

طراحی ماشین شستشوی فازی (دو ورودی و دو خروجی) با کنترل کننده منطق فازی

تهیه و تالیف: مهندس محمد ابراهیم ابراهیمیان

ناشر: وب سایت میکرو دیزاینر الکترونیک

www.Melce.ir

زمستان ۱۳۹۶

تقدیم به روح پاک

شهیدان

مدافع حرم

مقدمه مولف

سیستم‌های فازی روش توسعه یافته منطق بولی برای بکاربردن مفاهیم مبهم است. برای بیان ابهام در قالب یک عدد، سیستم فازی تابعی برای عضویت در یک دسته معرفی میکند، که به هر عنصر یک عدد حقیقی بین صفر و یک نسبت میدهد (صفر و یک هم شامل این اعداد میباشند). این عدد نشان دهنده درجه عضویت عنصر نسبت به مجموعه مورد نظر میباشد. عضویت صفر بیانگر این است که عنصر مورد نظر کاملاً خارج از مجموعه است. درحالیکه عدد یک نشاندهنده این است که عنصر مورد نظر کاملاً در مجموعه قرار دارد.

استادان ارجمند، دانشجویان عزیز، مهندسين برق و تمامی متخصصين محترم که به طریقی با منطق فازی و سیستم‌های تخصصی آن در ارتباط هستند، می‌توانند از این مقاله استفاده نمایند در این مقاله سعی بر آن داریم که روش کنترل فازی را در حد مورد نیاز و به بیانی ساده آموزش دهیم. از این رو مطالعه این مقاله خالی از لطف نبوده و به کلیه دوستانی که برای اولین بار قصد یادگیری آن را دارند، توصیه می‌گردد. این مقاله شامل دو فصل است که در مورد ساختار، طراحی و برنامه نویسی گرافیکی پروژه فوق در نرم افزار MATLAB بحث می‌کند.

شما عزیزان در فصل اول با اصول کلی کاربردها و ویژگی‌های منطق فازی همراه با مثال‌های گوناگون آشنا خواهید شد. و در فصل دوم با اصول کلی طراحی سیستم ماشین شستشوی فازی در نرم افزار MATLAB آشنا خواهید شد.

از آنجا که هیچ اثری نمی‌تواند به تنهایی کامل و جامع باشد و این مقاله نیز از این قاعده مستثنی نیست، از اساتید و خوانندگان محترم استدعا دارم بر بنده منت نهاده و پیشنهادات و نظرات سازنده خود را در رابطه با بهتر کردن این مقاله به آدرس جیمیل زیر ارسال نمایید.

ebrahim303032@gmail.com



چکیده

کنترل‌های فازی چند سالی است که کاربردهای مناسبی در وسایل خانگی پیدا کرده‌اند. در این مقاله سعی بر آن شده است تا به قسمتی دیگر از کاربردهای منطق فازی در ماشین لباسشویی پرداخته شود. ماشین‌های لباسشویی که در حال حاضر دارای کنترل فازی هستند، نیاز دارند تا کاربر زمان و میزان پودر شستشو را به آنها اعلام نماید، اما اگر خود وسیله بتواند این کنترل را اتوماتیک انجام دهد، می‌تواند باعث افزایش کیفیت حاصل شستشو شده، از هدر رفتن برق مصرفی نیز جلوگیری می‌کند.

ساختار کلی ماشین شستشوی مورد استفاده یک سیستم دو ورودی دو خروجی است که دو ورودی فوق مقدار اندازه‌گیری شده کثیفی و حجم لباس‌ها بوده و خروجی زمان و میزان پودر مناسب شستشو می‌باشد. بعنوان ورودی (سنسورهای) در این سیستم تعبیه شده این سنسورها که از نوع نوری می‌باشند میزان نوری را که از طرف مقابل ساطع شده و از آب عبور کرده را اندازه‌گیری می‌نمایند. سنسور نوری همچنین می‌تواند معین کند که نوع کثیفی چیست لباس گل آلود است یا چرب؟ گل در آب سریعتر حل می‌شود بنابراین اگر نور دریافتی سرعت کاهش پیدا کند در آن صورت لباس گل آلود است در حالی که اگر لباس روغنی باشد کندتر در آب حل شده و کاهش نور دریافتی کندتر خواهد بود. ماشین همچنین دارای یک سنسور بار ۷ کیلو گرمی می‌باشد که حجم لباس‌ها را ثبت می‌کند واضح است که تعداد لباس‌های بیشتر زمان و پودر بیشتری برای شستشو لازم دارد. این پروژه در حقیقت از یک قسمت کلی، برنامه کامپیوتری (در محیط برنامه نویسی MATLAB) تشکیل شده است. در قسمت برنامه کامپیوتری یک برنامه گرافیکی توسط نرم افزار MATLAB نوشته شده است. که در آن پروتکل‌ها و استانداردها پیاده سازی شده است.

فهرست مطالب

فصل اول: آشنایی با منطق فازی

مقدمه	۱۰
تاریخچه منطق فازی	۱۰
مفاهیم اولیه در منطق فازی	۱۲
مجموعه های فازی	۱۲
عملگرهای اصلی بر روی مجموعه های فازی	۱۳
متغیر های زبانی	۱۳
استدلال و استنتاج تقریبی	۱۴
کنترل کننده های فازی	۱۵
به طور کلی دو منطق تعریف شده است؛ Fuzzy و Crisp	۱۵
مثال ۱. می خواهیم دمای یک تانکر آب را ثابت نگه داریم؛	۱۷
خواص کلی منطق فازی	۲۰
مثال ۲: می خواهیم کنترل دمای تانکر آب را به روش فازی و با دو شیر انجام دهیم؛	۲۱
سیستم های فازی چگونه سیستم هایی هستند؟	۲۶
انواع سیستم های فازی و مقایسه آن ها	۲۷
کاربردهای سیستم فازی	۲۸
مثالی از منطق فازی	۲۹
کنترل فازی کوره سیمان (چند ورودی و چندخروجی)	۲۹
بررسی روش های چهارگانه استفاده از منطق فازی	۳۰
فازی کردن :	۳۰
مثالی برای فازی کردن :	۳۰
استنتاج:	۳۱

مثالی برای استنتاج :	۳۱
پایگاه قواعد :	۳۱
مثالی برای پایگاه قواعد :	۳۱
غیرفازی ساز :	۳۲
مثالی برای غیر فازی ساز :	۳۲
دلایل استفاده از منطق فازی	۳۲
کمبود ها و نواقص سیستم های فازی	۳۳
چه زمانی نباید از منطق فازی استفاده نمود؟	۳۴
جعبه ابزار منطق فازی در MATLAB	۳۴
نحوه عملکرد جعبه ابزار منطق فازی	۳۶
کنترل کننده های عصبی	۳۷
کنترل کننده های فازی- عصبی	۳۸
فصل دوم: شرح پروژه	
مقدمه طراحی سیستم کنترل شستشوی فازی	۴۲
کنترل کننده های فازی	۴۲
طراحی سیستم فازی	۴۲
قدم اول جمع آوری داده ها	۴۳
ساختار کلی ماشین شستشوی مورد بحث!	۴۳
قدم دوم تعیین ورودی و خروجیهای کنترل کننده	۴۵
قدم سوم اختصاص توابع عضویت به هریک از ورودی ها و خروجی ها	۴۵
ورودی های سامانه فازی	۴۵
خروجی های سامانه فازی	۴۹
قدم چهارم تشکیل قوانین فازی	۵۱
رابط گرافیکی جعبه ابزار منطق فازی	۵۵

ویرایشگر سیستم استنتاج فازی (FIS)	۵۷
ویرایشگر توابع عضویت	۵۹
فرآیند تعیین توابع عضویت مربوط به ورودی های پروژه	۶۱
الف) تعیین توابع عضویت مربوط به ورودی اول (Dirty)	۶۱
ب) تعیین توابع عضویت مربوط به ورودی دوم (Weight)	۶۴
فرآیند تعیین توابع عضویت مربوط به خروجی های پروژه	۶۶
الف) تعیین توابع عضویت مربوط به خروجی اول (Time)	۶۶
ب) تعیین توابع عضویت مربوط به خروجی دوم (Flour)	۶۹
ویرایشگر قواعد	۷۱
نمایشگر قواعد فازی	۷۴
نمایشگر سطوح	۷۶
ورود و خروج داده ها در رابط های گرافیکی	۷۸
برنامه نویسی در محیط m.file نرم افزار MATLAB	۷۹
چند مثال از پروژه ماشین شستشو فازی	۸۴
مثال اول	۸۴
مثال دوم	۸۶
مثال سوم	۸۷
فهرست مراجع	۸۸

فصل اول

آشنایی با منطق فازی

مقدمه

مفاهیم نادقیق بسیاری در پیرامون ما وجود دارند که آنها را به صورت روزمره در قالب عبارتهای مختلف بیان می کنیم . به این جمله دقت کنید: " هوا خوب است." هیچ کمیتی برای خوب بودن هوا مطرح نیست تا آن را اندازه بگیریم بلکه این یک حس کیفی است. در واقع مغز انسان با در نظر گرفتن فاکتور های مختلف و براساس تفکر استنتاجی جملات را تعریف و ارزش گذاری می نماید که مدل سازی آنها به زبان و فرمولهای ریاضی اگر غیر ممکن نباشد کاری بسیار پیچیده خواهد بود. منطق فازی تکنولوژی جدیدی است که شیوه هایی را که برای طراحی و مدل سازی یک سیستم نیازمند ریاضیات پیچیده و پیشرفته است، با استفاده از مقادیر زبانی و دانش فرد خبره جایگزین می سازد.

تاریخچه منطق فازی

خلق، بسط و گسترش اندیشه فازی توسط « پروفیسور لطفی زاده » استاد دانشگاه کالیفرنیا، برکلی. در سال 1965 میلادی این دانشمند ایرانی اولین مقاله خود در زمینه « فازی » را با عنوان « مجموعه های فازی » منتشر کرد. شاید در تصور کسی نمی گنجید که این مقاله اولین جرقه از یک جهان بینی جدید در عرصه ریاضیات و علوم و اولین قدم در معرفی بینشی نو و واقع گرایانه از جهان، در چارچوب مفاهیمی کاملاً بدیع اما بسیار سازگار با طبیعت انسان باشد.

تفکر فازی از دیدگاه فلسفی نشأت می گیرد که سابقه ای چند هزار ساله و به قدمت تاریخ فلسفه دارد. همانگونه که فلسفه ادیان الهی با طبیعت و سرشت انسان سازگار است، تفکر فازی با الهام از فلسفه شرقی جهان را همانگونه که هست معرفی می کند. در فلسفه ارسطویی که در مقابل فلسفه شرقی قرار دارد، همه چیز چیز به دو دسته سیاه و سفید یا بله و خیر تقسیم می شود. مفاهیم منطقی و نتایج حاصله از استدالات منطقی نیز در

فلسفه ارسطویی هیچگونه حالت میانه‌ای ندارد. در این فلسفه نمی‌توان تا اندازه‌ای راستگو و ضمناً کمی دروغ گو بود. نمی‌شود همزمان نسبتاً جوان و تا اندازه‌ای هم پیر بود. در فلسفه ارسطویی مرزها کاملاً مشخص و تعریف شده هستند.

در تفکر فازی مرز مشخصی وجود ندارد و تعلق عناصر مختلف به مفاهیم و موضوعات گوناگون نسبی است. به این ترتیب می‌بینیم که این تفکر تا چه اندازه با طبیعت جهان و انسان سازگار است. تفکر فازی دیدگاهی تازه را معرفی می‌کند که تعمیم منطق ارسطویی است؛ اما براساس این دیدگاه، ریاضیات کلاسیک نیز که بر منطق ارسطویی استوار است زیر سوال می‌رود. از این رو مخالفت‌های بسیاری را نیز از بدو شکوفایی خود به همراه داشته است.

منطق دیجیتال متداول که فقط از یک تصمیم "بله" یا "خیر" تشکیل شده است، برای پروژه‌های شامل هوش مناسب نمی‌باشد. اگر یک حالت سوم متناظر با جواب "شاید" را پیاده سازی نماییم، کمی بیشتر به مفهوم هوش نزدیک شده‌ایم. این روش اساس منطق فازی است.

دانشمندان کامپیوتر، رشته "هوش مصنوعی" را ساخته‌اند زیرا دانش و معلومات در نظر آن‌ها همان قوانین هستند که با منطق دیجیتال قابل نوشتن می‌باشند، ولی با گذشت مدت زمان زیاد و صرف هزینه بسیار در این رشته هنوز نتوانسته‌اند محصولات هوشمند قابل توجهی را عرضه کنند.

مهندسان فازی، نرم افزارها و تراشه‌هایی را تهیه می‌کنند که می‌توانند به سیستم‌های کامپیوتری قدرت استدلالی نزدیک به قدرت استدلال انسان بدهند. این توانایی باعث می‌شود ماشین‌ها هوشمندتر شده و کار با آن‌ها ساده تر گردد. محققان فازی نیز این پیشرفت در زمینه هوشمندی را مدیون قوانین هستند ولی نه قوانین بکار گرفته شده توسط مهندسان هوش مصنوعی بلکه "قوانین فازی"؛ و علت آن نیز در نسبی بودن آن قوانین می‌باشد که در قوانین دیجیتال چنین چیزی به چشم نمی‌خورد.

انواع دیگری از سیستم‌های فازی با استفاده از تجربیات خود توانایی یادگیری و برنامه ریزی دارند. این نوع از سیستم‌های فازی، خیلی سریع می‌توانند هوشمند شوند.

مفاهیم اولیه در منطق فازی

سیستم‌های فازی یک فرا مجموعه از منطق بولی است که بر مفهوم درستی نسبی، دلالت می‌کند. منطق کلاسیک هر چیزی را بر اساس یک سیستم دوتائی نشان می‌دهد (درست یا غلط، ۰ یا ۱، سیاه یا سفید) ولی منطق فازی درستی هر چیزی را با یک عدد که مقدار آن بین صفر و یک است نشان می‌دهد. مثلاً اگر رنگ سیاه را عدد صفر و رنگ سفید را عدد ۱ نشان دهیم، آن گاه رنگ خاکستری عددی نزدیک به صفر خواهد بود. منطق فازی معتقد است که ابهام در ماهیت علم است. بر خلاف دیگران که معتقدند که باید تقریب‌ها را دقیق‌تر کرد تا بهره‌وری افزایش یابد، لطفی‌زاده معتقد است که باید به دنبال ساختن مدل‌هایی بود که ابهام را به عنوان بخشی از سیستم معرفی کند.

مجموعه‌های فازی

در فضای U ، مجموعه فازی A ، با یک تابع عضویت $\mu_A: U \rightarrow [0,1]$ مشخص می‌شود. یعنی برای هر عضو u در U یک مقدار $\mu_A(u)$ در فاصله $[0,1]$ تعریف می‌شود، که درجه عضویت u در A نامیده می‌شود.

یک مجموعه فازی را برای مجموعه مرجع پیوسته به شکل زیر نشان می‌دهیم:

$$A = \int \limits_U (\mu_A(u)/u)$$

تکیه گاه مجموعه فازی، مجموعه u هایی است که در رابطه $\mu_A(u) > 0$ صدق کند.

عملگرهای اصلی بر روی مجموعه‌های فازی

از آنجا که هر مجموعه فازی را با تابع عضویت آن بیان می‌کنیم، اعمال اصلی بر روی مجموعه‌های فازی نیز بر حسب توابع عضویت قابل تعریف هستند که چند نمونه از این عملگرها در ادامه آورده شده است:

• اجتماع

$$\mu_{A \cup B}(u) = \max\{\mu_A(u), \mu_B(u)\}$$

• اشتراک

$$\mu_{A \cap B}(u) = \min\{\mu_A(u), \mu_B(u)\}$$

• مکمل

$$\mu_{\bar{A}}(u) = 1 - \mu_A(u)$$

متغیرهای زبانی

متغیرهای زبانی یا محاوره‌ای، متغیرهایی هستند که مقادیر آنها نه اعداد، بلکه کلمات یا جملات در زبان طبیعی می‌باشند. این متغیرها یکی از ابزارهای اساسی منطق فازی و استنتاجات تقریبی می‌باشند؛ هر متغیر زبانی X بوسیله یک پنج تایی مرتب به فرم $(X, T(x), G, U, M)$ مشخص می‌شود که در آن X نام متغیر است که در مجموعه U تغییر می‌کند و G یک قاعده نحوی (معمولا به فرم دستور زبان) برای تولید مجموع نرم‌های $T(x)$ مربوط به متغیر X است و M یک قاعده معنایی است که به هر نرم از $T(x)$ معنای آن را مربوط می‌سازد. $M(x)$ نیز زیر مجموعه فازی U است.

