**KAUNO TECHNOLOGIJOS UNIVERSITETAS**

**INFORMATIKOS FAKULTETAS**

INTELEKTIKOS PAGRINDAI (P176B101)

**4 laboratorinio darbo ataskaita.**

Atliko:

IFF-4/1 gr. studentas

Mangirdas Kazlauskas

Priėmė:

Doc. Agnė Paulauskaitė-Tarasevičienė

**KAUNAS 2017**

Turinys

[1. Darbo užduotis 3](#_Toc479757775)

[2. Konceptualus modelis 4](#_Toc479757776)

[3. Matlab modeliavimas 4](#_Toc479757777)

[3.1. Įėjimo kintamųjų funkcijos 4](#_Toc479757778)

[3.1.1. Atstumo įėjimo funkcija 5](#_Toc479757779)

[3.1.2. Apkrautumo įėjimo funkcija 6](#_Toc479757780)

[3.1.3. Laiko įėjimo funkcija 7](#_Toc479757781)

[3.2. Išėjimo kintamojo funkcija 9](#_Toc479757782)

[3.3. Taisyklių sudarymas 10](#_Toc479757783)

[3.4. Išvesties kintamųjų prognozavimas 11](#_Toc479757784)

[3.5. Priklausomybių grafiniai paviršiai 13](#_Toc479757785)

[4. Agregacijos metodų max ir probor palyginimas 14](#_Toc479757786)

[5. Defuzikacijos metodų centroid ir bisector palyginimas 16](#_Toc479757787)

# Darbo užduotis

Pasirinktas laboratorinio darbo užduoties variantas – 6. Užduotis skamba taip:

Sukurkite Fuzzy modelį naudojant Mamdani metodą, kuris apskaičiuotų apsaugos reagavimo trukmę į signalizacijos suveikimo signalą.

Įėjimo kintamieji:

• Atstumas iki objekto nuo artimiausios būstinės (pvz., [1-25 km]);

• Maršruto gatvių apkrautumas (pvz., [1 – 5], kur 1- mažas, 5 – gatvės su daug šviesoforų);

• Paros laikas (pvz., [1.00 -24.00]).

Išėjimo kintamieji:

• Trukmė (pvz., 1- 13 min.).

Ataskaitoje pateikti: įėjimo kintamųjų funkcijų parametrus ir priklausomybių grafikus, išėjimo kintamųjų funkcijų parametrus ir priklausomybių grafikus, taisykles, rezultatus su trimis testinėmis imtimis (Rule lange), priklausomybių grafinius paviršius.

Parinkite skirtingus 2 Agregacijos metodus (max, probor) ir palyginkite išėjimo atsakymus su ta pačia įėjimo duomenų imtimi. Su max Agregacijos ir prod Implikacijos metodais parinkite skirtingus Defuzikacijos metodus centroid ir bisector. palyginkite atsakymus su ta pačia įėjimo duomenų imtimi.

# Konceptualus modelis

Užduoties sąlygoje yra pateikta tam tikra informacija, reikalinga nustatyti apsaugos reagavimo trukmę į signalizacijos suveikimo signalą:

1. Atstumas iki objekto;
2. Maršruto iki objekto gatvių apkrautumas;
3. Paros laikas.

Šie kriterijai yra suskirstomi į tam tikrus miglotosios logikos lygmenis, kurie yra pateikti 1-3 lentelėse:

1 lentelė. Atstumo iki objekto Fuzzy lygmenys

|  |  |
| --- | --- |
| Atstumas, km | Atstumo Fuzzy lygmuo |
| <13 | artimas\_a |
| 7.5-17.5 | vidutinis\_a |
| >13 | tolimas\_a |

2 lentelė. Maršruto iki objekto gatvių apkrautumo Fuzzy lygmenys

|  |  |
| --- | --- |
| Apkrautumas | Apkrautumo Fuzzy lygmuo |
| <3 | mazas\_a |
| 1-5 | vidutinis\_a |
| >3 | didelis\_a |

3 lentelė. Paros laiko Fuzzy lygmenys

|  |  |
| --- | --- |
| Paros laikas, h | Laiko Fuzzy lygmuo |
| 01:00-06:00 | Naktis |
| 04:00-12:00 | Rytas |
| 10:00-18:00 | Popietė |
| 16:00-24:00 | Vakaras |

Pagal šias tris rodiklių grupes yra nusprendžiama, kokia bus apsaugos reagavimo trukmė į signalizacijos suveikimo signalą (4 lentelė):

4 lentelė. Reagavimo trukmės Fuzzy lygmenys

|  |  |
| --- | --- |
| Trukme, min | Reagavimo Fuzzy lygmuo |
| 1-3 | zaibiskas\_r |
| 2-7 | greitas\_r |
| 5-11 | letas\_r |
| 9-12 | labaiLetas\_r |

