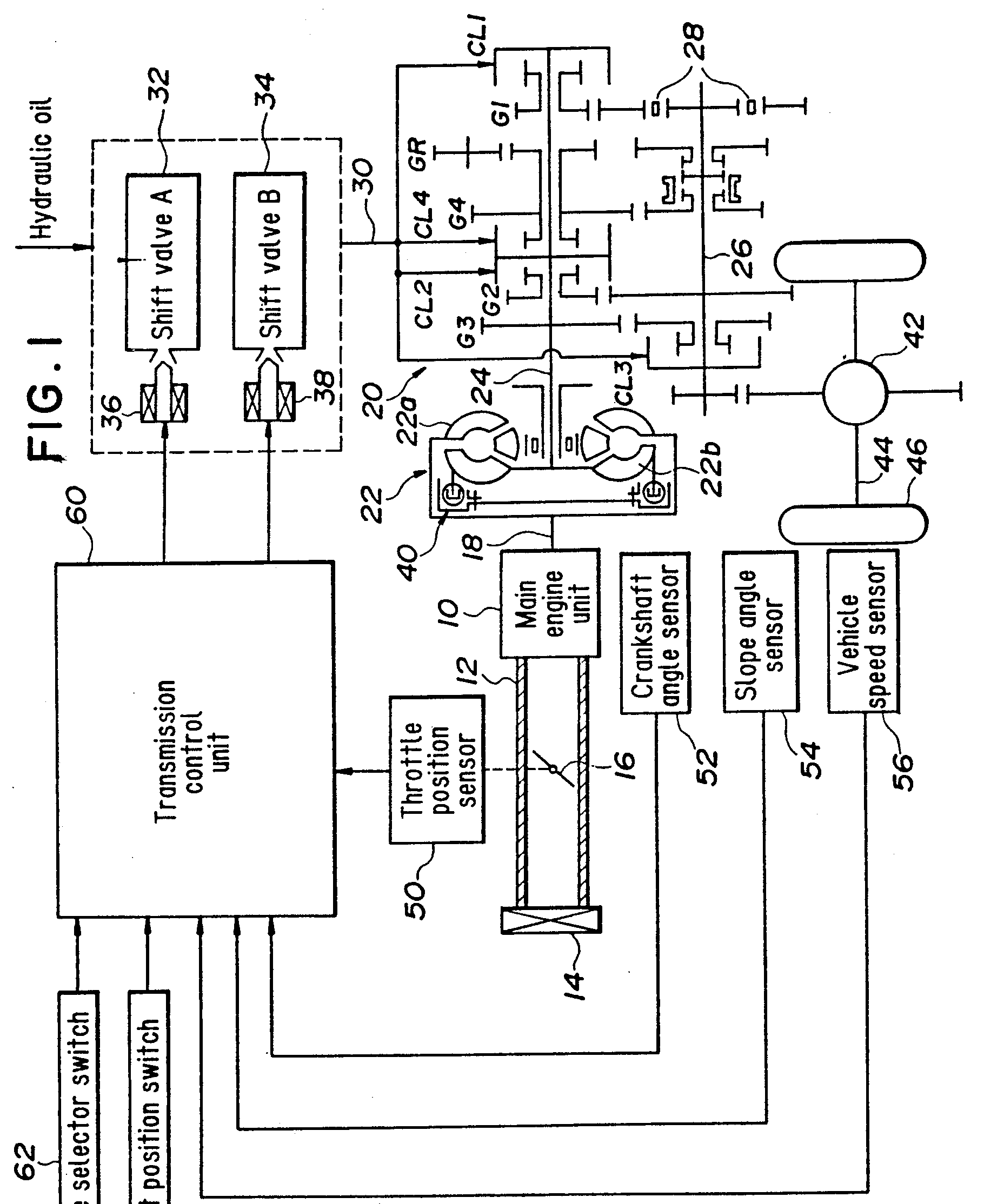
Cu toate acestea, chiar si cu o transmisie automata controlata electronic, pot sa apara erori in schimbarea vitezelor atunci cand conducem in rampa sau cand avem o remorca. Cu ajutorul logicii *fuzzy* s-a dezvoltat o tehnica care determina rezistenta de inaintare reprezentata de unghiul de inclinare a drumului. Aceasta tehnica a fost incorporata intr-o noua metoda de control care elimina erorile de schimbare ale vitezelor ce apareau in aceste conditii.