استدلال و استنتاج تقریبی

گزاره‌های شرطی اهمیت اساسی در استنتاج و استخراج نتایج از مجموعه‌ای مفروضات، دارند. قوانین کنترل

کننده‌های فازی، یک مجموعه گزاره‌های شرطی هستند که به یکی از دو صورت نوعی زیر بکار برده می‌شوند.

1) If X_1 is A_1 and X_2 is A_2 and ... Then Y_1 is B_1 and Y_2 is B_2 and ...

2) If X_1 is A_1 and X_2 is A_2 and ... Then $Y_1 = F_1(X_1, X_2, \dots)$ and

$Y_2 = F_2(X_1, X_2, \dots)$ and ...

که X_i ها متغیرهای زبانی و A_i ها مجموعه‌های فازی هستند که به عنوان مقادیر زبانی این متغیرها

می‌باشند و f یک تابع ریاضی است که می‌تواند یک رابطه خطی به شکل زیر باشد:

$$y = f(X_1, X_2, \dots) = c_0 + c_1 x_1 + c_2 x_2 + \dots$$

که C_i ها مقادیر عددی غیر فازی و x_1 و x_2 و ... مقادیر غیر فازی به ترتیب مربوط به متغیرهای زبانی x_1

و x_2 و ... هستند.

با داشتن مقادیر مربوط به قسمت مقدم قوانین (ورودی‌ها)، روش‌های مختلفی برای استنتاج و بدست آوردن

خروجی‌ها وجود دارد که یک روش ساده آن به صورت زیر می‌باشد:

فرض کنید n قانون داریم که قانون i ام به شکل زیر است:

$$R_i: \text{If } X_1 \text{ is } A_1^i \text{ and } X_2 \text{ is } A_2^i \text{ Then } y \text{ is } B^i$$

در این قانون X_1 و X_2 ورودی‌های یک کنترلر و y خروجی آن هستند بطوریکه مقدار عددی X_1 برابر

x_1^* و مقدار عددی X_2 برابر x_2^* (که x_1^* و x_2^* اعداد حقیقی هستند) می‌باشد. برای هر قانون یک

درجه همسازی به شکل زیر محاسبه می‌کنیم:

$$w_i = \min \left(\mu_{A_1^i}(x_1^*), \mu_{A_2^i}(x_2^*) \right)$$

لازم به توضیح است که اگر در قسمت مقدم قوانین بجای رابط and از رابط or استفاده شده بود، در رابطه بالا از عملگر max بجای min استفاده می شد.

با داشتن w_i ها می توان خروجی استنتاج شده از قوانین را به شکل زیر بدست آورد:

$$y = \frac{\sum_{i=1}^n w_i \cdot b_i}{\sum_{i=1}^n w_i}$$

که در رابطه فوق، b_i محل پیک تابع عضویت B_i است.

کنترل کننده های فازی

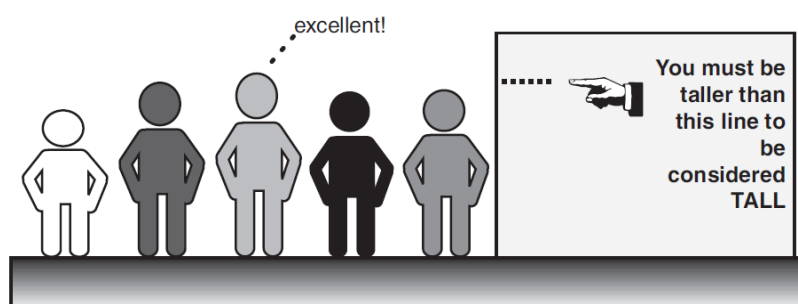
یکی از روشهای هوشمندی که امروزه در کنترل سیستم ها بکار می رود، کنترل فازی است. در سال 1985 با راهیابی روشهای فازی در کنترل ربات، گام جدیدی در بکارگیری روشهای هوشمند در کنترل ربات برداشته شد. معمولاً از کنترل کننده های فازی در مواقعی که دقت بالایی مد نظر نبوده و مدل مناسبی برای سیستم موجود نباشد، همچنین انسان در کنترل سیستم، بهتر و سریعتر از ماشین عمل کند، استفاده می شود. البته یکی از شروط اصلی موفقیت در جایگزین نمودن کنترل کننده های فازی بجای کنترلگرهای انسانی، بیان صحیح تجارب انسان در قالب قوانین شرطی می باشد.

به دلیل اینکه روشهای کنترل فازی متکی به مدل ریاضی سیستم نمی باشند، می توان در مواقعی که ساختار و یا پارامترهای سیستم تحت کنترل، بطور دقیق مشخص نبوده و یا خیلی پیچیده باشند، علوم تجربی انسان در کنترل این سیستمها را در قالب کنترل کننده های فازی پیاده سازی نمود.

به طور کلی دو منطق تعریف شده است؛ Fuzzy و Crisp.

در منطق Crisp کلیه توضیحات در ارتباط با موضوع مورد بحث تماماً بصورت عددی بیان می شوند، به طور مثال اگر فرض را در تمام نقاط روی زمین بر این بنا نهیم که ۱ متر قد، یعنی کوتاه قد و ۲ متر قد، یعنی بلند قد، پس بنا بر قرارداد فوق تنها کسانی که قدشان دقیقاً ۱ متر است، قد کوتاه و همچنین فقط کسانی که قدشان

دقیقا ۲ متر باشد، قد بلند هستند؛ به این ترتیب کسانی که قدشان کمتر از ۱ متر (حتی ۹۹،۹۹۹ سانتیمتر) یا بین ۱ متر و ۲ متر (حتی ۱،۰۰۱ متر یا ۱،۹۹۹ متر) و یا بیش از ۲ متر (حتی ۲،۰۰۱ متر) باشد، نه قد کوتاه هستند و نه قد بلند به حساب می‌آیند، به بیانی دیگر هرکس که قد کوتاه باشد، دارای قدی برابر ۱ متر و هرکس که قد بلند باشد یعنی قدش ۲ متر است! پس به این دلیل و برای مشخص کردن قد همه افراد روی زمین بایستی برای هر قد موجود روی زمین یک تعریفی داشته باشیم! که این غیر ممکن می‌باشد.



شکل ۱-۱: مجموعه افراد بلند قد

در مثالی دیگر به دمای آب می‌پردازیم؛ اگر صفر درجه، سرد و ۱۰۰ درجه، گرم تعریف شده باشد پس هر آب سردی دمایش صفر درجه مطلق و هر آب گرمی دمایش ۱۰۰ درجه است! حال اگر آبی داشته باشیم که دمایش ۵۵ درجه باشد، دمایش چقدر است؟ جواب این است که هیچ! یعنی هیچ آبی در دنیا یافت نمی‌شود که دمایش ۵۵ درجه یا هر مقداری دیگر بجز صفر یا ۱۰۰ درجه باشد! و یا پاسخی دیگر برای این پرسش به این نحو است که هر آبی با دمای کمتر از صفر درجه یا کمی بیشتر از آن، سرد و هر آبی با دمای بیشتر از ۱۰۰ درجه یا کمی کمتر از آن، گرم است ولی این «کمی» چقدر است و هرکس چه تعریفی از آن دارد، مشخص نیست. حال، آبی با دمای ۵۰ درجه، یا به هیچ گروهی تعلق ندارد و یا هرکس به دلخواه خود آن را تعریف و بیان می‌کند.

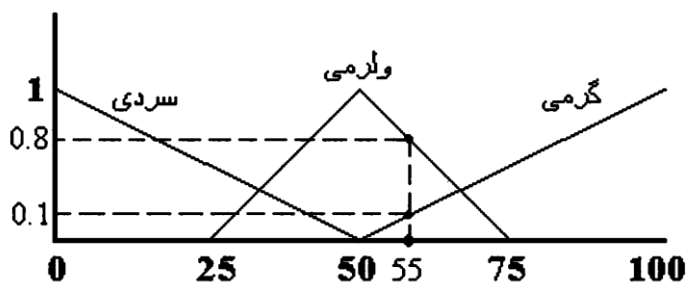
همانطور که ملاحظه شد در منطق Crisp برای بیان صحیح هر چیز بایستی کاملاً عددی پیش برویم و با قراردادهای بیشماری سر و کار داریم که این خود، کار را بسیار سخت و بهتر بگوییم غیر عملی می‌سازد.

در منطق Fuzzy بر خلاف منطق قبلی، تماما با عناوین و اسامی سر و کار داریم؛ به طور مثال می‌گوییم دمای آب سرد است یا ولرم است یا گرم است یا خیلی سرد است یا خیلی گرم است و یا درصدی گرم و درصدی نیز ولرم است و یا به این ترتیب برای هر دمایی از آب می‌توانیم دمای آن را نه بطور صریح و واضح ولی با بیانی ساده و قابل فهم مشخص کنیم. در اینجا می‌بینیم که با مقادیر عددی مطلق سر و کار نداریم و این دستمان را باز می‌گذارد و نحوه بیان و فهم قضیه را آسان می‌نماید. به بیانی ساده تر در منطق Fuzzy با گرمی، سردی، ولرمی، کمی، زیادی، کوتاهی، بلندی و از این قبیل عناوین سر و کار داریم.

با یک مثال به توضیح کنترل فازی می‌پردازیم؛

مثال ۱. می‌خواهیم دمای یک تانکر آب را ثابت نگه داریم؛

یک « فضای ورودی » به شکل زیر برای کنترل دمای آب تانکر تعریف می‌کنیم:



شکل ۱-۲: تابع تعلق ورودی تانکر آب

محور افقی در فضای ورودی فوق، مشخصه درجه آب می‌باشد.

اولین موضوع مهم در کنترل فازی، فضای ورودی می‌باشد؛ در این مثال برای فضای ورودی، سه تابع

تعریف کرده ایم: سردی، گرمی، ولرمی.

$$\text{تابع سردی: } y = 1 - \frac{x - 0}{50 - 0}$$

$$\text{تابع ولرمی: } y = \frac{x - 25}{50 - 25} \quad ; \quad y = 1 - \frac{x - 50}{75 - 50}$$

$$y = \frac{x-50}{100-50} \quad \text{تابع گرمی:}$$

فرض کنید دماسنج عدد ۵۵ را نشان دهد؛ در مرحله آغازین بایستی «درجه عضویت» این مقدار عددی را نسبت به هریک از توابع تعریف شده در فضای ورودی تعیین کنیم؛

با مشاهده فضای ورودی و از روی «توابع عضویت» میزان تعلق عدد ۵۵ به هریک از توابع را تعیین می‌کنیم:

درجه عضویت به سردی: صفر

درجه عضویت به ولرمی: ۰,۸

درجه عضویت به گرمی: ۰,۱

در هر سیستم کنترل فازی، به ازای هر فضای ورودی، تعدادی «قوانین حاکم بر سیستم» که دومین موضوع مهم در کنترل فازی می‌باشد، وجود دارند. قوانین حاکم بر فضای ورودی این مثال به شرح زیر تعریف شده‌اند:

R1: اگر سرد است، شیر A خیلی باز شود.

R2: اگر ولرم است، شیر A متوسط باز شود.

R3: اگر گرم است، شیر A کم باز شود.

در این مثال شیر A یک عامل (محرک) نامیده می‌شود.

سومین و آخرین موضوع اصلی در کنترل فازی، «فضای خروجی» است، که در این مثال فضای خروجی

شیر A مطابق شکل زیر تعریف شده است:



شکل ۱-۳: تابع تعلق خروجی شیر A

محور افقی در فضای خروجی فوق، مشخصه درجه پیچ شیر A می‌باشد و لزومی به قرینه بودن آن نیست.

در مرحله دوم، بایستی با توجه به درجه عضویت عدد ۵۵ در سه تابع ورودی و با توجه به قوانین حاکم و

فضای خروجی، مقدار خروجی کنترلر را محاسبه کنیم. به این ترتیب که:

از قانون اول، "ماکزیمم تابع «خیلی باز» را در "درجه عضویت عدد ۵۵ در تابع «سردی» ضرب کرده، و از قانون دوم، "ماکزیمم تابع «متوسط باز» و "درجه عضویت عدد ۵۵ در تابع «ولرمی» را در هم ضرب می‌کنیم، و با توجه به قانون سوم، "ماکزیمم تابع «کم باز» را در "درجه عضویت عدد ۵۵ در تابع «گرمی» ضرب می‌کنیم. حال "مجموع سه مقدار حاصله" را بخش بر "مجموع سه درجه عضویت ذکر شده" کرده و جواب را به عنوان خروجی کنترلر (درجه‌ای که شیر A باید روی آن قرار گیرد) در نظر می‌گیریم. به این ترتیب داریم:

$$\frac{0(50) + 0.8(0) + 0.1(-50)}{(0 + 0.8 + 0.1)} = \frac{(-5)}{0.9} = -5.5$$

مطالب اخیر را به عبارتی ساده تر بیان میکنیم:

از آنجا که طبق قانون اول داریم: (اگر سرد است، شیر A خیلی باز شود) و از طرفی دمای ۵۵ درجه در

فضای ورودی، به تابع «سردی» اصلاً تعلق ندارد (سرد نیست) پس قانون اول در تنظیم شیر A اصلاً تاثیری

ندارد.

براساس قانون دوم (اگر ولرم است، شیر A متوسط باز شود) و از آنجا که عدد ۵۵ با اندازه (۰,۸) متعلق به تابع « ولرمی» است (۸۰ درصد ولرم است)، پس بایستی شیر A، " ۸۰ درصد به طور متوسط باز شود" که متوسط ترین درجه باز شدن شیر A، درجه (۰) است. پس این قانون به مقدار " ۸۰ درصد از متوسط ترین درجه باز شدن شیر (۰)" ، در تنظیم شیر A تاثیر می گذارد.

با توجه به قانون سوم نیز (اگر گرم است، شیر A کم باز شود) و از آنجا که عدد ۵۵ به اندازه (۰,۱) متعلق به تابع « گرمی» است (۱۰ درصد گرم است)، پس بایستی شیر A، ۱۰ درصد کم باز شود " که کمترین درجه باز شدن شیر A، درجه (۵۰-) است. پس این قانون به مقدار " ۱۰ درصد از کمترین درجه باز شدن شیر (۵۰-) "(، در تنظیم شیر A برابر خواهد شد با:

$$\frac{0(50) + 0(0) + 1(-50)}{(0 + 0 + 1)} = -50$$

زیرا:

درجه عضویت به سردی: صفر

درجه عضویت به ولرمی: صفر

درجه عضویت به گرمی: ۱

پس درجه شیر A باید روی مقدار (۵۰-) قرار گیرد (کم باز شود).

خواص کلی منطق فازی

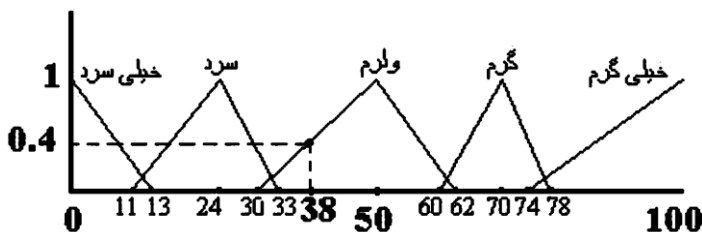
با کمی دقت در مثال ذکر شده، می توان به خواص کلی کنترل فازی پی برد، در زیر نکات مهم را مرور می کنیم:

- سه موضوع کنترل فازی: فضای ورودی، قوانین حاکم بر سیستم، فضای خروجی.
- هر سیستم کنترل فازی، شامل یک یا چند « فضای ورودی» می باشد.
- هر فضای ورودی، از یک یا چند « تابع عضویت» تشکیل شده است که تعداد آنها معمولاً بین ۳ تا ۵ تابع می باشد و از آنجا که انسان دارای ۵ حس گوناگون است و در آن واحد بیش از ۵ کار را نمی تواند انجام دهد، نهایتاً ۵ تابع و البته به ندرت ۸ یا ۹ تابع عضویت هم تعریف می شود.