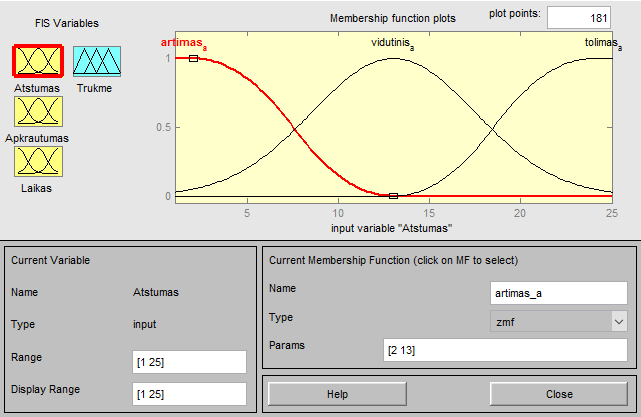
# Matlab modeliavimas

## Įėjimo kintamųjų funkcijos

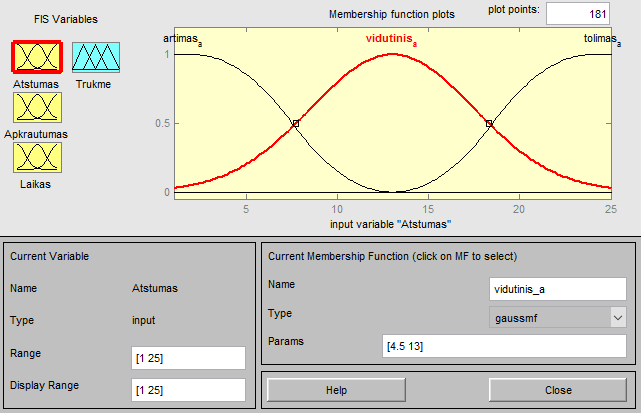
Remiantis sudarytais Fuzzy lygmenimis bei naudojantis Matlab *fuzzy* komandos grafine sąsaja sudaromos įėjimo kintamųjų funkcijos.

### Atstumo įėjimo funkcija

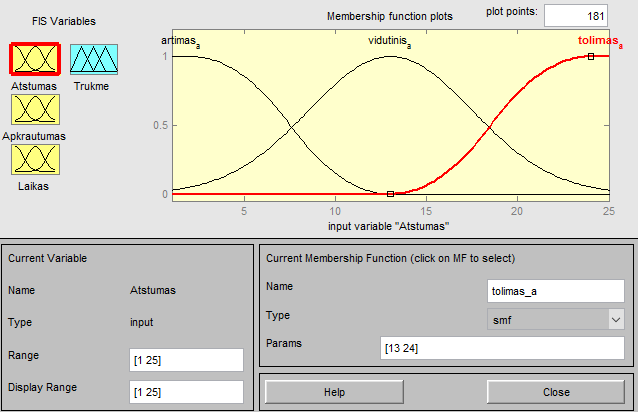
Atstumas iki objekto apibrėžiamas 3 skirtingais lygmenimis, todėl naudojamos trys priklausomumo funkcijos, aprašančios kintamąjį Atstumas:



1 pav. Atstumo Fuzzy lygmens artimas\_a priklausomumo funkcija ir jos parametrai



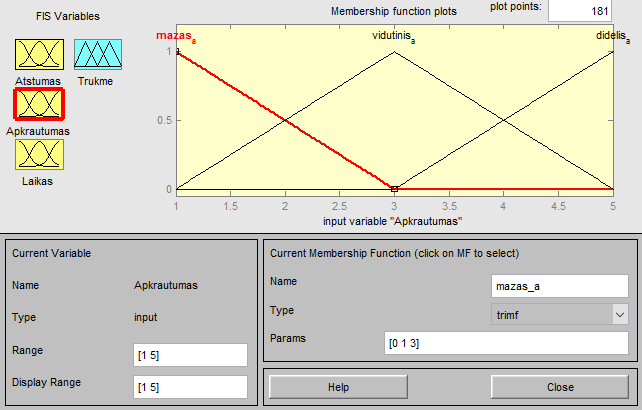
2 pav. Atstumo Fuzzy lygmens vidutinis\_a priklausomumo funkcija ir jos parametrai



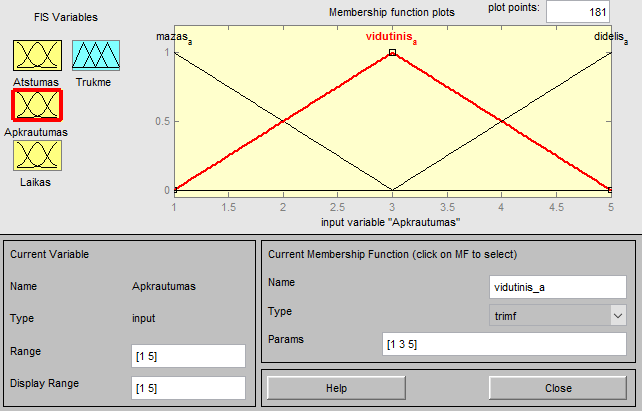
3 pav. Atstumo Fuzzy lygmens tolimas\_a priklausomumo funkcija ir jos parametrai