Extras din lucrarea: **“Vehicle automatic transmission control system”-** [Ichiro Sakai](http://www.google.com/search?tbo=p&tbm=pts&hl=en&q=ininventor:%22Ichiro+Sakai%22), [Yasuhisa Arai](http://www.google.com/search?tbo=p&tbm=pts&hl=en&q=ininventor:%22Yasuhisa+Arai%22), [Hiroki Matsui](http://www.google.com/search?tbo=p&tbm=pts&hl=en&q=ininventor:%22Hiroki+Matsui%22),[Masataka Yamamoto](http://www.google.com/search?tbo=p&tbm=pts&hl=en&q=ininventor:%22Masataka+Yamamoto%22)- [Honda Giken Kogyo Kabushiki Kaisha](http://www.google.com/search?tbo=p&tbm=pts&hl=en&q=inassignee:%22Honda+Giken+Kogyo+Kabushiki+Kaisha%22)

In aceasta lucrare avem un sistem de control automat, in mai multi pasi sau continuu, a transmisiei unui vehicul folosind logica *fuzzy*. In acest sistem au fost definite reguli *fuzzy* bazate pe deciziile unui conducator auto expert si pe operatiile efectuate pe un vehicul cu transmisie manuala. Diferite conditii de functionare, precum gradul de deschidere a pedalei de acceleratie si viteza vehiculului sunt detectate si in urma acestor detectii, diferite predictii sunt realizate cum ar fi raspunsul adecvat al vehiculului la intentionarea schimbarii treptei de viteza de catre conducatorul auto si schimbarea iesirii motorului, vor schimba treapta de viteza intr-o anumita viteza.

Toate aceste valori de masura sunt puse pe o anumita scara numita univers de discurs iar valorile de apartenenta ale regulilor sunt calculate succesiv. Dintre toate regulile, o regula este in cele din urma selectata si in functie de treapta de viteza indicata de regula respectiva, treapta de viteza actuala este schimbata in sus sau in jos, sau este mentinuta aceeasi viteza, in cazul unei transmisii efetuate in mai multi pasi. In mod similar treapta de viteza in cazul transmisiei cu variatie continua este determinata de regula selectata.

In figura urmatoare este prezentata o diagrama schematica aratand dispunerea generala a sistemului de control al transmisiei automate:



*Figura nr 1.* **Dispunerea generala al sistemului de control al transmisiei automate**

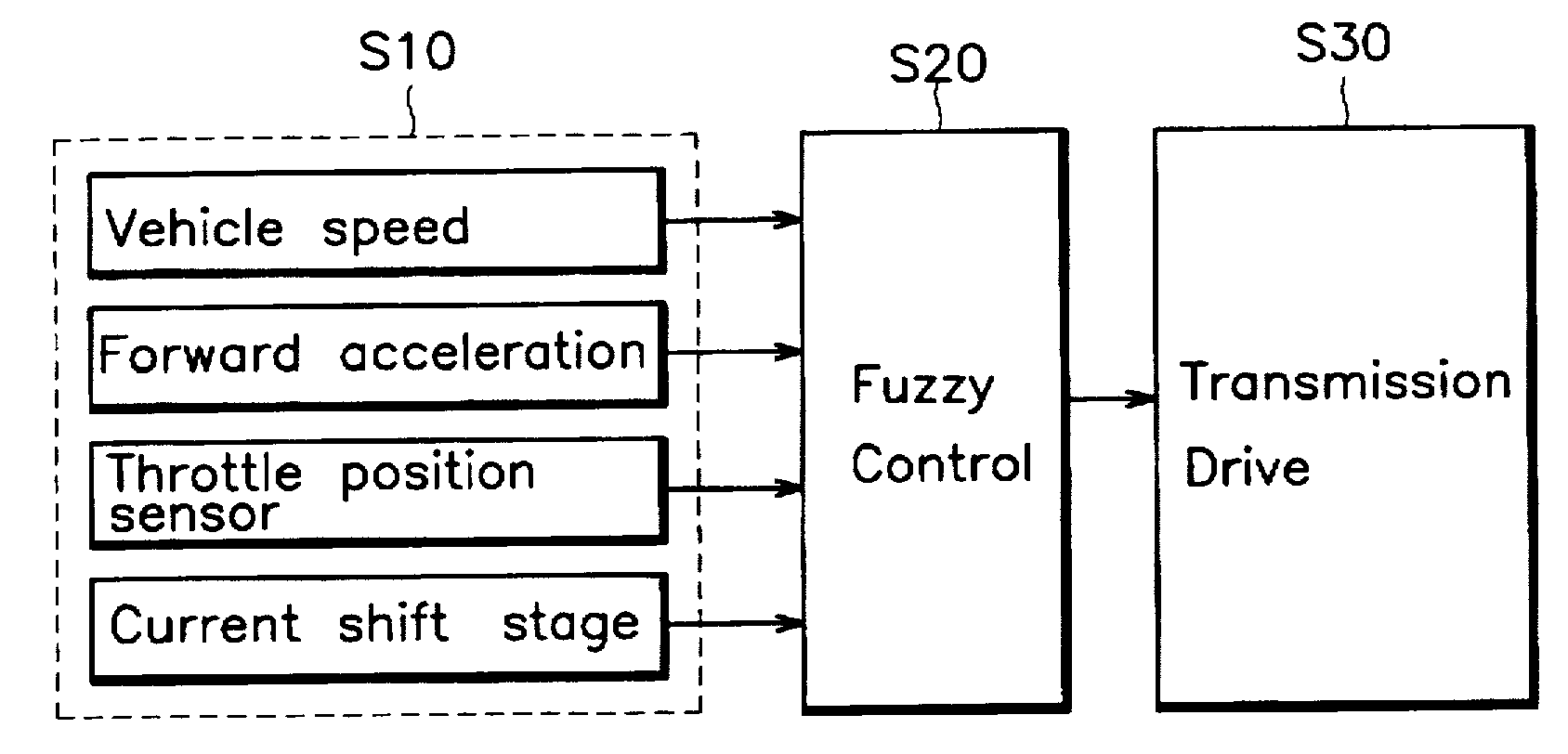
Extras din lucrarea**:” System for determining the shift stage of an automatic transmission by using fuzzy inference to decide road gradient”-** [Soo-Yong Jung](http://www.google.com/search?tbo=p&tbm=pts&hl=en&q=ininventor:%22Soo-Yong+Jung%22), [Hun Kang](http://www.google.com/search?tbo=p&tbm=pts&hl=en&q=ininventor:%22Hun+Kang%22)- [Hyundai Motor Company](http://www.google.com/search?tbo=p&tbm=pts&hl=en&q=inassignee:%22Hyundai+Motor+Company%22)