- توابع عضویت هر فضای ورودی باید از یک جنس باشند.
 - مجموع توابع عضویت تعریف شده در فضای ورودی، در هر نقطه (عدد Crisp ورودی) مقداری بین ۰ و ۱ و یا برابر با آنهاست.
 - به ازای هر مقدار Crisp ورودی، هر یک از توابع عضویت، یک « درجه عضویت» برای آن مقدار، در خود تعریف می کنند؛ که معرف میزان تعلق آن ورودی به مجموعه های فازی می باشد.
 - اگر یک فضای ورودی داشته باشیم، به تعداد توابع عضویت ورودی، « قانون حاکم» موجود است. و اگر بیش از یک فضای ورودی داشته باشیم، به تعداد قوانین حاکم بر سیستم، برابر است با حاصلضرب تعداد توابع عضویت فضاهای ورودی در یکدیگر.
 - در هر قانون، یک یا چند عامل (علت) وجود دارد.
 - به تعداد عاملها (محرکها) در قوانین، « فضای خروجی» داریم.
 - در فضای خروجی هر یک از عاملها، به تعداد حالت هایی که همان محرک (در قوانین حاکم) می پذیرد، « تابع عضویت» وجود دارد.
 - برای محاسبه خروجی کنترلر به سه روش می توان عمل کرد؛ ماکزیمم مقدار، متوسط مقدار و متوسط ماکزیمم. که به عنوان نمونه، در مثال فوق که از روش ماکزیمم مقدار استفاده شده بود، ماکزیمم هر یک از توابع را در درجه عضویت عدد Crisp ورودی ضرب کردیم.
- با ذکر مثالی دیگر با این خصوصیات بیشتر آشنا می شویم:

مثال ۲: می خواهیم کنترل دمای تانکر آب را به روش فازی و با دو شیر انجام دهیم؛

یک فضای ورودی با توابع مثلثی به صورت زیر داریم:



شکل ۱-۴: تابع تعلق ورودی تانکر آب

محور افقی در فضای ورودی فوق، نشانگر دمای آب تانکر است و همانطور که در شکل پیداست، لزومی به قرینه بودن توابع عضویت آن نیست.

فضای ورودی از ۵ تابع عضویت مثلثی تشکیل شده است:

$$\text{تابع خیلی سرد: } y = 1 - \frac{x-0}{13-0}$$

$$\text{تابع سرد: } y = \frac{x-11}{24-11} ; y = 1 - \frac{x-24}{33-24}$$

$$\text{تابع ولرم: } y = \frac{x-30}{50-30} ; y = 1 - \frac{x-50}{62-50}$$

$$\text{تابع گرم: } y = \frac{x-60}{70-60} ; y = 1 - \frac{x-70}{78-70}$$

$$\text{تابع خیلی گرم: } y = \frac{x-74}{100-74}$$

قوانین حاکم بر سیستم، به قرار زیر می باشند:

R1: اگر خیلی سرد است، شیر A باز شود و شیر B بسته شود.

R2: اگر سرد است، شیر A باز شود و شیر B باز شود.

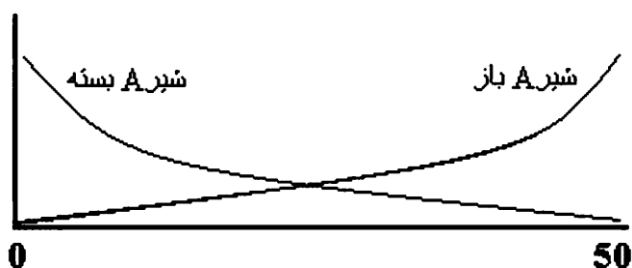
R3: اگر ولرم است، شیر A بسته شود و شیر B بسته شود.

R4: اگر گرم است، شیر A باز شود و شیر B باز شود.

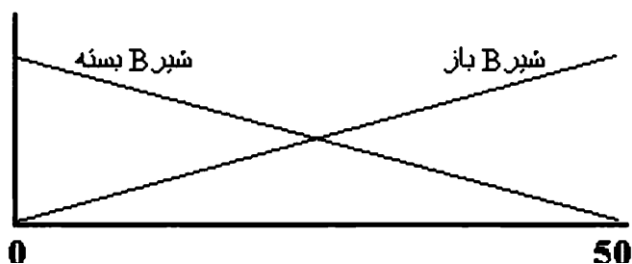
R5: اگر خیلی گرم است، شیر A بسته شود و شیر B باز شود.

همانطور که می‌بینید، در قوانین دو عامل (شیر A و B) وجود دارند، از این رو سیستم ما دو فضای خروجی

دارد؛ فضای خروجی شیر A و فضای خروجی شیر B:



شکل ۱-۵: تابع تعلق خروجی شیر A



شکل ۱-۶: تابع تعلق خروجی شیر B

محور افقی دو فضای خروجی فوق به ترتیب، درجه پیچ شیر A و درجه پیچ شیر B هستند که در اینجا درجه

صفر یعنی شیر کاملاً (۱۰۰ درصد) بسته و درجه ۵۰ یعنی شیر کاملاً باز.

فرض کنید دماسنج عدد ۳۸ را نشان می‌دهد؛ درجه عضویت عدد ۳۸ به هریک از توابع عضویت ورودی را

بدست می‌آوریم:

درجه عضویت در تابع خیلی سرد: صفر

درجه عضویت در تابع سرد: صفر

درجه عضویت در تابع ولرم: ۰,۴

درجه عضویت در تابع گرم: صفر

درجه عضویت در تابع خیلی گرم: صفر

خروجی عملگر A را به روش متوسط مقدار و خروجی عملگر B را به روش ماکزیمم مقدار محاسبه می کنیم:

$$\frac{0.4(5)}{0.4} = 5 \quad \text{خروجی عملگر A:}$$

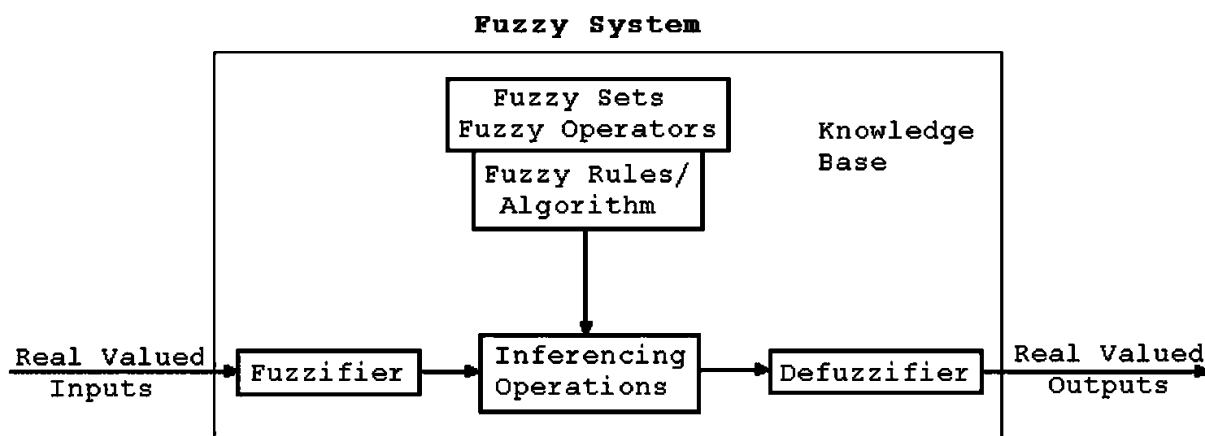
برای محاسبه مقدار متوسط بایستی متوسط انتگرال تابع خروجی مربوطه (مطرح شده در قانون) را بدست آوریم؛

در اینجا فرض کرده ایم که مقدار متوسط تابع "شیر A بسته" در فضای خروجی شیر A، (۵) باشد.

$$\frac{0.4(0)}{0.4} = 0 \quad \text{خروجی کنترلر B:}$$

همانطور که در نمودار تابع خروجی "شیر B بسته" در فضای خروجی شیر B پیداست، مقدار ماکزیمم تابع (ماکزیمم مقدار بسته بودن شیر B)، درجه (صفر) است.

باتوجه به دو مثال فوق نتیجه می گیریم که یک کنترل کننده فازی از توابع عضویت مجموعه های فازی، تعدادی قوانین فازی و یک روش استنتاج فازی تعریف می شود. دیاگرام بلوکی یک سیستم کنترل کننده فازی در شکل زیر آمده است:



بنابراین یک کنترل کننده فازی بر اساس سه مفهوم اصلی مشخص می‌شود:

۱. فازی کردن مقادیر اندازه گیری شده (Fuzzification)

۲. مکانیزم استنتاج (Decision Making)

۳. فازی زدایی یک مجموعه فازی (Defuzzyfication)

ورودی‌ها به قسمت فازی گر داده می‌شوند و توابع عضویت تعریف شده در این قسمت آن‌ها را به ورودی‌های مناسب برای ماشین استنتاج فازی تبدیل می‌کنند. ماشین استنتاج فازی برای استفاده از ورودی‌های دریافت شده از قسمت فازی گر و با توجه به قواعد فازی که به آن داده شده است، خروجی مناسب را تولید می‌کند. این خروجی به قسمت فازی زدا رفته و در آنجا خروجی نهایی کنترلر تولید می‌شود.

نوع طراحی قوانین و توابع عضویت فازی به نوع سیستمی که قرار است کنترل شود، وابسته می‌باشد. در حالی که جهت استنتاج فازی می‌توان مستقل از سیستم موردنظر، از میان روش‌های متعدد استنتاج (از قبیل روش‌های Min-Max, Max-Product, Max, Min و...) یکی را جهت استنتاج قوانین انتخاب نمود.

بطور کلی چهار روش برای تعیین قوانین کنترل کننده و توابع عضویت وجود دارد:

۱. استفاده از تجربه فرد متخصص در کنترل پروسه مورد نظر.

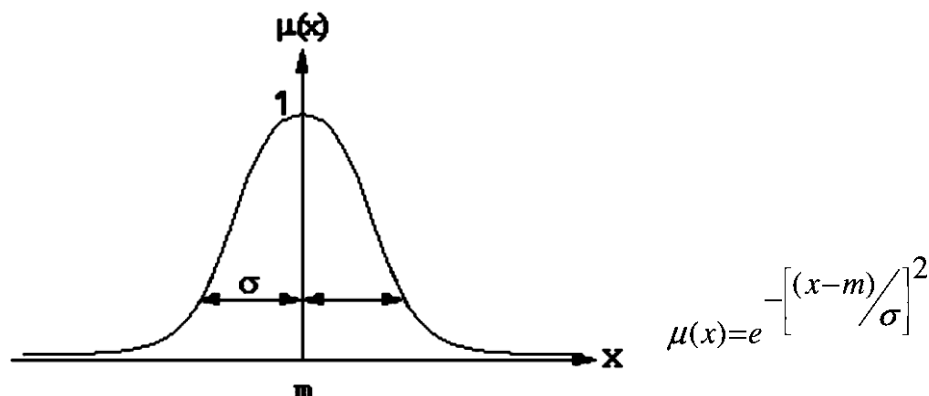
۲. از طریق آزمایش و سعی و خطا روی پروسه یا مدل ریاضی آن.

۳. استفاده از مدل فازی پروسه جهت استخراج قوانین کنترلی مناسب برای پروسه.

۴. استفاده از روش‌های یادگیری.

در روش آخر که جدیدتر از سه روش قبلی است، سعی می‌شود که با بهره گیری از مکانیزم‌های یادگیری، قواعد کنترل سیستم و مجموعه توابع عضویت، بدست آید. این کنترل کننده‌ها که "کنترل کننده‌های فازی خود سازمانده" نامیده می‌شوند، می‌توانند بصورت وفقی نیز کار کنند.

شایان ذکر است که توابع عضویت مجموعه‌های فازی به فرم‌های مختلفی همچون مثلثی، دوزنقه‌ای، گوسی، زنگوله‌ای و ... در نظر گرفته می‌شوند. در زیر فرم کلی یک تابع گوسی را ملاحظه می‌کنید:



سیستم‌های فازی چگونه سیستم‌هایی هستند؟

سیستم‌های فازی سیستم‌ها مبتنی بر دانش یا قواعد می‌باشد. قلب یک سیستم فازی یک پایگاه دانش بوده که از قواعد اگر-آنگاه فازی تشکیل شده است. یک قاعده اگر-آنگاه فازی یک عبارت اگر-آنگاه بوده که بعضی کلمات آن بوسیله توابع تعلق پیوسته مشخص شده‌اند.

بعنوان مثال اگر سرعت اتومبیل بالاست آنگاه نیروی کمتری به پدال گاز وارد کنید.

رفتار رانندگان در شرایط طبیعی شامل سه قاعده زیراست:

۱- اگر سرعت پایین است آنگاه نیروی بیشتری به پدال گاز وارد کنید.

۲- اگر سرعت متوسط است آنگاه نیروی متعادلی به پدال گاز وارد کنید.

۳- اگر سرعت بالاست آنگاه نیروی کمتری به پدال گاز وارد کنید.

ما می‌توانیم یک سیستم فازی را بر اساس این قواعد بسازیم. از آنجا که سیستم فازی بعنوان کنترل کننده

استفاده شده. آن را کنترل کننده فازی می‌نامند. بطور خلاصه نقطه شروع ساخت یک سیستم فازی بدست

آوردن مجموعه‌ای از قواعد اگر-آنگاه فازی از دانش افراد خبره یا دانش حوزه مورد بررسی می‌باشد.

مرحله بعدی ترکیب این قواعد در یک سیستم واحد است.

انواع سیستم های فازی و مقایسه آنها

سیستم فازی روش توسعه یافته منطق بولی برای بکاربردن مفاهیم مبهم است. برای بیان ابهام در قالب یک عدد، سیستم فازی تابعی برای عضویت در یک دسته معرفی میکند، که به هر عنصر یک عدد حقیقی بین صفر و یک نسبت میدهد (صفر و یک هم شامل این اعداد میباشند). این عدد نشان دهنده درجه عضویت عنصر نسبت به مجموعه مورد نظر میباشد. عضویت صفر بیانگر این است که عنصر مورد نظر کاملاً خارج از مجموعه است. درحالیکه عدد یک نشاندهنده این است که عنصر مورد نظر کاملاً در مجموعه قرار دارد.

معمولاً از سه نوع سیستم فازی صحبت به میان می آید :

- (۱) سیستم های فازی خالص
- (۲) سیستم های فازی تاکاگی – سوگنو و کانگ (TSK)
- (۳) سیستم های با فازی ساز و غیر فازی ساز (ممدانی)

سیستم فازی خالص پایگاه قواعد فازی مجموعه ای از قواعد اگر-آنگاه فازی است. موتور استنتاج فازی این قواعد را به یک نگاشت از مجموعه های فازی در فضای ورودی به مجموعه های فازی در فضای خروجی بر اساس اصول منطق فازی ترکیب می کند. مشکل اصلی در رابطه با سیستم های فازی خالص این است که ورودی ها و خروجی های آن مجموعه های فازی می باشند. در حالی که درسیستمهای مهندسی ورودی و خروجی ها متغیرهایی با مقادیر حقیقی می باشند.

برای حل این مشکل تاکاگی _ سوگنو و کانگ نوع دیگری سیستم های فازی معرفی کرده اند که ورودی ها و خروجی های آن متغیر هایی با مقادیر واقعی هستند. بدین ترتیب قاعده فازی از یک عبارت توصیفی با مقادیر زبانی، به یک رابطه ساده تبدیل شده است.

به طور مثال اگر سرعت اتومبیل X باشد، آنگاه نیروی وارد بر پدال گاز برابر است با $Y=CX$ مقایسه نشان می دهد که بخش “آنگاه” قاعده فازی از یک عبارت توصیفی با مقادیر زبانی به یک رابطه ریاضی ساده تبدیل شده

این تغییر ترکیب قواعد فازی را ساده تر می سازد. در حقیقت سیستم فازی TSK یک میانگین وزنی از مقادیر بخش های “آنگاه” قواعد می باشد. مشکلات عمده سیستم فازی TSK عبارتند از:

- (۱) بخش “آنگاه” قاعده یک فرمول ریاضی بوده و بنابراین چهارچوبی را برای نمایش دانش بشری فراهم نمیکند.
- (۲) این سیستم دست ما را برای اعمال اصول مختلف منطق فازی باز نمی گذارد و در نتیجه انعطاف پذیری سیستم های فازی در این ساختار وجود ندارد.

برای حل این مشکل ما از نوع سومی از سیستم های فازی یعنی سیستم های فازی با فازی ساز و غیر فازی ساز استفاده میکنیم. به منظور استفاده از سیستم های فازی خالص در سیستم های مهندسی یک روش ساده اضافه کردن یک فازی ساز در ورودی که متغیر هایی با مقادیر حقیقی را به یک مجموعه فازی تبدیل کرده و یک غیر فازی ساز که یک مجموعه فازی را به یک متغیر با مقدار حقیقی در خروجی تبدیل میکند می باشد. نتیجه یک سیستم فازی با فازی ساز و غیر فازی ساز است. این سیستم فازی معایب سیستم فازی خالص و سیستم فازی TSK را می پوشاند. از این پس منظور ما از سیستم های فازی سیستم های فازی با فازی ساز و غیر فازی ساز خواهد بود.