### Apkrautumo įėjimo funkcija

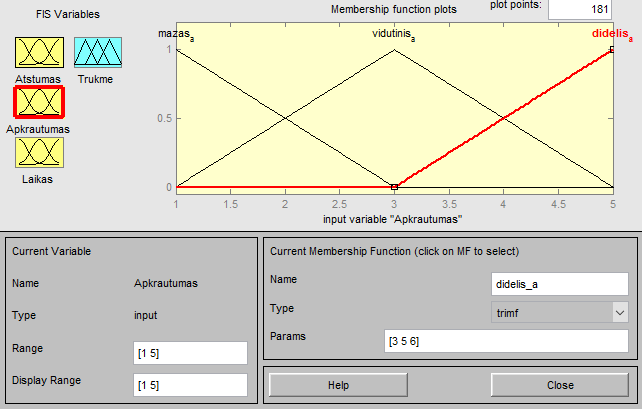
Maršruto apkrautumas iki objekto apibrėžiamas 3 skirtingais lygmenimis, todėl naudojamos trys priklausomumo funkcijos, aprašančios kintamąjį Apkrautumas:



4 pav. Apkrautumo Fuzzy lygmens mazas\_a priklausomumo funkcija ir jos parametrai



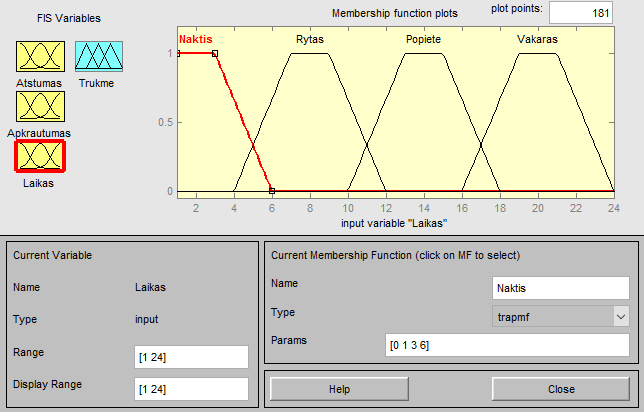
5 pav. Apkrautumo Fuzzy lygmens vidutinis\_a priklausomumo funkcija ir jos parametrai



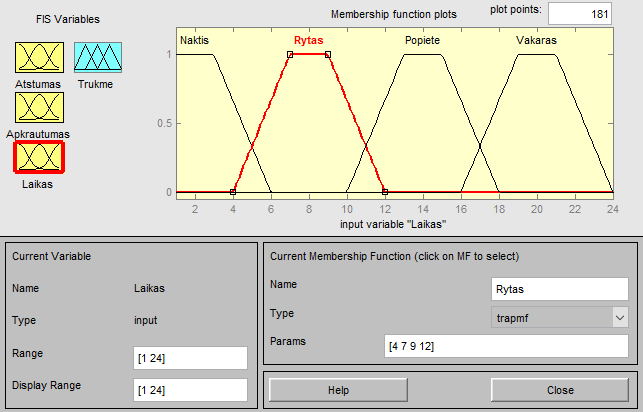
6 pav. Apkrautumo Fuzzy lygmens didelis\_a priklausomumo funkcija ir jos parametrai

### Laiko įėjimo funkcija

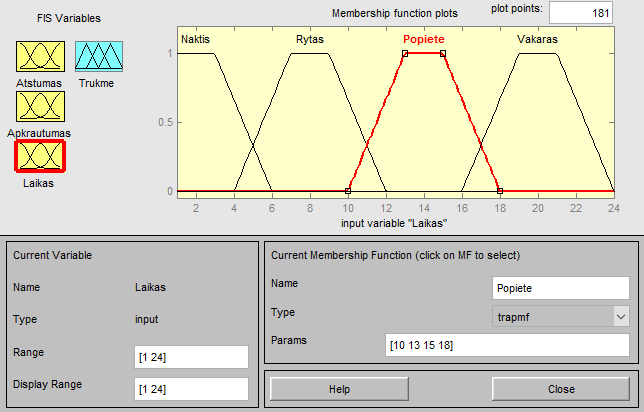
Paros laikas apibrėžiamas 4 skirtingais lygmenimis, todėl naudojamos keturios priklausomumo funkcijos, aprašančios kintamąjį Laikas:



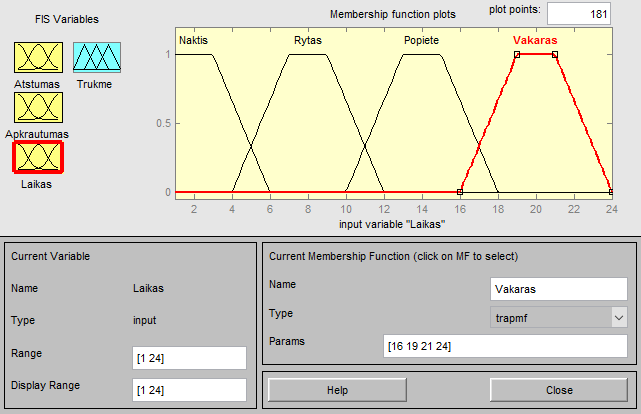
7 pav. Laiko Fuzzy lygmens Naktis priklausomumo funkcija ir jos parametrai



8 pav. Laiko Fuzzy lygmens Rytas priklausomumo funkcija ir jos parametrai



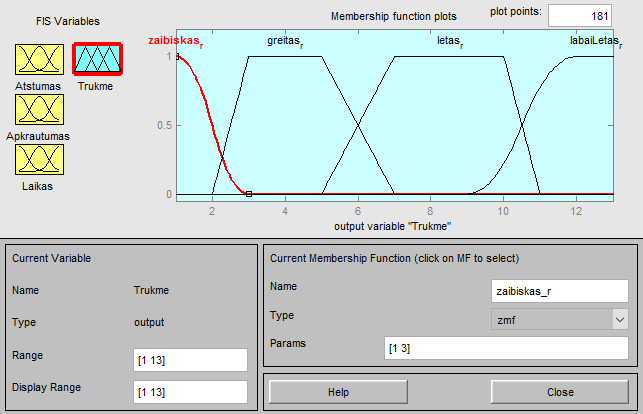
9 pav. Laiko Fuzzy lygmens Popiete priklausomumo funkcija ir jos parametrai



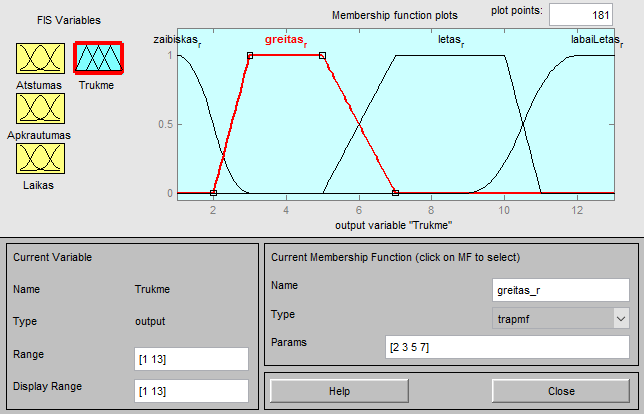
10 pav. Laiko Fuzzy lygmens Vakaras priklausomumo funkcija ir jos parametrai

## Išėjimo kintamojo funkcija

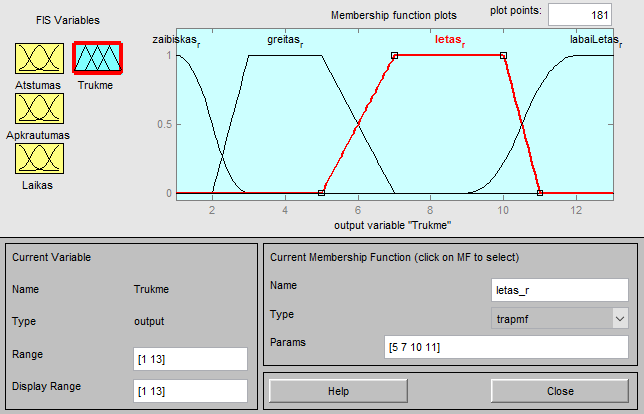
Išėjimo kintamasis – reagavimo laikas (trukmė) į signalizacijos signalą apibrėžiamas 4 skirtingais lygmenimis, todėl naudojamos keturios priklausomumo funkcijos, aprašančios kintamąjį Trukme:



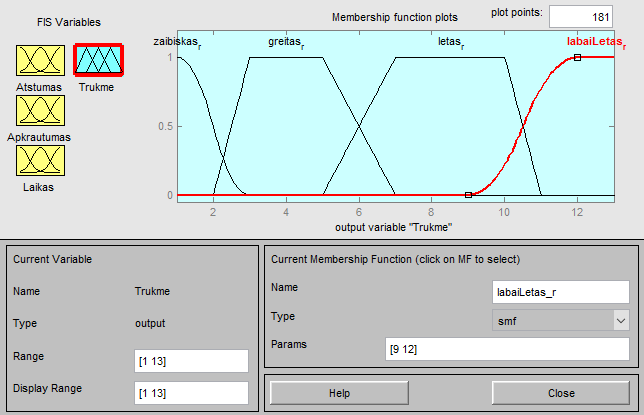
11 pav. Trukmės Fuzzy lygmens zaibiskas\_r priklausomumo funkcija ir jos parametrai



12 pav. Trukmės Fuzzy lygmens greitas\_r priklausomumo funkcija ir jos parametrai