Un obiectiv al acestei lucrari este furnizarea unui system bazat pe logica *fuzzy* pentru determinarea treptei de viteza optime folosind logica *fuzzy* bazata pe viteza vehiculului, acceleratia, pozitia pedalei de acceleratie si pozitia curenta a treptei de viteza.

Conform acestei inventii, este furnizat un procedeu de determinare a treptei optime de viteza a unei transmisii automate, procedeu care este format din urmatorii pasi:

* obtinerea algoritmilor pentru invatarea gradului de inclinare a drumului;
* obtinerea algoritmului pentru deciderea gradului de inclinare a drumului utilizand logica *fuzzy*;
* determinarea treptei de viteza optime cu ajutorul unui procesor care foloseste logica *fuzzy*.

Sistemul inventat cuprinde o intrare de tip *fuzzy* care colecteaza date privitoare la viteza vehiculului, acceleratia, pozitia pedalei de acceleratie si treapta de viteza actuala, un control de tip *fuzzy* pentru controlarea datelor de intrare preluate de la intrarea de tip *fuzzy* si unitatea de transmisie pentru trecerea de la treapta de viteza actuala la o treapta de viteza optima. Ilustrarea sistemului este prezentata in urmatoarea diagrama bloc:



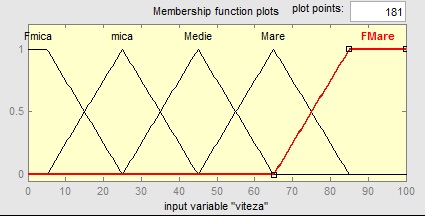
*Figura nr 2.* **Diagrama bloc**

**Descrierea proiectului**

In cadrul acestui proiect vom realizara controlul unei cutii de viteze cu transmisie automata folosind sisteme cu logica de control Fuzzy. Sistemul Fuzzy are 2 intrari si o iesire:

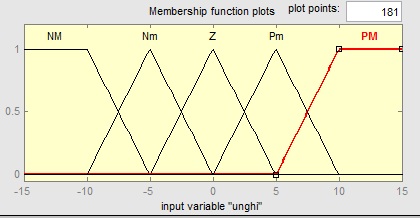
* intrarea 1- viteza vehiculului (km/h)
* intrarea 2- unghiul de inclinare al drumului (grade)
* iesirea- treapta de viteza care va fi selectata (1, 2, 3, 4 sau 5).