کاربردهای سیستم فازی

در اینجا به برخی از موارد کاربرد سیستم های فازی اشاره می گردد

- (۱) جاروبرقی های فازی که با اندازه گیری میزان غبار و با توجه به جنس سطح زیر، به بهترین نحو عمل نظافت را انجام می دهند.
- (۲) ماشین های لباسشویی: چرخه شستشو را با امتحان اندازه لباس ها، مقدار پودر لباسشویی و میزان پاک کنندگی بهینه می کند.
- (۳) دیگ بخار کشتی: دما، فشار و محتویات شیمیایی را کنترل کرده و در سطح قابل اطمینانی قرار می دهد.
- (۴) دستگاه تهویه مطبوع: دستگاه طوری تنظیم می شود تا به تدریج دمای اتاق به دمای مورد نظر برسد

۵) دوربین‌های عکاسی و فیلم برداری با قابلیت تنظیم و قدرت زوم بهینه.

۶) هدایت و کنترل نرم و سریع هلی‌کوپترها به طور اتوماتیک

۷) سیستم‌های کنترل ترافیک هوشمند در خیابان‌ها.

۸) طراحی روباتی که قادر به تشخیص رنگها باشد.

۹) ساخت کنترل کننده های لوازم خانگی.

۱۰) پیش بینی زردی نوزاد.

مثالی از منطق فازی

کنترل فازی کوره سیمان (چند ورودی و چند خروجی)

سیمان بوسیله آسیاب کلینکر که ترکیبی از مواد معدنی است در یک کوره ساخته میشود . بدلیل این که عملکرد این کوره غیر خطی و متغییر با زمان میباشد وداده های نمونه برداری کمی نیز دارد کنترل آن با استفاده از روشهای کنترل متعارف کاری مشکل است. در اواخر دهه ۱۹۷۰ شرکتی در دانمارک یک سیستم فازی را برای کنترل کوره سیمان ابداع نمود. سیستم فازی ابداعی چهار ورودی و دو خروجی داشت.

ورودی های چهارگانه عبارتند اند از:

۱- درصد اکسیژن در گازهای اگزوز ۲- درجه حرارت گازهای اگزوز ۳- گشتاور آسیاب کوره

۴- وزن حجمی کلینکر

خروجی های این سیستم نیز عبارتند از:

۱- میزان زغال سنگ ریخته شده به کوره ۲- میزان جریان هوا

مجموعه ی که از قواعد اگر-آنگاه فازی رابطه خروجی ها را با ورودی ها مشخص می کند. بعنوان مثال :

۱) اگر درصد اکسیژن بالا ودرجه حرارت پایین است آنگاه درجه هوا را افزایش دهید.

(۲) اگر درصد اکسیژن بالا و درجه حرارت بالا است آنگاه میزان زغال سنگ را اندکی کاهش دهید.

بررسی روش های چهارگانه استفاده از منطق فازی

حال این سوال مطرح است که این وسایل چگونه از منطق فازی استفاده میکنند؟

فازی کردن :

در این مرحله واقعیات بر اساس سیستم فازی تعریف می شوند. ابتدا باید ورودی و خروجی سیستم معرفی شده، سپس قوانین اگر-آنگاه مناسب به کار گرفته شوند. برای ساخت تابع عضویت بایستی از داده های خام استفاده شود. حال سیستم برای اعمال منطق فازی آماده است.

مثالی برای فازی کردن :

دستگاه تهویه ای را در نظر بگیرید که با اندازه گیری دما و رطوبت اتاق میزان به جریان در آوردن هوا را مشخص می کند. در این مورد ورودی عبارتست از دما و میزان رطوبت و خروجی نیز سطح جریان هوای خروجی از دستگاه تهویه مطبوع است که شامل سه حالت ، خاموش، کم و زیاد میباشد همچنین این قوانین اگر-آنگاه استفاده می شوند:

(۱) اگر اتاق گرم باشد آنگاه هوای زیادی منتشر کن

(۲) اگر اتاق خنک باشد، آنگاه هیچ هوایی منتشر نکن

(۳) اگر اتاق سرد و مرطوب است ، آنگاه کمی هوا را به جریان بینداز.

در آخر، متخصص باید دو تابع عضویت تعیین کند. یکی برای اینکه دما را به مقدار فازی تبدیل کند و دیگری برای تبدیل میزان رطوبت به مقدار فازی.

استنتاج:

هنگامی که ورودی ها به سیستم استنتاج می‌رسند، همه قوانین اگر-آنگاه را مورد ارزیابی قرار می‌دهد و "درجه درستی" آنها را مشخص می‌کند. اگر یک ورودی داده شده به طور صریح با یک قانون اگر-آنگاه مشخص نشده باشد، آنگاه تطابق بخشی مورد استفاده قرار می‌گیرد که دارای وزن بیشتر است تا جواب مشخص شود.

مثالی برای استنتاج :

فرض کنید دستگاه تهویه مطبوع دما و درجه رطوبت را اندازه‌گیری کرده و به آنها به ترتیب مقادیر فازی ۰,۷ و ۰,۱ را نسبت داده باشد. حال این سیستم بایستی درستی هر یک از قوانین فازی را که در بالا بحث شد مورد بررسی قرار دهد. برای این منظور روشهای استنتاج بسیاری وجود دارد. این مثال ساده ترین روش را مورد استفاده قرار می‌دهد که روش ماکسیمم-مینیمم نامیده می‌شود. این روش مقدار فازی قسمت آنگاه (نتیجه) را به قسمت اگر نسبت می‌دهد. بنابراین این روش مقادیر فازی ۰,۷ و ۰,۱ و ۰,۱ را به ترتیب به قوانین ۱، ۲ و ۳ نسبت می‌دهد.

پایگاه قواعد :

در این قسمت برای بدست آوردن یک نتیجه کلی تمامی مقادیر بدست آمده از قسمت استنتاج با هم ترکیب می‌شوند. قوانین فازی مختلف نتایج مختلفی خواهند داشت. بنابراین ضروری است تا همه قوانین در نظر گرفته شوند.

مثالی برای پایگاه قواعد :

هر نتیجه استنتاجی درباره سیستم تهویه مطبوع عمل خاصی را پیشنهاد می‌کند. در مثال فوق قانون اول، سطح گردش هوای زیاد را پیشنهاد می‌کند. قانون دوم، خاموش کردن، و قانون سوم، سطح گردش هوای کم را بیان می‌کند. تکنیک های متعددی برای بدست آوردن نتیجه کلی وجود دارند. این مثال از روش ماکزیمم-مینیمم که روش ساده ای است استفاده می‌کند. این روش ماکزیمم مقدار فازی قسمت استنتاج به عنوان نتیجه در نظر

می گیرد. یعنی در عمل، قسمت ساخت مقدار ۰,۷ را انتخاب می کند چون مقدار بیشتری را بین مقادیر فازی دارا است.

غیر فازی ساز :

در این مرحله مقدار فازی بدست آمده از قسمت پایگاه قواعد به یک داده قابل استفاده تبدیل می شود. این قسمت از کار اغلب پیچیده است چون مجموعه فازی نبایستی مستقیماً به داده قابل استفاده تبدیل شود. از آنجا که کنترلرهای سیستم‌های فیزیکی به سیگنال‌های گسسته نیاز دارند، این مرحله بسیار مهم می باشد.

مثالی برای غیر فازی ساز :

به خاطر دارید که مقدار فازی بدست آمده از مرحله قبل ۰,۷ بود. این مقدار عددی برای سیستم تهویه مطبوع قابل فهم نیست. باید مشخص شود که دستگاه کدامیک از فرامین کم، زیاد یا خاموش را به جریان بیندازد. مرحله بازگرداندن از حالت فازی بایستی عدد ۰,۷ را به یکی از فرامین فوق تبدیل کند. در این مثال واضح است که مقدار خروجی ۰,۷ بیانگر این است که سیستم تهویه مطبوع بایستی در حالت زیاد باشد.

دلایل استفاده از منطق فازی

می توان دلایل استفاده گسترده از منطق فازی را در موارد ذیل خلاصه نمود:

- منطق فازی از نظر مفهومی بسیار ساده می باشد؛ زیرا مفاهیم ریاضی مورد استفاده در استدلال فازی بسیار ساده هستند.

- منطق فازی بسیار انعطاف پذیر است. در واقع به آسانی می توان یک سیستم فازی را برای حل یک مسئله جدید سازماندهی نمود و نیازی به طراحی دوباره سیستم وجود ندارد.

- منطق فازی دارای قابلیت مدل سازی توابع غیر خطی پیچیده می باشد. می توانید برای برقراری ارتباط بین هر مجموعه از داده های ورودی و خروجی از سیستم فازی استفاده کنید. این فرایند از طریق تکنیک هایی نظیر ANFIS بسیار ساده می شود.

- منطق فازی توان تحمل داده های غیر دقیق فازی را به شکل مطلوبی دارا می باشد. با نگاه دقیق به هر چیزی، متوجه نوعی نادرستی در آن می شویم و منطق فازی از این مفهوم یک پردازش سازمان یافته می سازد.

- منطق فازی بر مبنای تجربه متخصصان عمل می کند. برخلاف شبکه های عصبی که داده های آموزشی را دریافت کرده و مدل های مبهم تولید می کند، منطق فازی به شما اجازه می دهد تا بر تجربه متخصصانی که شناخت دقیقی از سیستم مورد بررسی دارند، تکیه نمایید.

- منطق فازی با تکنیک های کنترلی مرسوم قابل ادغام است. سیستم های فازی الزاما جایگزینی برای روش های کنترلی نیستند؛ بلکه در بسیاری از موارد سیستم های فازی پیاده سازی سیستم های کنترلی را تکمیل و تسهیل می کنند.

- منطق فازی بر مبنای زبان طبیعی می باشد. اصول منطق فازی بر مبنای نوع ارتباط بشر است. از آنجا که منطق فازی بر پایه ساختارهای توصیف کیفی در زبان روزمره استوار است، استفاده از آن بسیار ساده می باشد. مورد آخر شاید مهمترین مورد بوده و سزاوار بحث بیشتری است. زیرا زبان طبیعی که توسط مردم در امور روزمره مورد استفاده قرار می گیرد بر مبنای تاریخ هزاران ساله زندگی بشر شکل گرفته و بهینه سازی شده است.

کمبود ها و نواقص سیستم های فازی

منطق فازی و منطق بولی هر دو بر پایه واقعیات می باشند. با این تفاوت که منطق فازی توانایی کارکردن با داده های مبهم را نیز داراست. با این وجود منطق فازی هنوز قادر به حل بعضی مسائل نیست .

بسیاری از سیستم ها، مانند آنچه در بحث کاربرد گفته شد می توانند از منطق فازی بدون هیچ مشکلی استفاده کنند. چون نیاز به هیچ تصمیم گیری درونی و فکری ندارند. اما بعضی سیستم ها به منطق پیچیده تری نیاز دارند تا بتوانند به بیان گمان، تعقل، شک و ... بپردازند.

برای نشان دادن نقصان منطق فازی می توان به سیستم شناخت دوست یا دشمن اشاره کرد

این سیستم برای شناخت هواپیماهای ارتشی یا مسافربری دوست و دشمن به کار می رود. در حالت عادی سیستم از هر هواپیما یک سیگنال شناسایی دریافت می کند. دلایل زیادی وجود دارند که ممکن است سیستم این سیگنال را دریافت نکند مانند: بد عمل کردن سیستم، بد عمل کردن فرستنده، نبودن این سیستم روی هواپیماها، پارازیت سیگنال یا خاموش بودن رادیو. در این موارد، این سیستم باید از منطق برای شناسایی هواپیماها استفاده کند. که فقط مربوط به داده های از قبیل مسیر پرواز نیست. چون این داده ها هواپیماهای دوست را از دشمن تشخیص نمی دهند. بنابراین منطق فازی در سیستم تشخیص دوست از دشمن کاربردی ندارد! برای این منظور باید از منطقی استفاده شود که توانایی تصمیم گیری درونی را داشته باشد.

چه زمانی نباید از منطق فازی استفاده نمود؟

منطق فازی همیشه یک راه حل مناسب نیست؛ اما چه زمانی نباید از آن استفاده کرد؟ همانطور که میدانید منطق فازی یک روش مناسب و آسان برای نگاشت فضای ورودی به فضای خروجی است. حال اگر در یک مساله ای متوجه شدید که استفاده از منطق فازی چندان ساده نیست، بهتر است به دنبال راه حل دیگری بروید. بسیاری از کنترل کننده ها، بدون کمک منطق فازی به خوبی عمل می کنند.

جعبه ابزار منطق فازی در MATLAB

جعبه ابزار منطق فازی متشکل از مجموعه ای از توابع فازی در قالب محیط محاسباتی MATLAB می باشد. این جعبه ابزار امکاناتی را برای استنتاج سیستم های فازی در چارچوب MATLAB فراهم می آورد. همچنین در راستای شبیه سازی سیستم های فازی، امکان یک پارچه سازی آنها در قالب نرم افزار Simulink وجود دارد.

می‌توانید از توابع مربوط به این جعبه ابزار در برنامه‌های دیگر (مثلاً به زبان C) استفاده نمایید. این جعبه ابزار از یک محیط گرافیکی برای کمک به کاربر بهره می‌گیرد، اما در صورت تمایل می‌توانید از توابع مربوط به آن روی خط فرمان MATLAB استفاده نمایید.

این جعبه ابزار دارای سه دسته متفاوت از امکانات می‌باشد:

- توابع خط فرمان،

- توابع محاوره‌ای گرافیکی،

- بلوک‌های Simulink،

دسته‌ی اول از توابعی تشکیل می‌شود که می‌توانید آنها را روی خط فرمان ویا از برنامه کاربردی خود فراخوانی نمایید. بسیاری از این توابع در قالب M-File های MATLAB پیاده سازی شده‌اند. در واقع الگوریتم های مربوط به منطق فازی به وسیله‌ی دستورات MATLAB در قالب این فایل‌ها پیاده سازی شده‌اند. برای مشاهده کد مربوط به این توابع، از دستور زیر روی خط فرمان استفاده نمایید.

Type function_name

می‌توانید با اعمال تغییر در هریک از توابع مربوط به جعبه ابزار و ذخیره سازی مجدد آن با یک نام جدید، توابع مورد نظر خود را ساخته و قابلیت‌های این جعبه ابزار را گسترش دهید.

در دسته دوم، جعبه ابزار مجموعه‌ای از ابزارهای محاوره‌ای را برای دسترسی کاربر از طریق محیط گرافیکی به ابزارهای مختلف فراهم می‌آورد. این ابزارهای گرافیکی محیطی را برای طراحی، تحلیل و پیاده سازی سیستم های فازی فراهم می‌آورند.

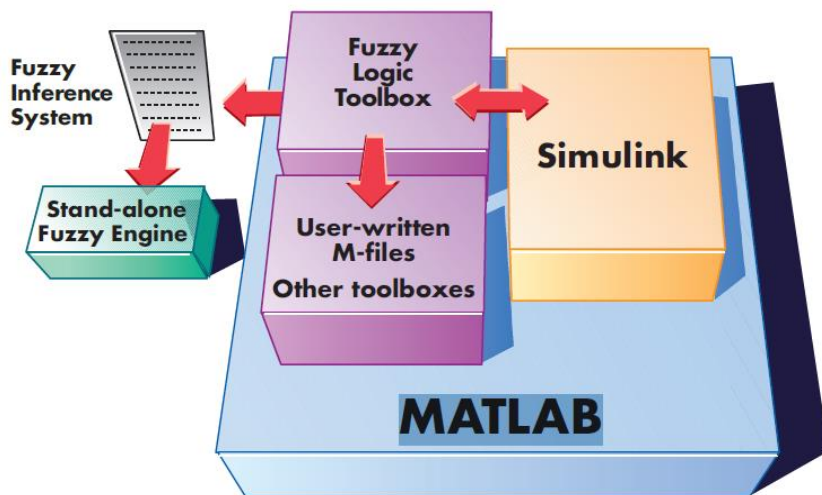
دسته سوم امکانات این جعبه ابزار، از مجموعه‌ای از بلوک‌های Simulink تشکیل شده‌اند که برای شبیه سازی سریع سیستم‌های فازی در قالب محیط Simulink به کار می‌روند.

با توجه به اینکه غالب استدلال های بشر با قواعد فازی مرتبط هستند، اهمیت و کاربرد این جعبه ابزار روز به روز در حال گسترش می باشد. این جعبه ابزار با فراهم آوردن یک چارچوب نظام مند برای محاسبات مربوط به قواعد فازی، به شکل مطلوبی قدرت استدلال کاربر را افزایش می دهد.