13 pav. Trukmės Fuzzy lygmens letas\_r priklausomumo funkcija ir jos parametrai



14 pav. Trukmės Fuzzy lygmens labaiLetas\_r priklausomumo funkcija ir jos parametrai

## Taisyklių sudarymas

Sudarius įvesties ir išvesties kintamųjų funkcijas, sudaromos taisyklės, kurios susieja įėjimo kintamuosius su išėjimo kintamuoju. Tam, kad būtų apibrėžtos visos galimos įvestys, užteko sudaryti 10 taisyklių rinkinį (5 lentelė):

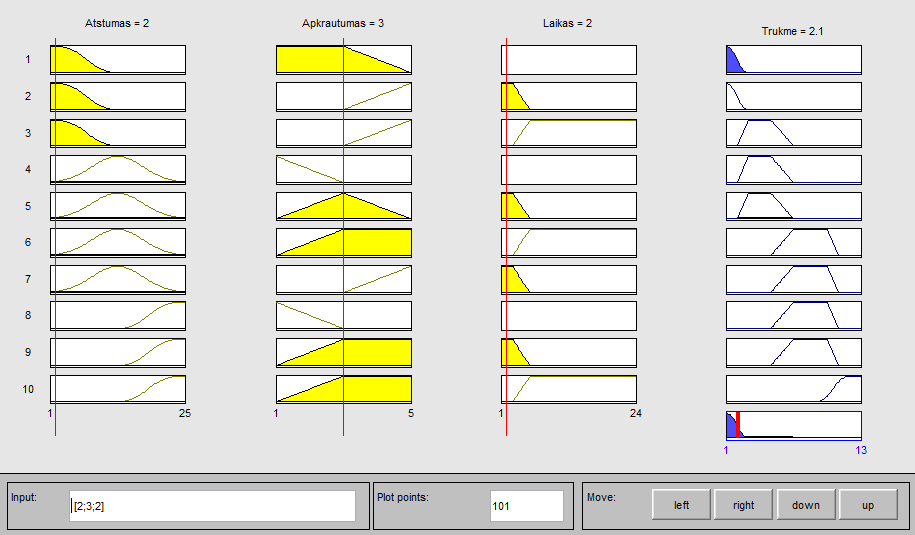
5 lentelė. Taisyklių rinkinys

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nr. | Atstumas |  | Apkrautumas |  | Laikas | Trukme |
| 1. | artimas\_a | *and* | not(didelis\_a) | *and* | null | zaibiskas\_r |
| 2. | artimas\_a | *and* | didelis\_a | *and* | Naktis | zaibiskas\_r |
| 3. | artimas\_a | *and* | didelis\_a | *and* | not(Naktis) | greitas\_r |
| 4. | vidutinis\_a | *and* | mazas\_a | *and* | null | greitas\_r |
| 5. | vidutinis\_a | *and* | vidutinis\_a | *and* | Naktis | greitas\_r |
| 6. | vidutinis\_a | *and* | not(mazas\_a) | *and* | not(Naktis) | letas\_r |
| 7. | vidutinis\_a | *and* | didelis\_a | *and* | Naktis | letas\_r |
| 8. | tolimas\_a | *and* | mazas\_a | *and* | null | letas\_r |
| 9. | tolimas\_a | *and* | not(mazas\_a) | *and* | Naktis | letas\_r |
| 10. | tolimas\_a | *and* | not(mazas\_a) | *and* | not(Naktis) | labaiLetas\_r |

## Išvesties kintamųjų prognozavimas

Sudarytą modelį galime naudoti atsakymų prognozavimui, t.y., pagal įvestis galime spręsti, kokios bus išvestys. Prognozavimas atliktas su 3 skirtingais duomenų (įvesties kintamųjų) rinkiniais:

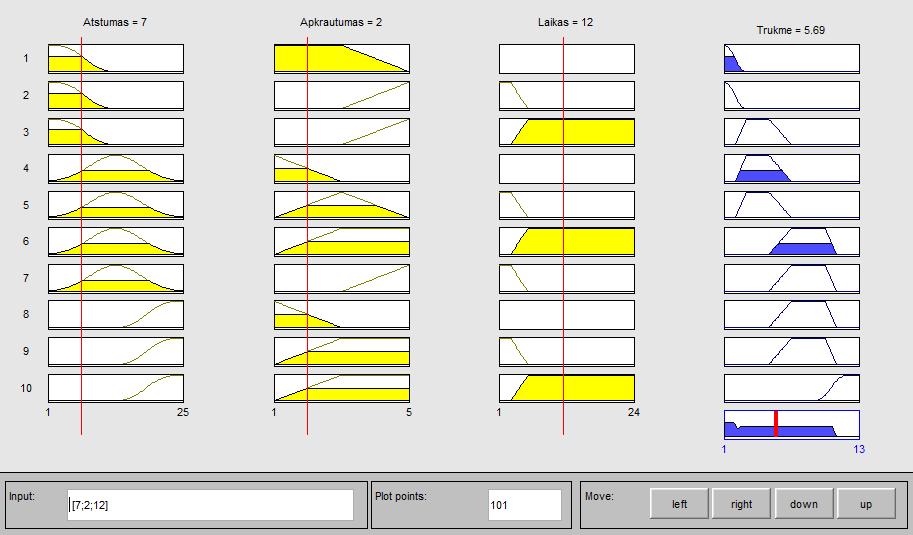
Įvesties kintamųjų reikšmės: Atstumas – 2km, Apkrautumas – 3, Laikas – 2 (02:00h):