Fiecare intrate contine 5 functii de apartenenta. Intrarea 1 are urmatoarele functii de apartenenta: *Fmica* (foarte mica), *mica*, *medie*, *Mare*, *FMare* (foarte mare). Functia de intrare corespunzatoare vitezei vehiculului arata in felul urmator:



*Figura nr 3.* **Viteza vehiculului**

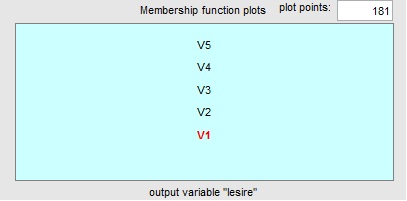
Intrarea 2 are urmatoarele 5 functii de apartenenta: *NM* (negative mare), *Nm* (negativ mic), *Z* (zero), *Pm* (pozitiv mic), *PM* (pozitiv mare). Functia de intrare corespunzatoare vitezei vehiculului arata in felul urmator:



*Figura nr 4.* **Unghiul de inclinare al drumului**

Aceasta intrare reprezinta unghiul drumului. Valorile cu + sunt associate unghiului de inclinare a unei rampe iar valorile cu – unghilui de inclinare a unei pante.

Iesirea contine tot 5 singletoni, acestea fiind urmatoarele: **V1** (treapta de viteza nr 1), **V2** (treapta de viteza nr 2), **V3** (treapta de viteza nr 3), **V4** (treapta de viteza nr 4) si **V5** (treapta de viteza nr 5). Functia de iesire arata in felul urmator:



*Figura nr 5.* **Iesirea sistemului Fuzzy (de tip Sugeno)**

In functie de valorile variabilelor de intrare se realizeaza un tabel de reguli care va determina treapta de viteza optima.

Tabelul de reguli este urmatoarul:

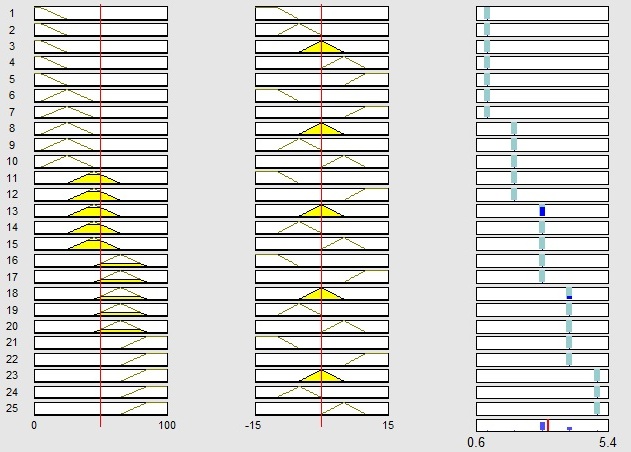
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Viteza \ Unghi | NM | Nm | Z | Pm | PM |
| Fmica | **V1** | **V1** | **V1** | **V1** | **V1** |
| Mica | **V1** | **V2** | **V2** | **V2** | **V1** |
| Medie | **V2** | **V3** | **V3** | **V3** | **V2** |
| Mare | **V3** | **V4** | **V4** | **V4** | **V3** |
| FMare | **V4** | **V5** | **V5** | **V5** | **V4** |

In cazul in care avem o inclinare foarte mare a unei rampei se selecteaza o treapta inferioara de viteza pentru a compensa forta de rezistenta opusa de inaintarea in rampa, iar cand avem o inclinare foarte mare a unei pante se selecteaza o treapta inferioara de viteza pentru a evita accelerarea brusca a vehiculului prin intrarea in frana de motor.

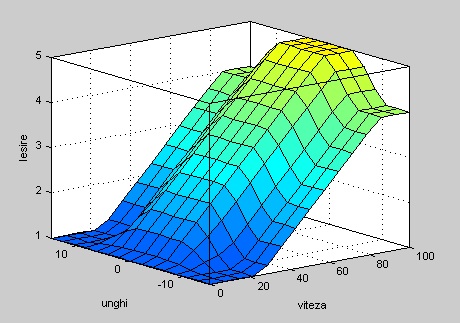
Setul de reguli este urmatorul:

1. IF *viteza* este foarte mica (Fmica) AND *unghiul* este negativ mare (NM) THEN *treapta de viteza* **V1**;
2. IF *viteza* este foarte mica (Fmica) AND *unghiul* este negativ mic (Nm) THEN *treapta de viteza* **V1**;
3. IF *viteza* este foarte mica (Fmica) AND *unghiul* este zero (Z) THEN *treapta de viteza* **V1**;
4. IF *viteza* este foarte mica (Fmica) AND *unghiul* este pozitiv mic (Pm) THEN *treapta de viteza* **V1**;
5. IF *viteza* este foarte mica (Fmica) AND *unghiul* este pozitiv mare (PM) THEN *treapta de viteza* **V1**;
6. IF *viteza* este mica (mica) AND *unghiul* este negativ mare (NM) THEN *treapta de viteza* **V1**;
7. IF *viteza* este mica (mica) AND *unghiul* este negativ mic (Nm) THEN *treapta de viteza* **V2**;
8. IF *viteza* este mica (mica) AND *unghiul* este zero (Z) THEN *treapta de viteza* **V2**;
9. IF *viteza* este mica (mica) AND *unghiul* este pozitiv mic (Pm) THEN *treapta de viteza* **V2**;
10. IF *viteza* este mica (mica) AND *unghiul* este pozitiv mare (PM) THEN *treapta de viteza* **V1**;
11. IF *viteza* este medie (medie) AND *unghiul* este negativ mare (NM) THEN *treapta de viteza* **V2**;
12. IF *viteza* este medie (medie) AND *unghiul* este negativ mic (Nm) THEN *treapta de viteza* **V3**;
13. IF *viteza* este medie (medie) AND *unghiul* este zero (Z) THEN *treapta de viteza* **V3**;
14. IF *viteza* este medie (medie) AND *unghiul* este pozitiv mic (Pm) THEN *treapta de viteza* **V3**;
15. IF *viteza* este medie (medie) AND *unghiul* este pozitiv mare (PM) THEN *treapta de viteza* **V2**;
16. IF *viteza* este Mare (Mare) AND *unghiul* este negativ mare (NM) THEN *treapta de viteza* **V3**;
17. IF *viteza* este Mare (Mare) AND *unghiul* este negativ mic (Nm) THEN *treapta de viteza* **V4**;
18. IF *viteza* este Mare (Mare) AND *unghiul* este zero (Z) THEN *treapta de viteza* **V4**;
19. IF *viteza* este Mare (Mare) AND *unghiul* este pozitiv mic (Pm) THEN *treapta de viteza* **V4**;
20. IF *viteza* este Mare (Mare) AND *unghiul* este pozitiv mare (PM) THEN *treapta de viteza* **V3**;
21. IF *viteza* este FMare (Foarte Mare) AND *unghiul* este negativ mare (NM) THEN *treapta de viteza* **V4**;
22. IF *viteza* este FMare (Foarte Mare) AND *unghiul* este negativ mic (Nm) THEN *treapta de viteza* **V5**;
23. IF *viteza* este FMare (Foarte Mare) AND *unghiul* este zero (Z) THEN *treapta de viteza* **V5**;
24. IF *viteza* este FMare (Foarte Mare) AND *unghiul* este pozitiv mic (Pm) THEN *treapta de viteza* **V5**;
25. IF *viteza* este FMare (Foarte Mare) AND *unghiul* este pozitiv mare (PM) THEN *treapta de viteza* **V4**.

In figura de mai jos este prezentat graficul rezultat si suprafata rezultata in urma acestor reguli:



*Figura nr 6.* **Reguli Fuzzy Matlab (Sugeno)**



*Figura nr 7.* **Suprafata rezultata**

**Rezultatele simularilor**

In tabelul urmator avem date niste exemple de valori de intrare si rezultatele simularii in Matlab, respective in cod C (codul in C este prezentat in Anexa 1).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr.crt | Viteza(km/h) | Unghi inclinare(grade) | Valoare iesire (Matlab) | Valoare iesire (C) |
| 1 | 20 | -4 | 1.75 | 1.75 |
| 2 | 43 | 0 | 2.9 | 2.9 |
| 3 | 66 | +10 | 3.05 | 3.05 |
| 4 | 66 | 0 | 4.05 | 4.05 |
| 5 | 11 | +4 | 1.3 | 1.3 |
| 6 | 80 | -3 | 4.75 | 4.7 |
| 7 | 33 | -10 | 1.4 | 1.4 |
| 8 | 72 | +5 | 4.35 | 4.35 |
| 9 | 90 | +9 | 4.2 | 4.2 |
| 10 | 14 | +10 | 1 | 1 |
| 11 | 100 | +2 | 5 | 5 |

**Descriere algoritm utilizat**

In figura de mai jos este prezentata structura unui sistem fuzzy.