نحوه عملکرد جعبه ابزار منطق فازی

به کمک جعبه ابزار منطق فازی MATLAB، می توان سیستم های استنتاج فازی را ایجاد و ویرایش کرد. می توانید این کار را از طریق ابزارهای گرافیکی و توابع خط فرمان انجام دهید. همچنین امکان ایجاد این سیستم ها به صورت خودکار از طریق تکنیک های کلاسترینگ و انطباقی neuro-fuzzy وجود دارد. همچنین در صورتی که به نرم افزار Simulink دسترسی داشته باشید، می توانید سیستم فازی خود را از طریق بلوک های Simulink شبیه سازی کنید.

همچنین این جعبه ابزار، قابلیت اجرای برنامه های C را به صورت مستقیم به شما می دهد این عملیات به کمک موتور استنتاج فازی ممکن می شود. می توانید این موتور را برای کاربرد مورد نظر خود سفارشی کنید.



شکل ۱-۷: جعبه ابزار منطق فازی در MATLAB

به دلیل یکپارچه بودن محیط MATLAB می‌توانید ابزارهای مورد نظر خود را برای پیاده سازی یک سیستم فازی به کمک جعبه ابزارهای مختلف مثل سیستم‌های کنترلی، شبکه‌های عصبی و یا جعبه ابزار بهینه سازی فراهم آوری کرد.

کنترل کننده‌های عصبی

کنترل کننده‌های عصبی، کنترل کننده‌های خودکاری هستند که در آن‌ها از شبکه عصبی جهت محاسبه و پردازش سیگنال کنترلی استفاده می‌شود. عملکرد شبکه عصبی بر پایه یادگیری بوسیله مثال می‌باشد. شبکه‌های عصبی الگوی ساده شده شبکه عصبی مغز هستند. هر شبکه عصبی از تعداد زیادی واحدهای مجزای محاسباتی خطی و یا غیر خطی بنام «نرون» تشکیل می‌شود. هر کدام از این واحدهای محاسباتی، پارامترهایی دارند که اصطلاحاً به آن‌ها وزن «Weight» گفته می‌شود. با تنظیم این وزن‌ها، اصطلاحاً شبکه آموزش می‌بیند؛ یعنی می‌توان توابع غیر خطی مختلفی را بین ورودی‌ها و خروجی‌های شبکه ایجاد نمود. در شبکه‌های عصبی، هدف نهایی انتخاب وزن‌ها است بطوریکه رابطه مورد نظر بین ورودی و خروجی برقرار گردد.

شبکه‌های عصبی از اطلاعات عددی که توسط حسگرهای ورودی و خروجی بدست می‌آید، برای تنظیم وزن‌های خود (یادگیری)، استفاده می‌کنند. به بیان دیگر یک شبکه عصبی از تجارب عینی جهت یادگیری استفاده می‌کند تا چنانچه بعداً مواردی شبیه این مورد پیش بیاید، با یادآوری تجارب قبلی بتواند خروجی مناسب را تولید نماید. از دیدگاه تئوری سیستم‌ها، شبکه‌های عصبی مصنوعی قابلیت نگاشت غیر خطی از یک فضای با بعد متناهی به فضای دیگر را دارا می‌باشند. چنین شبکه‌های در حالت ایده‌آل قادرند بر هر سه نوع مشکل کنترل سیستم‌های پیچیده (یعنی پیچیدگی محاسباتی، غیر خطی بودن و عدم قطعیت) غلبه کنند. شبکه‌های عصبی قادر به انجام هرگونه نگاشت غیر خطی می‌باشند. روش‌هایی که جهت تنظیم پارامترهای شبکه عصبی بکار می‌روند همگی متکی به داده‌های ورودی و خروجی شبکه می‌باشند و به شرط اعمال داده‌های متناسب با همه‌ی جنبه‌های ناشناخته سیستم، عدم قطعیت سیستم از بین می‌رود. نهایتاً به دلیل اینکه نرون‌ها می‌توانند بصورت

موازی با هم کار نمایند، پس می‌توان با استفاده از توازی، پیچیدگی محاسباتی را کم نموده و بر سرعت پردازش اضافه نمود.

با تفاسیر فوق معلوم می‌شود که بخش مهمی از طراحی یک کنترل کننده عصبی، تعیین چگونگی جمع آوری داده‌های آموزشی و استفاده از آن، جهت تنظیم پارامترهای آزاد شبکه عصبی می‌باشد. این مکانیزم به کنترل کننده‌های وفقی که در آن پارامترهای آزاد کنترل کننده در طول زمان تصحیح می‌شوند، شباهت زیادی دارد. بنابر این کنترل گره‌های عصبی، ذاتا وفقی می‌باشند؛ یعنی ساختار آن‌ها طوری طراحی شده که قادر به پذیرش اطلاعات جدید در جهت بهبود عملکرد خود می‌باشند.

با توجه به قابلیت شبکه‌های عصبی در کنترل سیستم‌های پیچیده، می‌توان از آن جهت کنترل ربات نیز استفاده نمود. تاکنون روش‌های متعددی توسط محققین، جهت پیاده سازی کنترل گره‌های عصبی پیشنهاد شده است. اکثر این روش‌ها به دنبال این هستند که شبکه عصبی، داینامیک ربات را فرا گیرد، حال به طور کامل یا تقریبی. همین هدف در کنترل کننده‌های فازی-عصبی به عنوان زیر مجموعه‌ای از کنترل گره‌های عصبی دنبال می‌شود. بنابراین شناخت روش‌های مختلف پیاده سازی و آموزش کنترل گره‌های عصبی و بررسی نقاط ضعف و قوت آن‌ها می‌تواند کمک بزرگی در جهت بهبود طراحی کنترل گره‌های عصبی یا فازی-عصبی باشد.

کنترل کننده‌های فازی-عصبی

در سال‌های اخیر تلفیق دو روش فازی و عصبی به عنوان یکی از روش‌های مطرح در کنترل سیستم‌های مختلف مطرح گردیده است. این روش اولین بار توسط «لی» در سال ۱۹۷۰ مطرح گردید. در این روش، تجربه شخص خبره و داده‌های واقعی بدست آمده از سیستم با هم تلفیق می‌شود و در حقیقت تجارب شخص خبره با استفاده از داده‌های واقعی تصحیح می‌شود.

استفاده از این روش جهت آموزش شبکه‌های عصبی، این حقیقت را روشن ساخت که شبکه‌های فازی-عصبی زیر مجموعه‌ای از شبکه‌های عصبی می‌باشند که نرون‌های آن‌ها قوانین فازی را پیاده سازی می‌نمایند. در حقیقت می‌توان به کمک این روش تجربیات شخص خبره را درون نرون‌های شبکه عصبی جاسازی نمود و به کمک مشاهدات عینی (داده‌های واقعی) به اصلاح این تجربیات پرداخت. در حقیقت شبکه فازی-عصبی، خصلت خوانایی و شفافیت قوانین فازی و دقت شبکه‌های عصبی را با هم یکجا گرد آورده است.

امروزه کاربرد سیستم‌های فازی-عصبی در کارهایی مانند کنترل سیستم‌های غیرخطی، شناسایی الگو، پزشکی، سیستم‌های خبره و ... مطرح گردیده است. ربات نیز به عنوان یک سیستم غیر خطی می‌تواند توسط کنترل کننده فازی-عصبی کنترل شود. به دلیل سریعتر بودن این کنترلگر نسبت به کنترل کننده عصبی و دقیقتر بودن آن نسبت به کنترل کننده‌های فازی، می‌توان از آن در کنترل بلادرنگ ربات استفاده نمود.

جهت پیاده سازی یک کنترلگر فازی توسط شبکه عصبی، یک شبکه چهار لایه با مشخصات زیر مورد نیاز است:

۱. لایه ورودی (Input Layer)

۲. لایه فازی کننده (Fuzzification Layer)

۳. لایه تصمیم گیری (Decision Layer)

۴. لایه خروجی (Output Layer)

در طراحی کنترل کننده فازی-عصبی می‌توان از دو ساختار استنتاج، یکی «ممدانی» و دیگری فرم «سوگینو» استفاده کرد. در ساختار استنتاج ممدانی خروجی هر قانون به صورت فازی بیان شده ولی در ساختار استنتاج به فرم سوگینو، خروجی هر قانون ترکیب خطی از ورودی‌های آن قانون و به عبارت دیگر بصورت غیر فازی (Crisp) می‌باشد.

مزیت استفاده از قوانین سوگینو این است که: خروجی هر قانون به صورت Crisp بوده، بنابراین با اندک محاسبه‌ای می‌توان خروجی کل قوانین را بدست آورد. پس عملیات محاسباتی ساده‌تر و سریعتر از فرم ممدانی انجام می‌شود.

فصل دوم

شرح پروژه

مقدمه طراحی سیستم کنترل شستشوی فازی

این پروژه در حقیقت از یک قسمت کلی، برنامه کامپیوتری (در محیط برنامه نویسی MATLAB) تشکیل شده است. در قسمت برنامه کامپیوتری یک برنامه گرافیکی توسط نرم افزار MATLAB نوشته شده است. که در آن پروتکل ها و استانداردها پیاده سازی شده است.

کنترل کننده های فازی

در این پروژه که قسمتی از آن کنترل زمان و میزان پودر شستشو می باشد. ورودی ها که شامل خطا و مشتق خطا می باشند (میزان کثیفی و وزن البسه) وارد سیستم استنتاج فازی می شوند و پس از اعمال قوانین برروی ورودی ها، خروجی ها غیر فازی می شوند و به ماشین شستشو فازی اعمال می گردند. قوانین بدست آمده در این پروژه با توجه به ساختار یک ماشین شستشو ۷ کیلو گرمی خانگی بدست می آید که در ادامه به طور مفصل در مورد آن توضیح داده می شود. ساختار کنترل فازی، مقاومت قابل توجهی را نسبت به نویز موجود در ماشین شستشوی فازی ارائه می نماید. که می توان به سهولت از منطق فازی استفاده نمود و کنترلرهای مناسبی را طراحی نمود.

طراحی سیستم فازی

به طور کلی در طراحی یک سیستم فازی ۴ مرحله باید مد نظر طراح باشد.

۱- جمع آوری داده ها

۲- تعیین ورودی و خروجی های کنترل کننده

۳- اختصاص توابع عضویت به هریک از ورودی ها و خروجی ها

۴- تشکیل قوانین فازی

در ادامه در مورد هر کدام به طور مفصل توضیح داده می‌شود و به طور کلی با اصول اولیه‌ی طراحی یک سیستم فازی آشنا خواهید شد.

قدم اول جمع آوری داده‌ها

قدم اول در طراحی یک سیستم فازی جمع آوری داده‌ها می‌باشد؛ (طراح باید بداند که ماشین شستشوی مورد بحث چند کیلو گرمی است همینطور باید بداند که میزان زمان شستشو و پودر لازم برای لباس‌های روغنی و سایر کثیفی‌های دیگر چقدر است) و سپس با استفاده از این اطلاعات است که طراح می‌تواند بازه و توابع عضویت، هر کدام از ورودی‌ها و خروجی‌ها را به آنها اختصاص دهد و سیستم را طراحی کند.

ساختار کلی ماشین شستشوی مورد بحث!

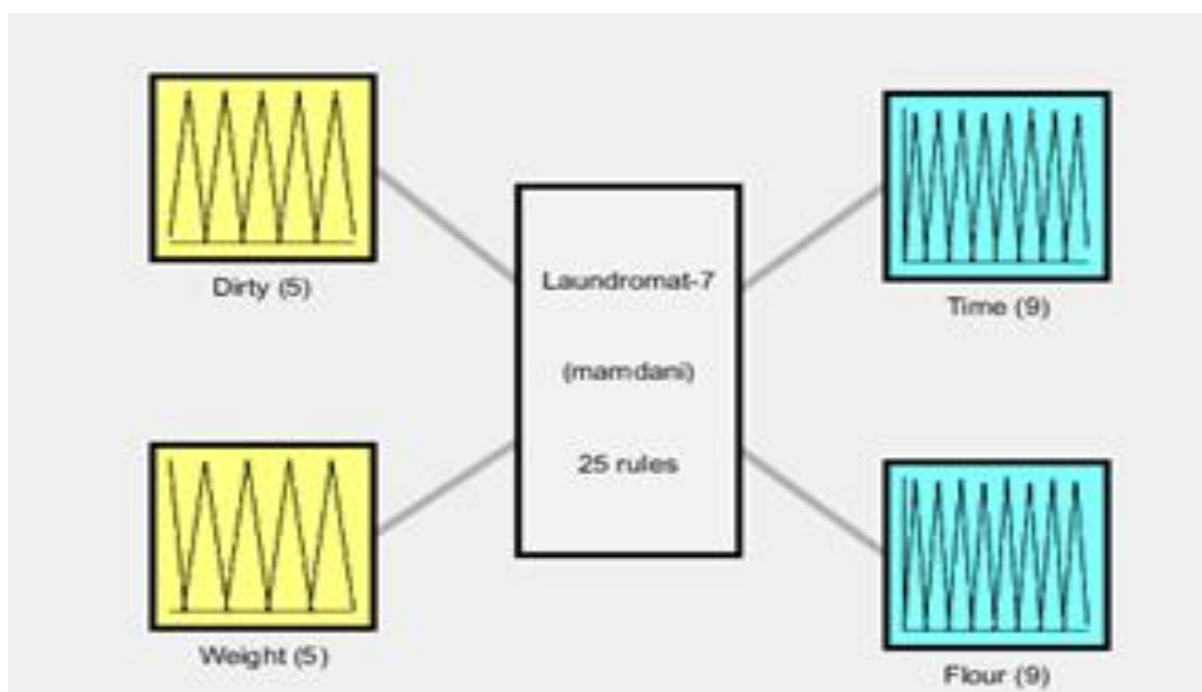
ساختار کلی ماشین شستشوی مورد استفاده یک سیستم دو ورودی دو خروجی است که دو ورودی فوق مقدار اندازه گیری شده کثیفی و حجم لباس‌ها بوده و خروجی زمان و میزان پودر مناسب شستشو میباشد. بعنوان ورودی (سنسورهایی) در این سیستم تعبیه شده این سنسورها که از نوع نوری می‌باشند میزان نوری را که از طرف مقابل ساطع شده و از آب عبور کرده را اندازه گیری می نمایند. سنسور نوری همچنین میتواند معین کند که نوع کثیفی چیست لباس گل آلود است یا چرب؟ گل در آب سریعتر حل می‌شود بنابراین اگر نور دریافتی بسرعت کاهش پیدا کند در آن صورت لباس گل آلود است در حالی که اگر لباس روغنی باشد کندتر در آب حل شده و کاهش نور دریافتی کندتر خواهد بود. ماشین همچنین دارای یک سنسور بار ۷ کیلو گرمی می باشد که حجم لباس‌ها را ثبت می کند واضح است که تعداد لباس‌های بیشتر زمان و پودر بیشتری برای شستشو لازم دارد. موارد فوق را می توان در تعدادی قاعده اگر- آنگاه فازی برای ساخت یک سیستم فازی خلاصه کرد.

پس با توجه به ساختار یک ماشین شستشو ۷ کیلو گرمی خانگی خواهیم داشت!

- ۱- سنسور نوری مورد استفاده در این پروژه قابلیت این را دارد که از صفر تا صد درصد میزان چرک و کثیفی لباس‌ها را اندازه گیری کند.
- ۲- حداکثر وزنی که سنسور بار می‌تواند تحمل کند ۷ کیلو گرم است و این به معنای این نیست که میتوان ۷ کیلو گرم لباس خالص در محفظه ماشین لباسشویی ریخت، بلکه ۷ کیلو گرم برابر است با وزنی که سنسور بار بعد از خیس شدن لباس‌ها میتواند اندازه‌گیری کند.
- ۳- مقدار زمان و پودر شستشو به مقدار لباس و میزان کثیفی بستگی دارد.
- ۴- حداکثر زمانی که برای شستشو لباس‌ها با حداکثر کثیفی و حداکثر وزن در نظر گرفته ایم برابر ۹۰ دقیقه است.
- ۵- حداکثر پودری که برای شستشو لباس‌ها با توجه به میزان پودر برای پیش شستشو و میزان پودر برای شستشو اصلی در نظر گرفته‌ایم برابر ۴۰۰ گرم است.
- ۶- سخت‌ترین لکه‌ها عبارت‌اند از؛ (لکه جوهر، روغن، گریس، خون، شکلات، قهوه و لک انواع سس‌ها)

قدم دوم تعیین ورودی و خروجی‌های کنترل کننده

همان طور که متوجه شده‌اید سیستم مورد نظردو ورودی تحت عنوان‌های میزان کثیفی البسه و وزن البسه و همینطور دو خروجی تحت عنوان‌های زمان شستشو و مقدار پودر دارد.



شکل ۱-۲: ورودی و خروجی کنترل کننده فازی

قدم سوم اختصاص توابع عضویت به هر یک از ورودی‌ها و خروجی‌ها

در این مرحله بایستی با توجه به اطلاعاتی که از قدم اول بدست آوردیم، بازه و توابع عضویت ورودی‌ها و خروجی‌ها را بدست آوریم.