15 pav. Fuzzy atsakymų langas su įvestims [2 3 2]

Kaip matome iš 15 pav., su tokiomis įvestimis buvo naudojamos 1 bei 5 taisyklės, tačiau didžiausią svarbą turėjo pirmoji, todėl prognozuojama reikšmė – 2.1 min, kas reiškia, jog reagavimo laikas yra žaibiškas.

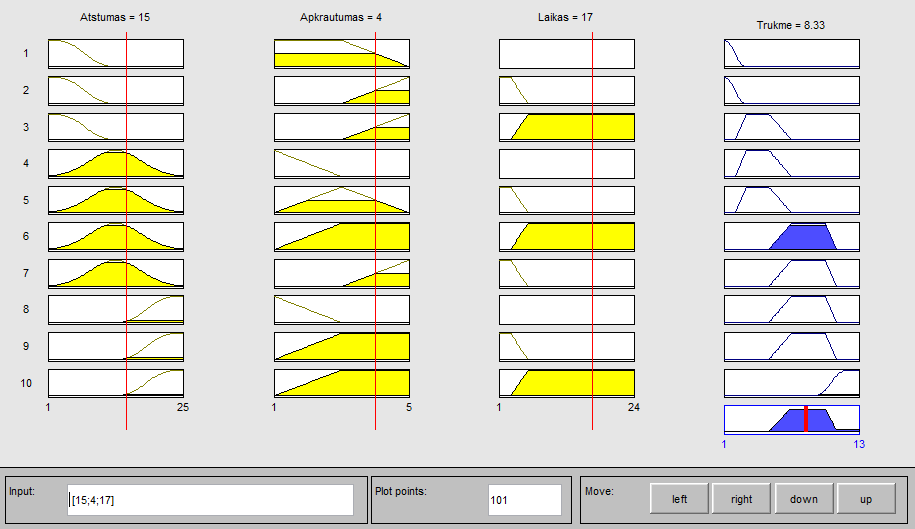
Įvesties kintamųjų reikšmės: Atstumas – 7km, Apkrautumas – 2, Laikas – 12 (12:00h):



16 pav. Fuzzy atsakymų langas su įvestims [7 2 12]

Kaip matome iš 16 pav., su tokiomis įvestimis buvo naudojamos 1, 4 bei 6 taisyklės, tačiau didžiausią svarbą turėjo 4 ir 6, todėl prognozuojama reikšmė – 5.69 min, kas reiškia, jog reagavimo laikas yra greitas.

Įvesties kintamųjų reikšmės: Atstumas – 15km, Apkrautumas – 4, Laikas – 17 (17:00h):

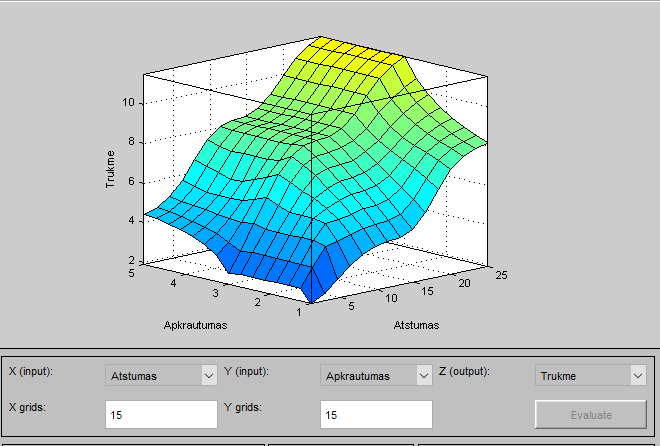


17 pav. Fuzzy atsakymų langas su įvestims [15 4 17]

Kaip matome iš 17 pav., su tokiomis įvestimis buvo naudojamos 6 bei 10 taisyklės, tačiau didžiausią svarbą turėjo šeštoji, todėl prognozuojama reikšmė – 8.33 min, kas reiškia, jog reagavimo laikas yra lėtas.