 *Figura nr 8.* **Sistem logica Fuzzy**

Pasii algoritmului sunt:

1. Definirea termenilor si variabilelor lingvistice (initializare);
2. Construirea functiilor de apartenenta (initializare);
3. Construirea setului de reguli (initializare);
4. Conversia datelor de intrare de tip crisp in valori fuzzy folosind functiile de apartenenta (fuzzificare);
5. Evaluarea regulilor din setul de reguli (interferenta);
6. Combinarea rezultatelor fiecarei reguli (interferenta);
7. Conversia datelor de iesire in valori non-fuzzy (defuzzificare).

**Studii de caz**

**Studiu de caz nr.1**

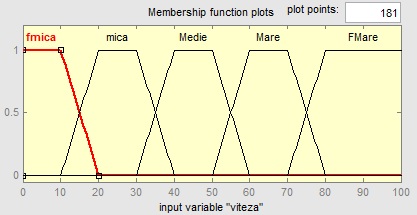
Primul studiu de caz consta in schimbarea tipului de sistem Fuzzy, din Sugeno in Mamdani. Vom avea iesiri triunghiulare in loc de singletoni. In tabelul de mai jos avem valorile obtinute in urma simularii sistemelor fuzzy de tip Sugeno si Mamdami:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr.crt | Viteza(km/h) | Unghi inclinare(grade) | Valoare iesire Sugeno | Valoare iesire Mamdami |
| 1 | 20 | -4 | 1.75 | 1.71 |
| 2 | 43 | 0 | 2.9 | 2.87 |
| 3 | 66 | +10 | 3.05 | 3.07 |
| 4 | 66 | 0 | 4.05 | 4.07 |
| 5 | 11 | +4 | 1.3 | 1.33 |
| 6 | 80 | -3 | 4.75 | 4.69 |
| 7 | 33 | -10 | 1.4 | 1.42 |
| 8 | 72 | +5 | 4.35 | 4.38 |
| 9 | 90 | +9 | 4.2 | 4.24 |
| 10 | 14 | +10 | 1 | 1 |
| 11 | 100 | +2 | 5 | 5 |

In urma rezultatelor obtinute in urma celor doua simulari se observa ca valorile nu variaza foarte mult, treapta de viteza care urmeaza a fi selectata fiind aceeasi in ambele sisteme fuzzy.

**Studiu de caz nr.2**

Al doilea studiu de caz consta in schimbarea functiilor de apartenenta ale intrarii care reprezinta viteza vehiculului din triunghi in trapez . In acest mod creste probabilitatea ca la un moment dat sa avem doar doua reguli active.



*Figura nr 9.* **Viteza vehiculului (trapez)**

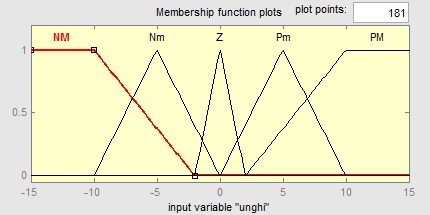
Rezultatele obtinute sunt prezentate in tabelul de mai jos:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr.crt | Viteza(km/h) | Unghi inclinare(grade) | Valoare iesire Sugeno(triunghi) | Valoare iesire Sugeno(trapez) |
| 1 | 20 | -4 | 1.75 | 2 |
| 2 | 43 | 0 | 2.9 | 3 |
| 3 | 66 | +10 | 3.05 | 3 |
| 4 | 66 | 0 | 4.05 | 4 |
| 5 | 11 | +4 | 1.3 | 1.1 |
| 6 | 80 | -3 | 4.75 | 5 |
| 7 | 33 | -10 | 1.4 | 1.3 |
| 8 | 72 | +5 | 4.35 | 4.2 |
| 9 | 90 | +9 | 4.2 | 4.2 |
| 10 | 14 | +10 | 1 | 1 |
| 11 | 100 | +2 | 5 | 5 |

Se observa ca valorile variaza de aceasta data insa treapta de viteza care urmeaza a fi selectata ramane aceeasi in ambele situatii.