ورودی‌های سامانه فازی

اولین ورودی این سامانه فازی که Dirty (میزان کثیفی البسه) نام گذاری شده است، برابر است با میزان کثیفی لباس‌ها، در داخل ماشین شستشوی فازی که میزان آن توسط سنسورنوری در ابتدای شستشو مورد اندازه گیری قرار

می‌گیرد. این ورودی شامل ۵ تابع تبدیل از نوع مثلثی با نام‌های D1,D2,D3,D4,D5 و بازه تغییرات این ورودی از (۰) درصد تا (۱۰۰) درصد باتوجه به اطلاعاتی که در قدم اول بدست آوردیم می‌باشد.

مفهوم این نام‌ها به صورت زیر است.

D1: زمانی که سنسور نوری میزان لکه‌های لباس‌ها را بسیار کم تشخیص داده است (از صفر تا بیست درصد)، لکه‌هایی چون گل‌ولای و همینطور لک عرق بدن در این بازه قرار می‌گیرند. پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه بسیار کم در نظر گرفته شود.

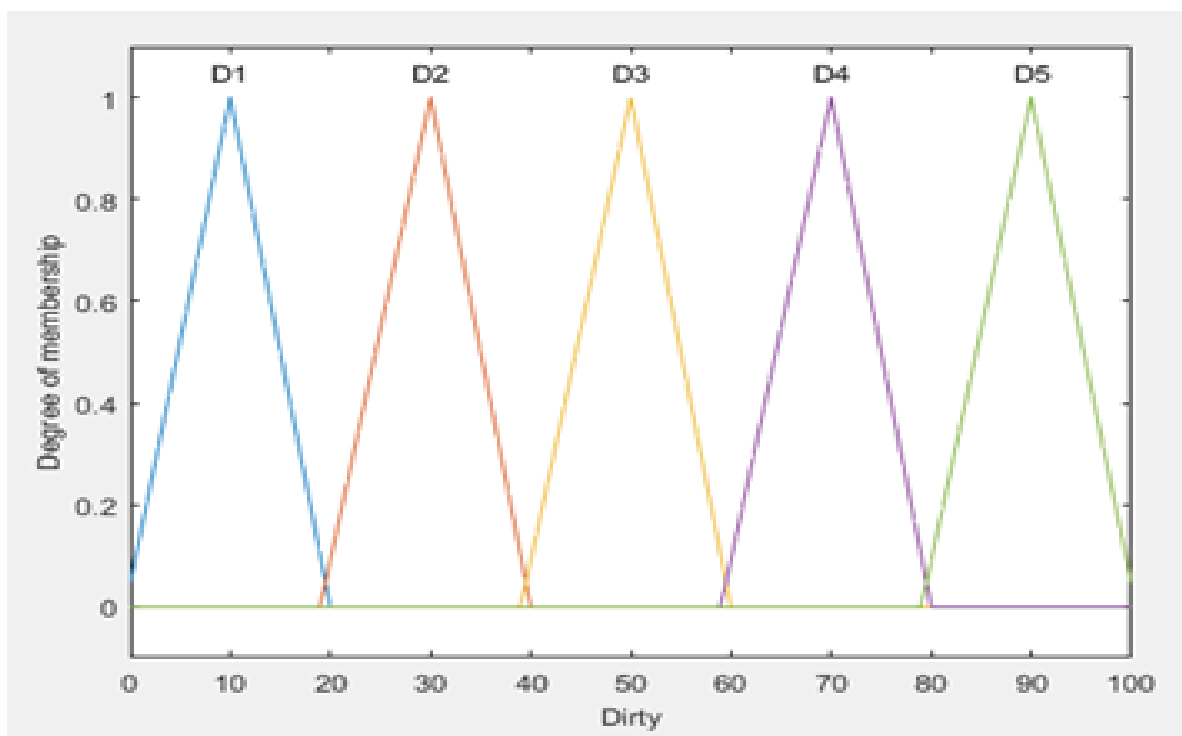
D2: زمانی که سنسور نوری میزان لکه‌های لباس‌ها را کم تشخیص داده است (از بیست تا چهل درصد)، همانند لک آرمیوه که در این بازه قرار می‌گیرد. پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه کم در نظر گرفته شود.

D3: زمانی که سنسور نوری میزان لکه‌های لباس‌ها را زیاد تشخیص داده است (از چهل تا شصت درصد)، لکه‌هایی چون چمن و سبزه و همینطور لک قهوه در این بازه قرار می‌گیرند. پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه زیاد در نظر گرفته شود.

D4: زمانی که سنسور نوری میزان لکه‌های لباس‌ها را بسیار زیاد تشخیص داده است (از شصت تا هشتاد درصد)، لکه‌هایی چون خون و شکلات و همینطور لک انواع سس‌ها در این بازه قرار می‌گیرند. پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه بسیار زیاد در نظر گرفته شود.

D5: زمانی که سنسور نوری میزان لکه‌های لباس‌ها را بسیار بسیار زیاد تشخیص داده است (از هشتاد تا صد درصد)، لکه‌هایی چون جوهر و رژلب و همینطور لک روغن و گریس در این بازه قرار می‌گیرند. پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه بسیار بسیار زیاد در نظر گرفته شود.

شکل زیر وضعیت توابع تعلق این ورودی را نشان می‌دهد.



شکل ۲-۲: وضعیت توابع تعلق ورودی Dirty

دومین ورودی این سامانه فازی که Weight (میزان وزن البسه) نام گذاری شده است، برابر است با میزان وزن لباس‌ها، در داخل ماشین شستشوی فازی که میزان آن توسط لودسل (سنسور بار) در ابتدای شستشو مورد اندازه گیری قرار می گیرد. این ورودی شامل ۵ تابع تبدیل از نوع مثلثی با نام‌های W_0, W_1, W_2, W_3, W_4 و بازه تغییرات این ورودی از (۰) کیلو تا (۷) کیلو باتوجه به اطلاعاتی که در قدم اول بدست آوردیم می باشد.

مفهوم این نام‌ها به صورت زیر است.

W_0 : زمانی که لودسل میزان وزن لباس‌ها را بسیار کم تشخیص داده است (زیر نیم کیلوگرم)، پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه بسیار کم (Error) در نظر گرفته شود. زیرا در این حالت از لحاظ اقتصادی و همینطور مصرف انرژی اصلا به صرفه نیست که ماشین شستشوی فازی روشن شود حتی اگر میزان کیفی البسه صد درصد باشد.

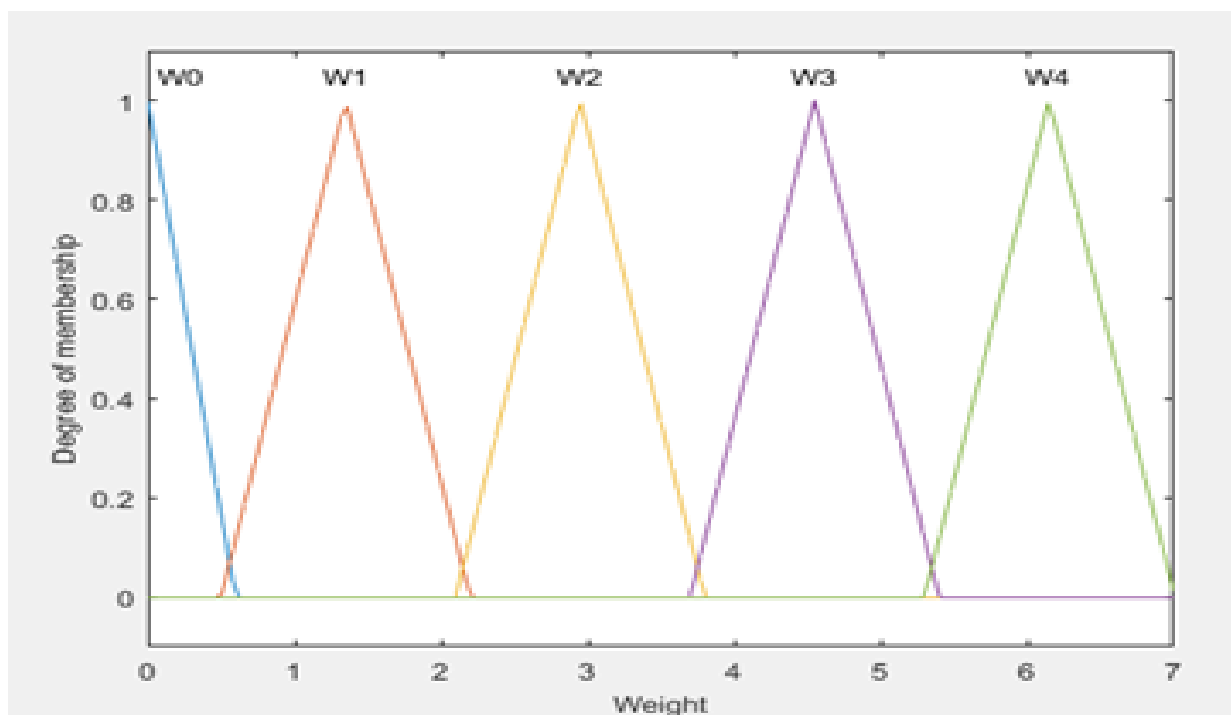
W1: زمانی که لودسل میزان وزن لباس‌ها را کم تشخیص داده است (۱,۵ کیلوگرم)، پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه کم در نظر گرفته شود.

W2: زمانی که لودسل میزان وزن لباس‌ها را زیاد تشخیص داده است (۳ کیلوگرم)، پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه زیاد در نظر گرفته شود.

W3: زمانی که لودسل میزان وزن لباس‌ها را بسیار زیاد تشخیص داده است (۵ کیلوگرم)، پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه بسیار زیاد در نظر گرفته شود.

W4: زمانی که لودسل میزان وزن لباس‌ها را بسیار بسیار زیاد تشخیص داده است (۷ کیلوگرم)، پس در این حالت لازم است میزان زمان و پودر برای شستشوی البسه بسیار بسیار زیاد در نظر گرفته شود.

شکل زیر وضعیت توابع تعلق این ورودی را نشان می‌دهد.



شکل ۲-۳: وضعیت توابع تعلق ورودی Weight

خروجی‌های سامانه فازی

اولین خروجی این سامانه فازی Time (زمان شستشو البسه) نام‌گذاری شده است و دارای ۹ تابع تبدیل از نوع مثلثی با نام‌های Error, T1, T2, T3, T4, T5, T6, T7, T8 می‌باشد و بازه‌ی تغییرات این خروجی از صفر دقیقه تا ۹۰ دقیقه در نظر گرفته شده است. مفهوم این نام‌ها به صورت زیر است

Error: زمانی که وزن لباسها زیر نیم کیلو گرم است میزان زمان برابر صفر در نظر گرفته شده است (در این حالت ماشین شستشوی فازی روشن نمی‌شود) و اصطلاحا Error می‌دهد.

T1: زمان شستشو بسیار کم

T2: زمان شستشو بسیار کم

T3: زمان شستشو کم

T4: زمان شستشو متوسط

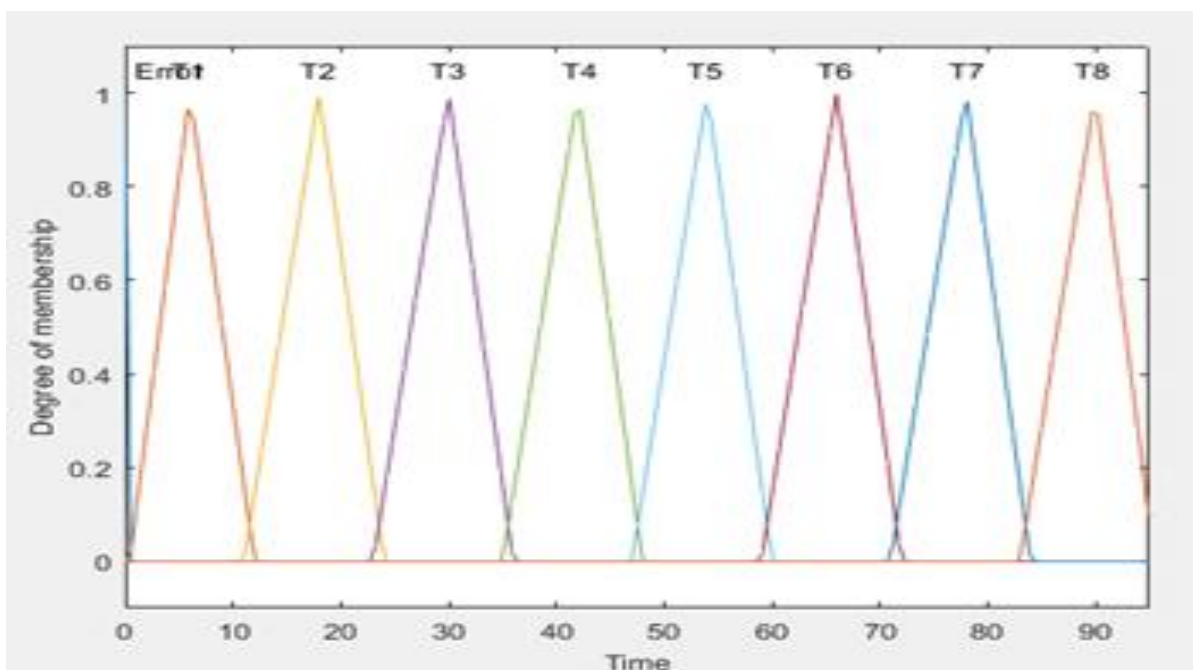
T5: زمان شستشو زیاد

T6: زمان شستشو بسیار زیاد

T7: زمان شستشو بسیار بسیار زیاد

T8: زمان شستشو بسیار بسیار بسیار زیاد

شکل زیر وضعیت توابع تعلق این خروجی (Time) را نشان می‌دهد.



شکل ۲-۴: وضعیت توابع تعلق خروجی Time

دومین خروجی این سامانه فازی Flour (میزان پودر شستشو) نام گذاری شده است و دارای ۹ تابع تعلق از نوع مثلثی با نام های Error, F1, F2, F3, F4, F5, F6, F7, F8 می باشد و بازه ی تغییرات این خروجی از (۰) گرم تا (۴۰۰) گرم در نظر گرفته شده است. مفهوم این نام ها به صورت زیر است.

Error: زمانی که وزن لباسها زیر نیم کیلو گرم است میزان پودر برابر صفر در نظر گرفته شده است (در این حالت ماشین شستشوی فازی روشن نمی شود) و اصطلاحا Error می دهد.

F1: میزان پودر بسیار کم

F2: میزان پودر بسیار کم

F3: میزان پودر کم

F4: میزان پودر متوسط

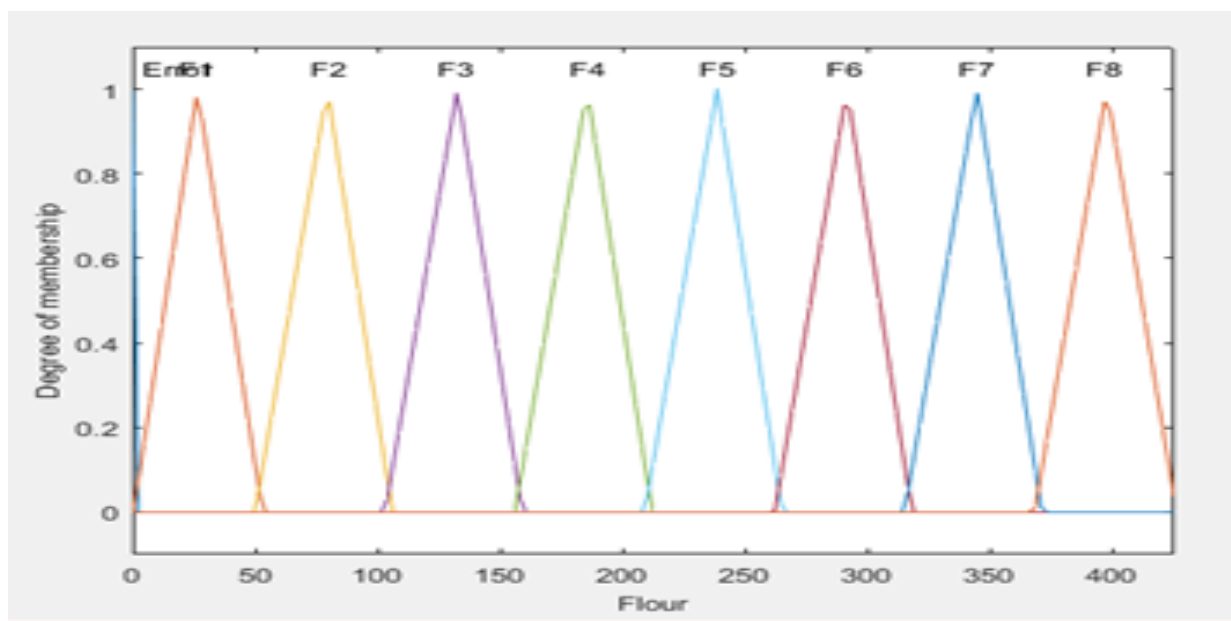
F5: میزان پودر زیاد

F6: میزان پودر بسیار زیاد

F7: میزان پودر بسیار بسیار زیاد

F8: میزان پودر بسیار بسیار زیاد

شکل زیر وضعیت توابع تعلق این خروجی (Flour) را نشان می‌دهد.