## Priklausomybių grafiniai paviršiai

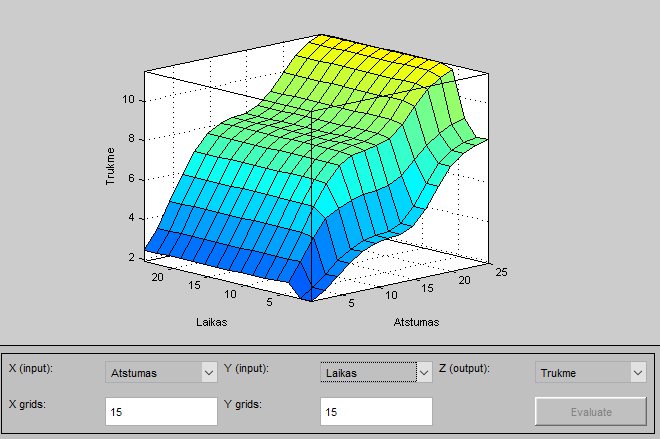
18 pav. pavaizduotoje plokštumoje matome reagavimo trukmės priklausomybę nuo atstumo ir apkrautumo:



18 pav. Reagavimo trukmės priklausomybė nuo apkrautumo ir atstumo

Iš plokštumos matome, kad didėjant gatvių apkrautumui bei atstumui iki objekto, taip pat ilgėja ir reagavimo trukmė.

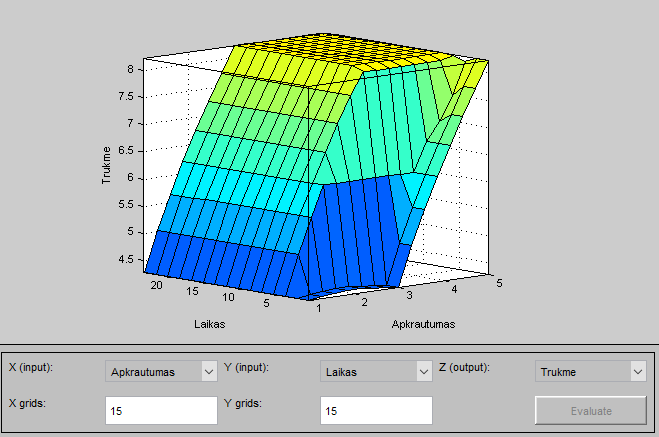
19 pav. pavaizduotoje plokštumoje matome reagavimo trukmės priklausomybę nuo atstumo ir paros laiko:



19 pav. Reagavimo trukmės priklausomybė nuo atstumo ir paros laiko

Iš plokštumos matome, kad trukmei didesnę įtaką daro atstumas, o paros laikotarpis didžiausią įtaką daro tada, kai paros laikas yra naktis, kitu paros laiku įtaka trukmei beveik nematoma.

20 pav. pavaizduotoje plokštumoje matome reagavimo trukmės priklausomybę nuo apkrautumo ir paros laiko:

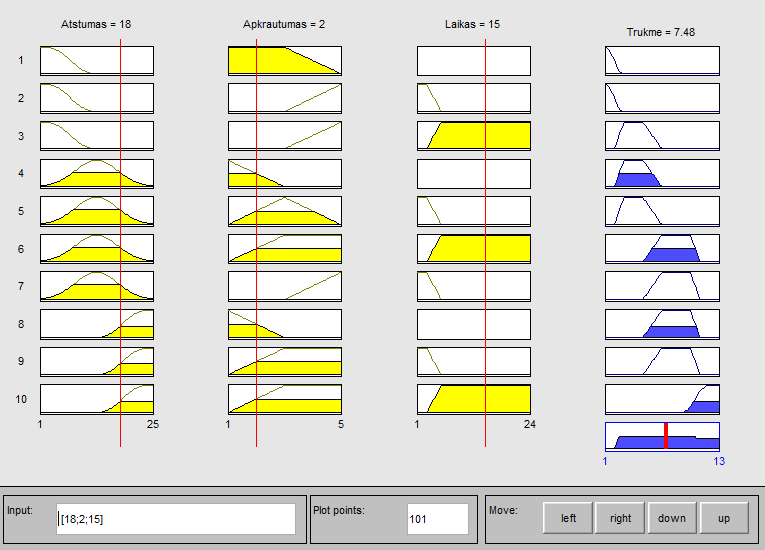


20 pav. Reagavimo trukmės priklausomybė nuo apkrautumo ir paros laiko

Iš plokštumos matome, kad trukmei didesnę įtaką daro apkrautumas, o laikas tokios didelės įtakos neturi, nebent, vėlgi, paros laikas yra naktis, tada apkrautumas yra mažesnis, dėl ko ir reagavimo trukmė yra mažesnė.