**Studiu de caz nr.3**

Al treilea studiu de caz consta in modificarea functiilor de apartenenta ale intrarii care reprezinta unghiul de inclinare al drumului. Modificarea este urmatoarea:



*Figura nr 10.* **Unghiul de inclinare al drumului (modificat)**

Rezultatele obtinute sunt prezentate in tabelul de mai jos:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr.crt | Viteza(km/h) | Unghi inclinare(grade) | Valoare iesire Sugeno | Valoare iesire Sugeno(modificat) |
| 1 | 20 | -4 | 1.75 | 1.57 |
| 2 | 43 | 0 | 2.9 | 2.9 |
| 3 | 66 | +10 | 3.05 | 3.05 |
| 4 | 66 | 0 | 4.05 | 4.05 |
| 5 | 11 | +4 | 1.3 | 1.23 |
| 6 | 80 | -3 | 4.75 | 4.58 |
| 7 | 33 | -10 | 1.4 | 1.4 |
| 8 | 72 | +5 | 4.35 | 4.08 |
| 9 | 90 | +9 | 4.2 | 4.2 |
| 10 | 14 | +10 | 1 | 1 |
| 11 | 100 | +2 | 5 | 5 |

Valorile care variaza de aceasta data sunt valorile care au ca variabila de intrare un unghi cuprins intre [-5 5] , pentru celelalte valori ale unghiului de inclinare rezultatele sunt identice.

**Studiu de caz nr.4**

In acest studiu de caz vom modifica tabela de reguli dupa cum urmeaza:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Viteza \ Unghi | NM | Nm | Z | Pm | PM |
| Fmica | **V1** | **V1** | **V1** | **V1** | **V1** |
| Mica | **V1** | **V1** | **V2** | **V1** | **V1** |
| Medie | **V2** | **V2** | **V3** | **V2** | **V1** |
| Mare | **V3** | **V3** | **V4** | **V3** | **V2** |
| FMare | **V4** | **V4** | **V5** | **V4** | **V3** |

Rezultatele obtinute in urma modificarii tabelei de reguli sunt prezentate in tabelul de mai jos:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr.crt | Viteza(km/h) | Unghi inclinare(grade) | Valoare iesire Sugeno | Valoare iesire Sugeno(modificat) |
| 1 | 20 | -4 | 1.75 | 1.15 |
| 2 | 43 | 0 | 2.9 | 2.9 |
| 3 | 66 | +10 | 3.05 | 2.05 |
| 4 | 66 | 0 | 4.05 | 4.05 |
| 5 | 11 | +4 | 1.3 | 1.06 |
| 6 | 80 | -3 | 4.75 | 4.15 |
| 7 | 33 | -10 | 1.4 | 1.4 |
| 8 | 72 | +5 | 4.35 | 3.35 |
| 9 | 90 | +9 | 4.2 | 3.2 |
| 10 | 14 | +10 | 1 | 1 |
| 11 | 100 | 2 | 5 | 4.6 |

In acest caz valorile variaza mult fata de volorile initiale mai ales cand conducem in rampa.

**Concluzii**

In urma studiilor de caz realizate se observa ca nu exista o diferenta notabila intre folosirea unui sistem fuzzy de tip Sugeno sau Mamdami, rezultatele fiind foarte apropiate ca valori, treapta de viteza selectata fiind aceeasi in ambele cazuri. Valorile de iesire rezultate in urma schimbarii functiilor de apartenenta ale celor doua intrari difera, nu cu foarte mult de valorile initiale.

Sistemul realizat elimina erorile aparute la schimbarea vitezelor atunci cand conducem in rampa sau in panta selectand treapta de viteza optima pentru a compensa forta de rezistenta opusa de inaintarea in rampa si pentru a preveni accelerarea brusca in panta.

**Bibliografie**

- http://ro.wikipedia.org/wiki/Logică\_fuzzy

- <http://papers.sae.org/930674/>

- <http://www.google.com/patents/US5036730>

- <http://www.google.com/patents/US5794169>

**Anexe**

**Anexa 1**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#define min(x,y) ((x<y)?(x):(y))

#define max(x,y) ((x>y)?(x):(y))

#define N 24

#define nax1 5

#define nax2 5

#define nay 5

float x1[nax1][3]={{0,5,25} , {5,25,45}, {25,45,65},{45,65,85},{65,85,100}};

float miux1[nax1][3]={{1,1,0} , {0,1,0}, {0,1,0}, {0,1,0} , {0,1,1}};

int npx1=3;

float x2[nax2][3]={{-15,-10,-5} , {-10,-5,0} , {-5,0,5} , {0,5,10} , {5,10,15}};

float miux2[nax2][3]={{1,1,0} , {0,1,0} , {0,1,0} , {0,1,0} , {0,1,1}};

int npx2=3;

float b[]={1,2,3,4,5};

float y[nay][3]={{0,1,2} , {1,2,3} , {2,3,4} , {3,4,5} , {4,5,6}};

float miuy[nay][3]={{0,1,0} , {0,1,0} , {0,1,0} , {0,1,0} , {0,1,0}};

float yinf=0, ysup=6;

int npy=3;

int reg[nax1][nax2]={{0,0,0,0,0} , {0,1,1,1,0} , {1,2,2,2,1},{2,3,3,3,2},{3,4,4,4,3}};

float fa(float xc, float x[], float miux[], int npx)