شکل ۲-۵: وضعیت توابع تعلق Flour

قدم چهارم تشکیل قوانین فازی

برای این پروژه با توجه به توابع تبدیل ورودی ۲۵ قانون میتوان نوشت. و این عدد برابر است با حاصلضرب توابع تبدیل ورودی ها (همانطور که قبلا نیز گفته شد ورودی اول ۵ تابع تبدیل و ورودی دوم نیز ۵ تابع تبدیل دارد).

قوانین فازی که در این پروژه از آن استفاده شده است به صورت زیر می‌باشد.

۱- اگر میزان چرک لباس‌ها برابر D1 و وزن لباس‌ها برابر W0 باشد. آنگاه زمان شستشو باید برابر Error و میزان پودر باید برابر Error باشد.

۲- اگر میزان چرک لباس‌ها برابر $D1$ و وزن لباس‌ها برابر $W1$ باشد. آنگاه زمان شستشو باید برابر $Z1$ و میزان پودر باید برابر $F1$ باشد.

۳- اگر میزان چرک لباس‌ها برابر $D1$ و وزن لباس‌ها برابر $W2$ باشد. آنگاه زمان شستشو باید برابر $Z2$ و میزان پودر باید برابر $F2$ باشد.

۴- اگر میزان چرک لباس‌ها برابر $D1$ و وزن لباس‌ها برابر $W3$ باشد. آنگاه زمان شستشو باید برابر $Z3$ و میزان پودر باید برابر $F3$ باشد.

۵- اگر میزان چرک لباس‌ها برابر $D1$ و وزن لباس‌ها برابر $W4$ باشد. آنگاه زمان شستشو باید برابر $Z4$ و میزان پودر باید برابر $F4$ باشد.

۶- اگر میزان چرک لباس‌ها برابر $D2$ و وزن لباس‌ها برابر $W0$ باشد. آنگاه زمان شستشو باید برابر $Error$ و میزان پودر باید برابر $Error$ باشد.

۷- اگر میزان چرک لباس‌ها برابر $D2$ و وزن لباس‌ها برابر $W1$ باشد. آنگاه زمان شستشو باید برابر $Z2$ و میزان پودر باید برابر $F2$ باشد.

۸- اگر میزان چرک لباس‌ها برابر $D2$ و وزن لباس‌ها برابر $W2$ باشد. آنگاه زمان شستشو باید برابر $Z3$ و میزان پودر باید برابر $F3$ باشد.

۹- اگر میزان چرک لباس‌ها برابر $D2$ و وزن لباس‌ها برابر $W3$ باشد. آنگاه زمان شستشو باید برابر $Z4$ و میزان پودر باید برابر $F4$ باشد.

۱۰- اگر میزان چرک لباس‌ها برابر $D2$ و وزن لباس‌ها برابر $W4$ باشد. آنگاه زمان شستشو باید برابر $Z5$ و میزان پودر باید برابر $F5$ باشد.

۱۱- اگر میزان چرک لباس‌ها برابر $D3$ و وزن لباس‌ها برابر $W0$ باشد. آنگاه زمان شستشو باید برابر $Error$ و میزان پودر باید برابر $Error$ باشد.

۱۲- اگر میزان چرک لباس‌ها برابر $D3$ و وزن لباس‌ها برابر $W1$ باشد. آنگاه زمان شستشو باید برابر $Z3$ و میزان پودر باید برابر $F3$ باشد.

۱۳- اگر میزان چرک لباس‌ها برابر $D3$ و وزن لباس‌ها برابر $W2$ باشد. آنگاه زمان شستشو باید برابر $Z4$ و میزان پودر باید برابر $F4$ باشد.

۱۴- اگر میزان چرک لباس‌ها برابر $D3$ و وزن لباس‌ها برابر $W3$ باشد. آنگاه زمان شستشو باید برابر $Z5$ و میزان پودر باید برابر $F5$ باشد.

۱۵- اگر میزان چرک لباس‌ها برابر $D3$ و وزن لباس‌ها برابر $W4$ باشد. آنگاه زمان شستشو باید برابر $Z5$ و میزان پودر باید برابر $F5$ باشد.

۱۶- اگر میزان چرک لباس‌ها برابر $D4$ و وزن لباس‌ها برابر $W0$ باشد. آنگاه زمان شستشو باید برابر $Error$ و میزان پودر باید برابر $Error$ باشد.

۱۷- اگر میزان چرک لباس‌ها برابر $D4$ و وزن لباس‌ها برابر $W1$ باشد. آنگاه زمان شستشو باید برابر $Z4$ و میزان پودر باید برابر $F4$ باشد.

۱۸- اگر میزان چرک لباس‌ها برابر $D4$ و وزن لباس‌ها برابر $W2$ باشد. آنگاه زمان شستشو باید برابر $Z5$ و میزان پودر باید برابر $F5$ باشد.

۱۹- اگر میزان چرک لباس‌ها برابر $D4$ و وزن لباس‌ها برابر $W3$ باشد. آنگاه زمان شستشو باید برابر $Z6$ و میزان پودر باید برابر $F6$ باشد.

۲۰- اگر میزان چرک لباس‌ها برابر $D4$ و وزن لباس‌ها برابر $W4$ باشد. آنگاه زمان شستشو باید برابر $Z7$ و میزان پودر باید برابر $F7$ باشد.

۲۱- اگر میزان چرک لباس‌ها برابر $D5$ و وزن لباس‌ها برابر $W0$ باشد. آنگاه زمان شستشو باید برابر $Error$ و میزان پودر باید برابر $Error$ باشد.

۲۲- اگر میزان چرک لباس‌ها برابر $D5$ و وزن لباس‌ها برابر $W1$ باشد. آنگاه زمان شستشو باید برابر $Z5$ و میزان پودر باید برابر $F5$ باشد.

۲۳- اگر میزان چرک لباس‌ها برابر $D5$ و وزن لباس‌ها برابر $W2$ باشد. آنگاه زمان شستشو باید برابر $Z6$ و میزان پودر باید برابر $F6$ باشد.

۲۴- اگر میزان چرک لباس‌ها برابر $D5$ و وزن لباس‌ها برابر $W3$ باشد. آنگاه زمان شستشو باید برابر $Z7$ و میزان پودر باید برابر $F7$ باشد.

۲۵- اگر میزان چرک لباس‌ها برابر $D5$ و وزن لباس‌ها برابر $W4$ باشد. آنگاه زمان شستشو باید برابر $Z8$ و میزان پودر باید برابر $F8$ باشد.

در ادامه با نحوه دقیق ساخت و پیاده سازی اطلاعات فوق، با استفاده از ابزار رابط گرافیکی در نرم افزار MATLAB را شرح خواهیم داد.

رابط گرافیکی جعبه ابزار منطق فازی

در این قسمت از رابط گرافیکی جعبه ابزار منطق فازی در راستای ایجاد یک سیستم استنتاج فازی استفاده خواهیم کرد. که مسلماً استفاده از رابط گرافیکی، موجب تسهیل عملیات خواهد شد.

می‌توانید برای ساخت، ویرایش و مشاهده سیستم‌های استنتاج فازی از ۵ ابزار گرافیکی اولیه در این جعبه ابزار استفاده کنید. این ۵ ابزار عبارتند از:

- ویرایشگر سیستم‌های استنتاج فازی (FIS)

- ویرایشگر توابع عضویت

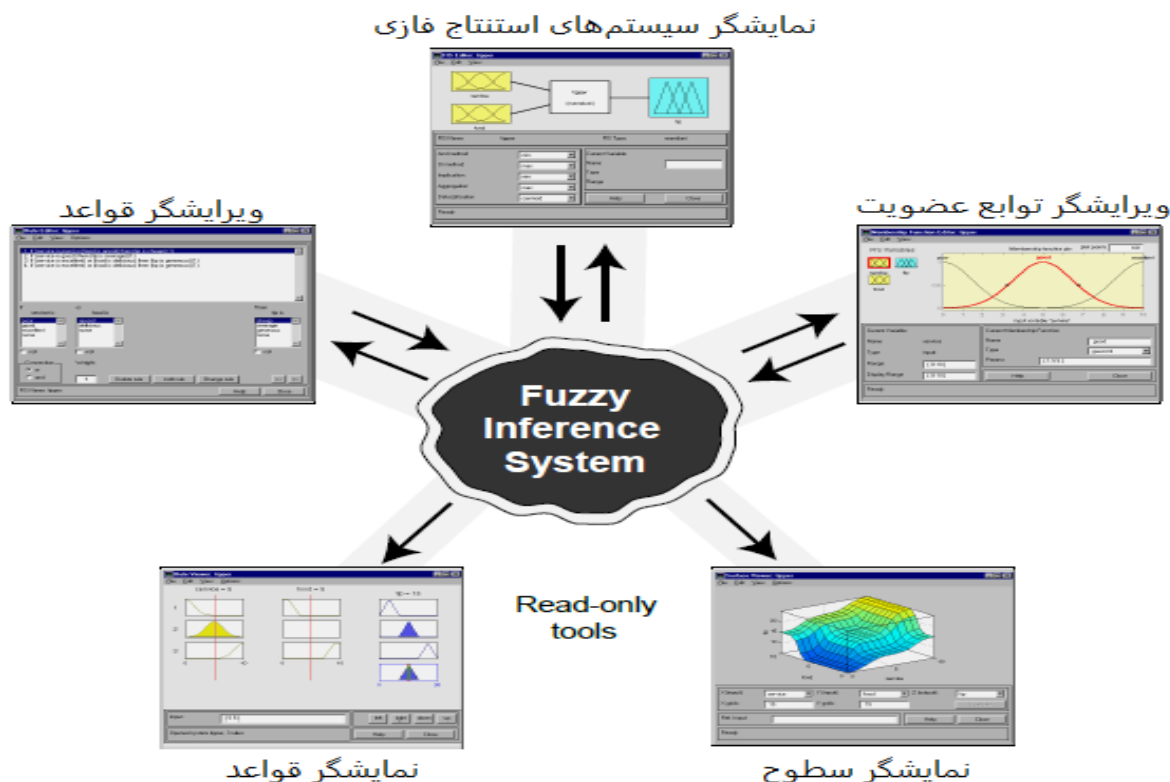
- ویرایشگر قواعد

- نمایشگر قواعد

- نمایشگر سطوح

رابط‌های گرافیکی یاد شده به صورت پویا به هم متصل هستند و ایجاد تغییرات از طریق هر یک از آنها بر روی سایرین تاثیر می‌گذارد. می‌توانید همزمان چند رابط را نیز باز کنید.

ویرایشگر سیستم استنتاج فازی برای کنترل سطوح بالای سیستم فازی مورد استفاده قرار می‌گیرد. در این ویرایشگر، تعداد متغیرهای ورودی و خروجی و نام آنها تعیین می‌شود جعبه ابزار منطق فازی هیچ محدودیتی برای تعداد ورودی‌ها ندارد؛ اما به هر حال با توجه به محدودیت حافظه‌ی کامپیوتر شما، تعداد متغیرهای ورودی محدود می‌شود. در صورتی که تعداد ورودی‌ها و یا توابع عضویت بسیار زیاد باشد، تحلیل از طریق ابزارهای گرافیکی بسیار دشوار می‌شود. از ویرایشگر توابع عضویت، برای تعریف شکل توابع عضویت مربوط به هر متغیر استفاده می‌شود. ویرایشگر قواعد، برای ویرایش لیست قواعد و تعیین رفتار سیستم مورد استفاده قرار می‌گیرد.



شکل ۲-۶: رابط‌های گرافیکی جعبه ابزار منطق فازی

نمایشگر قواعد و نمایشگر سطوح برای مشاهده، مورد استفاده قرار می‌گیرند. این دو رابط را می‌توان دو رابط فقط خواندنی دانست. نمایشگر قواعد یک محیط محاسبات تکنیکی در MATLAB است که برای نمایش نمودارهای مربوط به استنتاج فازی مورد استفاده قرار می‌گیرد. از این ابزار می‌توان برای تشخیص و نمایش قواعد فعال یا نحوه تاثیر هر یک از توابع عضویت در نتیجه نهایی استفاده نمود. نمایشگر سطوح برای نمایش وابستگی هر یک از خروجی‌ها به هر یک از ورودی‌ها مورد استفاده قرار می‌گیرد. از این نمایشگر برای رسم سطوح مربوط به خروجی‌ها به هر یک از ورودی‌ها مورد استفاده قرار می‌گیرد. از این نمایشگر برای رسم سطوح مربوط به خروجی‌های سیستم استفاده می‌شود.

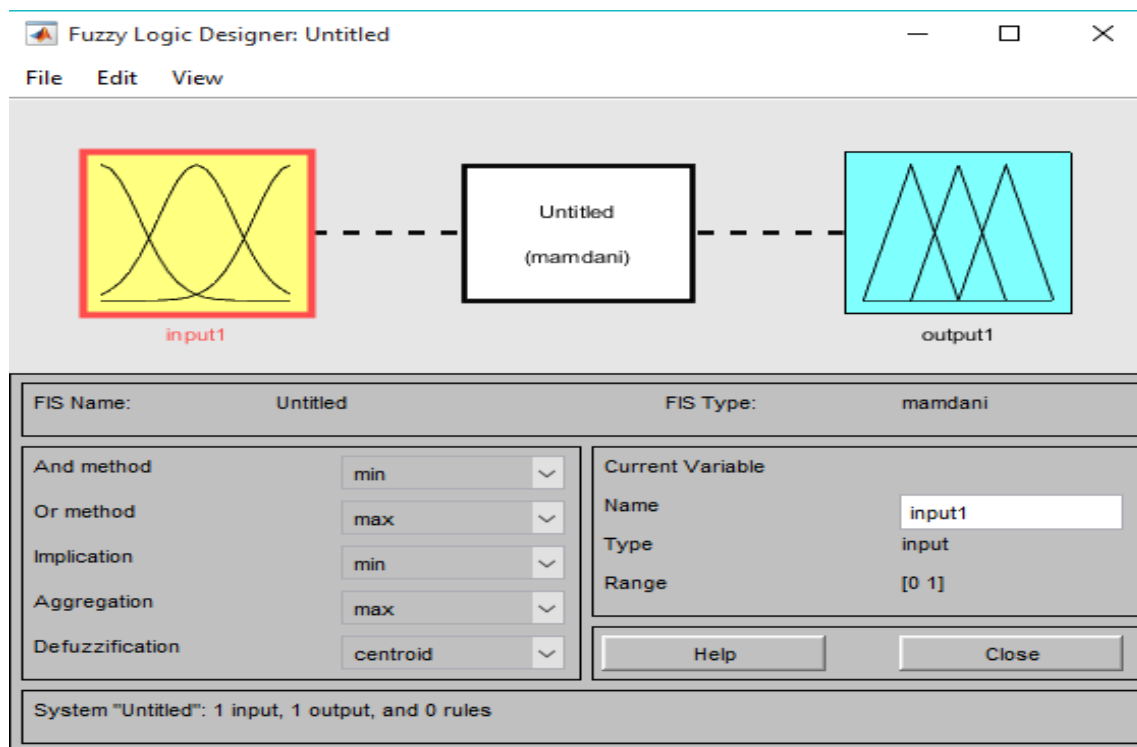
کلیه ابزارهای یاد شده دارای قابلیت تعامل و تبادل اطلاعات با یکدیگر هستند. هر یک از آنها دارای قابلیت خواندن و نوشتن در فضای کاری و یا یک فایل هستند. حتی نمایشگرها که قبلاً آنها را فقط خواندنی نامیدیم هم دارای قابلیت مبادله نمودارها با فضای کاری و ذخیره سازی آنها در قالب یک فایل هستند.