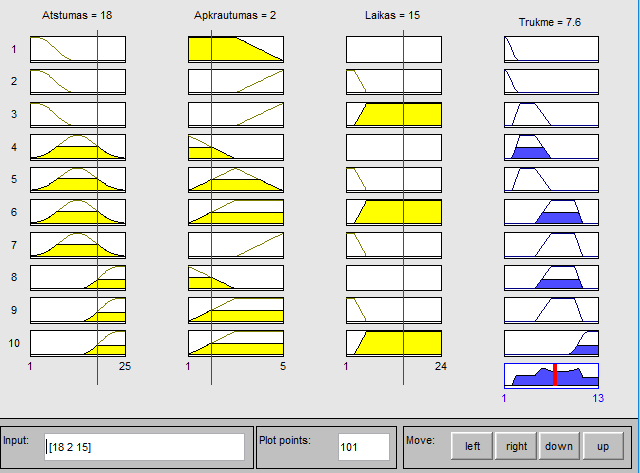
# Agregacijos metodų *max* ir *probor* palyginimas

Palyginimui naudojamos tos pačios įvesties kintamųjų reikšmės: Atstumas – 18 km, Apkrautumas – 2, Laikas – 15 (15:00h). Rezultatai, gauti naudojant *max* agregacijos metodą pavaizduoti 21 paveikslėlyje:



21 pav. Rezultatai, gauti naudojant max agregacijos metodą

Rezultatai, gauti naudojant *probor* agregacijos metodą pavaizduoti 22 paveikslėlyje:

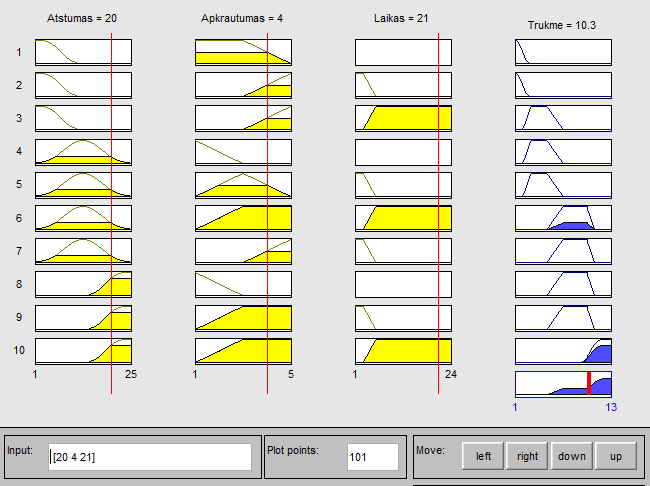


22 pav. Rezultatai, gauti naudojant probor agregacijos metodą

Kaip matome, rezultatai nežymiai skiriasi: naudojant *max* agregacijos metodą, gautas rezultatas – 7,48, o naudojant *probor* agregacijos metodą – 7,6. Tam įtaką daryti gali tai, kad *max* metodas agreguotos funkcijos reikšmę tam tikrame taške tiesiog ima didžiausią funkcijos reikšmę tam tikrame taške iš visų priklausomumo funkcijų verčių, o *probor* metodas sumuoja priklausomumo funkcijų vertes tam tikrame taške ir atima jų sandaugą, todėl rezultatų reikšmės ir skiriasi.

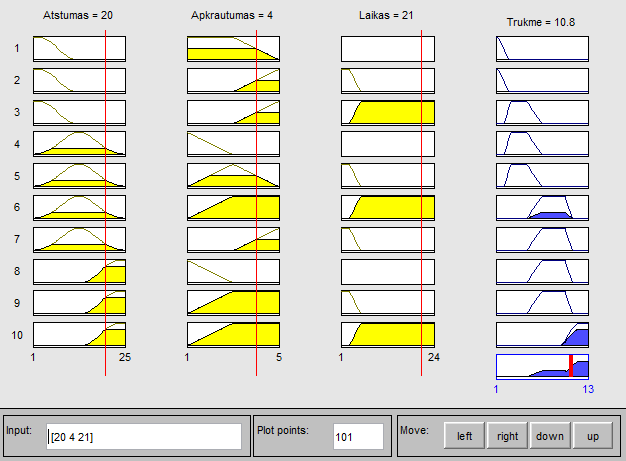
# Defuzikacijos metodų *centroid* ir *bisector* palyginimas

Palyginimui naudojamos tos pačios įvesties kintamųjų reikšmės: Atstumas – 20 km, Apkrautumas – 4, Laikas – 21 (21:00h). Rezultatai, gauti naudojant *centroid* defuzikacijos metodą pavaizduoti 23 paveikslėlyje:



23 pav. Rezultatai, gauti naudojant centroid defuzikacijos metodą

Rezultatai, gauti naudojant *bisector* defuzikacijos metodą pavaizduoti 24 paveikslėlyje:



24 pav. Rezultatai, gauti naudojant bisector defuzikacijos metodą

Kaip matome, rezultatai nežymiai skiriasi: naudojant *centroid* defuzikacijos metodą, gautas rezultatas – 10,3, o naudojant *bisector* defuzikacijos metodą – 10,8. Tam įtaką daryti gali tai, kad *centroid* metodas apskaičiuoja agreguotosios funkcijos reikšmių svorio centrą, o *bisector* metodas randa tašką y0, kuriame agreguotos funkcijos integralas intervale (ymin, y0) yra lygus integralui intevale (y0, ymax).