{ if (xc<=x[0]) return miux[0];

for(int i=0;i<=npx-2;i++)

if(x[i]<=xc && xc<=x[i+1])

return ((miux[i]<miux[i+1])?(xc-x[i]):(x[i+1]-xc)) /(x[i+1]-x[i]) \* ((miux[i+1]-miux[i]>0)?(miux[i+1]-miux[i]):(miux[i]-miux[i+1])) + ((miux[i]<miux[i+1])?miux[i]:miux[i+1]);

return miux[npx-1];

}

void intersectie(float xA[], float miuA[], int npA, float xB[], float miuB[], int npB, float xI[], float miuI[])

{ float xinf, xsup, step;

xinf=min(xA[0], xB[0]);

xsup=max(xA[npA-1], xB[npB-1]);

step=(xsup-xinf)/N;

for(int i=0; i<=N; i++)

{ xI[i]=xinf+(step\*i);

miuI[i]=min(fa(xI[i],xA,miuA, npA),fa(xI[i],xB,miuB, npB));

}

}

void trunchiere(float y[], float miuy[], int npy, float miu, float yT[], float miuT[])

{ float step;

step=(ysup-yinf)/N;

for(int i=0; i<=N; i++)

{ yT[i]=yinf+(step\*i);

miuT[i]=min(fa(yT[i],y,miuy, npy),miu);

}

}

float sugeno(float x1c, float x2c)

{ float miux1c[nax1], miux2c[nax2],miub[nay],Suma1,Suma2,yc;

int i,j;

//fuzificare

for(i=0;i<nax1;i++)

miux1c[i]=fa(x1c, x1[i], miux1[i], npx1);

for(i=0;i<nax2;i++)

miux2c[i]=fa(x2c, x2[i], miux2[i], npx2);

for(i=0;i<nay;i++)

miub[i]=0;

//terminare fuzificare

//aplicare reguli fuzzy

for(i=0;i<nax1;i++)

for(j=0;j<nax2;j++)

if(miux1c[i]!=0 && miux2c[j]!=0)

{ if(min(miux1c[i],miux2c[j]) > miub[reg[i][j]])

miub[reg[i][j]]=min(miux1c[i],miux2c[j]);

}

//centru de greutate sugenno

Suma1=0; Suma2=0;

for(i=0;i<nay;i++)

{

Suma1+=b[i]\*miub[i];

Suma2+=miub[i];

}

yc=Suma1/Suma2;

return yc;

}

float mandami(float x1c,float x2c)

{ float miux1c[nax1], miux2c[nax2], yT[N+1], miuT[N+1],miuR[N+1], miu, Suma1,Suma2, yc;

int i,j,k;

//fuzificare

for(i=0;i<nax1;i++)

miux1c[i]=fa(x1c, x1[i], miux1[i], npx1);

for(i=0;i<nax2;i++)

miux2c[i]=fa(x2c, x2[i], miux2[i], npx2);

for(k=0; k<=N; k++)

miuR[k]=0;

//terminare fuzificare

//aplicare reguli fuzzy

for(i=0;i<nax1;i++)

for(j=0;j<nax2;j++)

if(miux1c[i]!=0 && miux2c[j]!=0)

{ miu = min(miux1c[i],miux2c[j]);

trunchiere(y[reg[i][j]], miuy[reg[i][j]], npy, miu, yT, miuT);

//reuniune

for (k=0; k<N; k++)

if (miuR[k]<miuT[k]) miuR[k]=miuT[k];

}

//centru de greutate

Suma1=0; Suma2=0;

for(k=0;k<=N;k++)

{

Suma1+=yT[k]\*miuR[k];

Suma2+=miuR[k];

}

yc=Suma1/Suma2;

return yc;

}

void main(void)

{

float x1c, x2c, yc;

printf("Introduceti viteza:\n");

scanf("%f",&x1c);

printf("Introduceti unghiul:\n");

scanf("%f",&x2c);

clrscr();

yc = sugeno(x1c,x2c);

printf("Sugeno x1c:%f, x2c:%f -> yc:%f\n",x1c, x2c, yc);

yc = mandami(x1c,x2c);

printf("Mamadani x1c:%f, x2c:%f -> yc:%f\n",x1c, x2c, yc);

getch();

}