اگر بیش از یک ویرایشگر به صورت همزمان در یک سیستم باز باشند، هر یک از ویرایشگرها با هم در ارتباط بوده و در صورت بروز تغییر در هر یک، بقیه به روز رسانی می‌شوند. بنابراین اگر نام توابع عضویت با استفاده از ویرایشگر توابع عضویت تغییر کند، این تغییر به قواعد موجود در ویرایشگر قواعد منتقل می‌شود. همچنین امکان باز کردن ویرایشگرهای مختلف مربوط به چند سیستم متفاوت نیز وجود دارد.

ویرایشگر سیستم استنتاج فازی (FIS)

ویرایشگر سیستم استنتاج فازی، اطلاعاتی را در ارتباط با سیستم استنتاج فازی نمایش می‌دهد. برای باز کردن این ویرایشگر، دستور (fuzzy) را روی خط فرمان نرم افزار MATLAB اجرا کنید. با اجرای دستور، پنجره این ویرایشگر باز می‌شود. در شکل زیر، اجزای این پنجره نشان داده شده است.

در این مقاله قصد ساختن یک سیستم با ۲ ورودی و ۲ خروجی را داریم. ورودی‌های سیستم فازی مقدار چرک و وزن لباس‌ها بوده و دو خروجی آن نیز زمان و پودر شستشو می‌باشد.



شکل ۲-۷: پنجره ویرایشگر سیستم استنتاج فازی

برای اضافه کردن متغیر ورودی و خروجی دوم و تغییر نام متغیرها به صورت زیر عمل می‌کنیم:

۱- منوی `Input > Add variable > Edit` را انتخاب می‌کنیم. به این ترتیب یک جعبه زرد رنگ با برچسب `Input2` نمایش داده می‌شود.

۲- منوی `Output > Add variable > Edit` را انتخاب می‌کنیم. به این ترتیب یک جعبه آبی رنگ با برچسب `output2` نمایش داده می‌شود.

۳- روی `input1` کلیک کنید. به این ترتیب جعبه‌ی مربوط به آن با حاشیه قرمز رنگ برجسته می‌شود. فیلد `Name` مربوط به `input1` را ویرایش کرده و برابر `Dirty` (کثیفی لباس‌ها) قرار دهید و سپس `Enter` کنید.

۴- روی `input2` کلیک کنید. به این ترتیب جعبه‌ی مربوط به آن با حاشیه قرمز رنگ برجسته می‌شود. فیلد `Name` مربوط به `input2` را ویرایش کرده و برابر `Weight` (وزن لباس‌ها) قرار دهید و سپس `Enter` کنید.

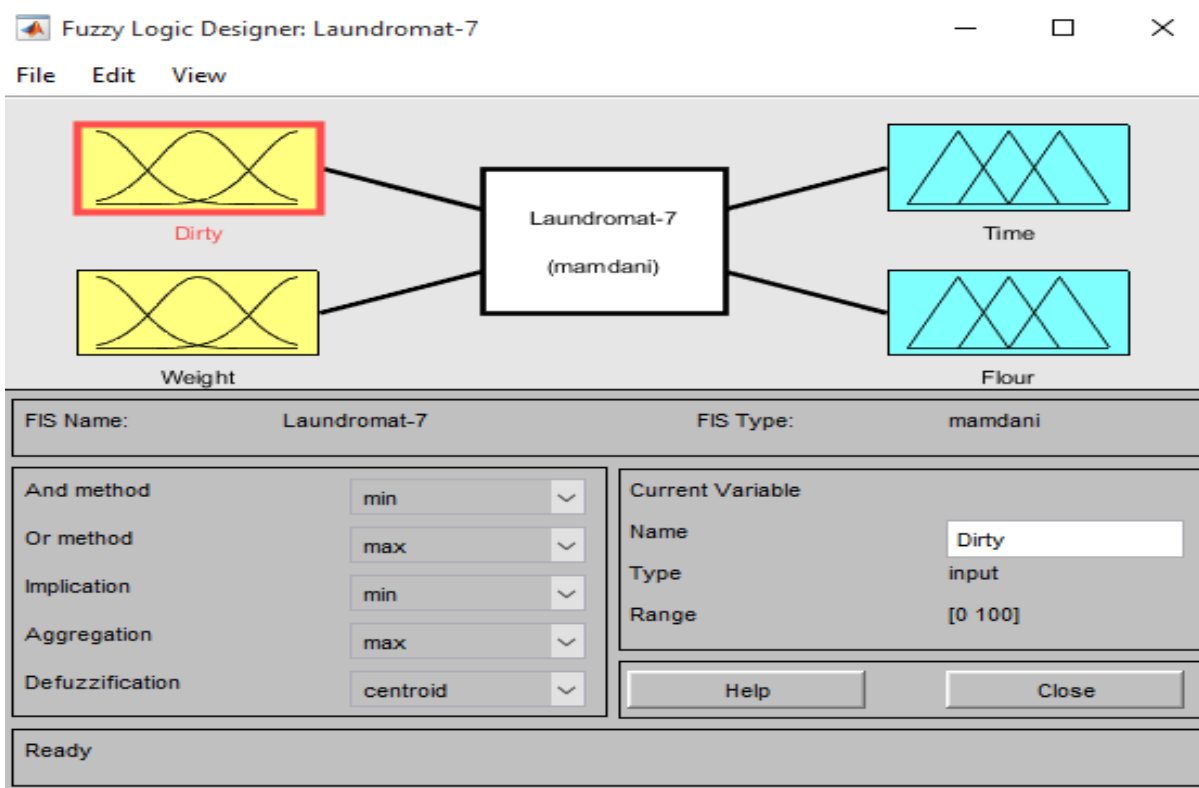
۵- روی جعبه آبی رنگ `output1` کلیک کنید تا برجسته شود. فیلد نام مربوط به `output1` را برابر `Time` (زمان شستشو) قرار داده و `Enter` کنید.

۶- سپس روی جعبه آبی رنگ مربوط به `output2` کلیک کنید. نام `output2` را به `Flour` (میزان پودر شستشو) تغییر دهید و `Enter` کنید.

۷- منوی `Workspace > Export to > File` را انتخاب کنید.

۸- روی پنجره باز شده، فیلد `Workspace variable` را برابر `Laundromat-7` قرار داده و روی `OK` کلیک کنید.

به این ترتیب یک متغیر جدید تحت عنوان `Laundromat-7` در فضای کاری ایجاد می‌شود که حاوی کلیه اطلاعات در مورد این سیستم است. با ذخیره سازی در فضای کاری با یک نام جدید می‌توانید نام کل سیستم را نیز عوض نمایید. پس از انجام تغییرات یاد شده، پنجره مربوط به ویرایشگر سیستم استنتاج فازی (FIS) به صورت شکل زیر خواهد بود.



شکل ۲-۸: پنجره مربوط به ویرایشگر سیستم استنتاج فازی (FIS)

اجازه دهید تا کلیه گزینه‌های مربوط به استنتاج در سمت چپ و پایین پنجره در حالت پیش فرض خود باقی بمانند. به این ترتیب شما کلیه اطلاعات مربوط به این پروژه را وارد کرده‌اید. حال نوبت به تعیین تابع عضویت مربوط به هریک از متغیرها می‌رسد. برای انجام این کار، باید ویرایشگر توابع عضویت را باز کنید. شما به سه طریق می‌توانید این کار را انجام دهید.

- از طریق پنجره ویرایشگر FIS و از منوی Edit>Membership functions

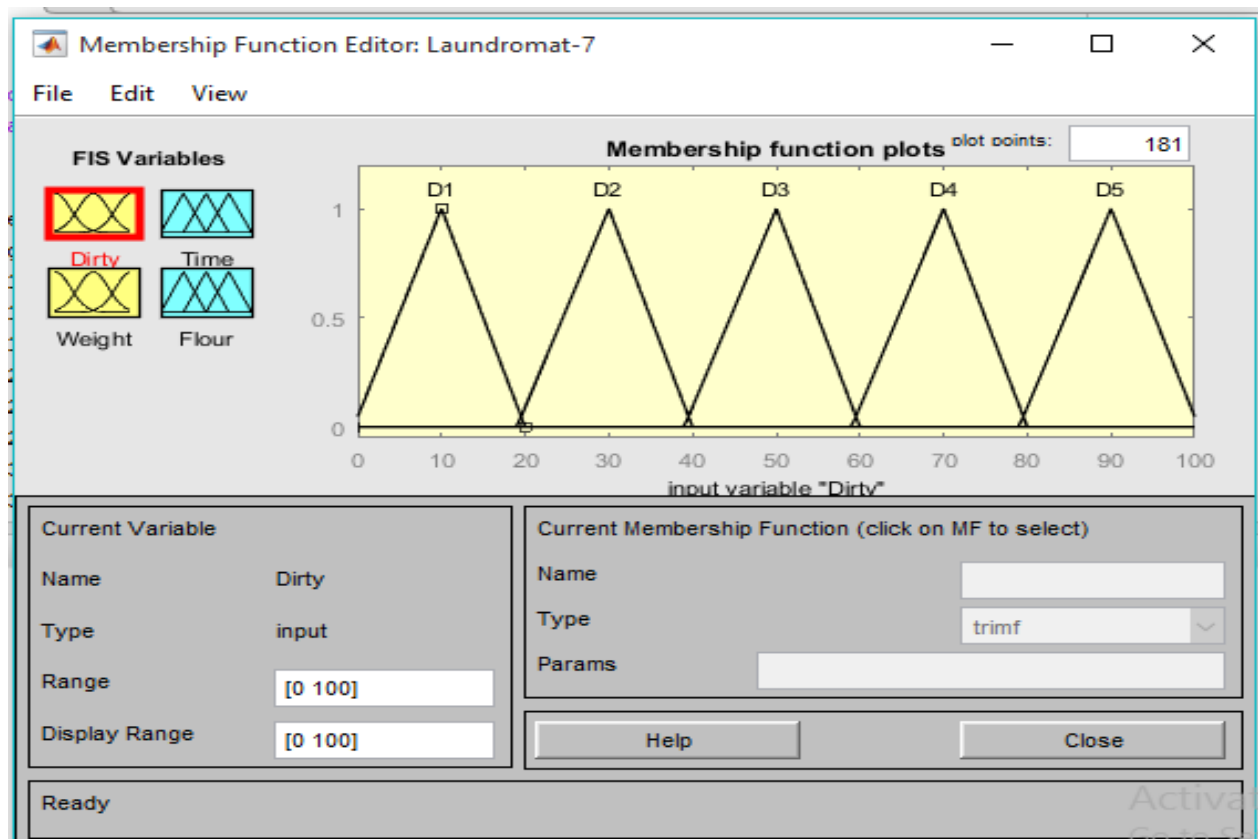
- از طریق پنجره ویرایشگر FIS و با دو بار کلیک روی آیکون آبی رنگ

- از روی خط فرمان با اجرای دستور mfededit

ویرایشگر توابع عضویت

ویرایشگر توابع عضویت، یک ابزار مناسب جهت نمایش و ویرایش توابع عضویت مربوط به هر یک از متغیرهای ورودی و خروجی سیستم استنتاج فازی می‌باشد. ویرایشگر تابع عضویت همان‌طور که در شکل زیر می‌بینید در برخی

از ویژگی‌ها با ویرایشگر FIS مشترک است. در واقع هر ۵ رابط گرافیکی اولیه در منو، خطوط وضعیت و کلیدهای Help و Close مشترک و مشابه می‌باشند.



شکل ۲-۹: ویرایشگر توابع عضویت

زمانی که ویرایشگر توابع عضویت را در مورد یک سیستم استنتاج فازی که هنوز وجود ندارد، باز می‌کنید مشاهده خواهید کرد که هیچ تابع عضویتی به متغیرهای تعریف شده در ویرایشگر FIS نسبت داده نشده است. در سمت چپ و بالای ناحیه‌ی مربوط به گراف در پنجره‌ی ویرایشگر توابع عضویت، می‌توانید هریک از متغیرها را برای تعیین تابع عضویت انتخاب نمایید. پس از انتخاب متغیر از منوی Edit گزینه‌ی Add MFs را انتخاب کنید. یک پنجره‌ی جدید باز می‌شود که اجازه‌ی انتخاب نوع تابع عضویت و تعداد توابع عضویت مربوط به متغیر انتخاب شده را می‌دهد. در گوشه‌ی پایین و سمت راست پنجره، می‌توانید نام، نوع و پارامترهای هر یک از توابع عضویت را پس از انتخاب آنها تغییر دهید.

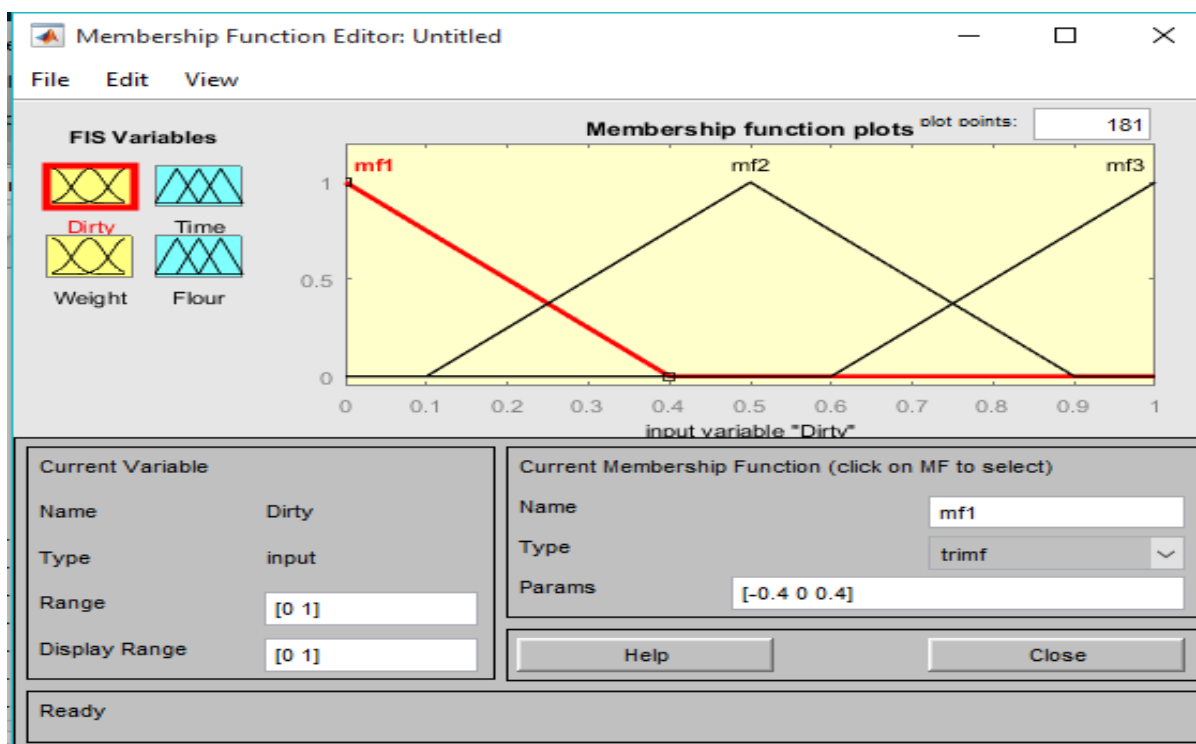
نمودار و تابع عضویت مربوط به متغیر انتخاب شده در قسمت گراف پنجره نشان داده می‌شود. این تابع عضویت را می‌توان به دو طریق دستکاری نمود. می‌توانید با استفاده از ماوس، هر یک از توابع عضویت نشان داده شده در این گراف را انتخاب نموده و سپس آن را روی گراف انتقال دهید. به این ترتیب توصیف ریاضی مربوط به تابع عضویت برای یک متغیر تغییر می‌کند. همچنین می‌توانید با کلیک و جابه‌جایی مربع‌های روی یک تابع عضویت در آن تغییر ایجاد کنید و آن را منبسط و یا منقبض نمایید. این عملیات باعث تغییر در پارامترهای مربوط به تابع عضویت می‌شود.

در زیرگراف، اطلاعاتی در مورد نوع و نام متغیرهای فعلی روی پنجره وجود دارد. در این ناحیه دو ناحیه‌ی متنی وجود دارد که شما با استفاده از آنها می‌توانید محدودیت روی بازه‌ی متغیرها و محدودیت‌های مربوط به رسم گراف را معین نمایید. تغییر در ناحیه‌ی دوم تاثیری بر روی سیستم ندارد.

فرآیند تعیین توابع عضویت مربوط به ورودی‌های پروژه به صورت ذیل می‌باشد:

الف) تعیین توابع عضویت مربوط به ورودی اول (Dirty)

۱- روی متغیر ورودی Dirty دو بار کلیک کنید تا پنجره ویرایشگر تابع عضویت باز شود. مطابق شکل زیر



شکل ۲-۱۰: پنجره ویرایشگر توابع عضویت

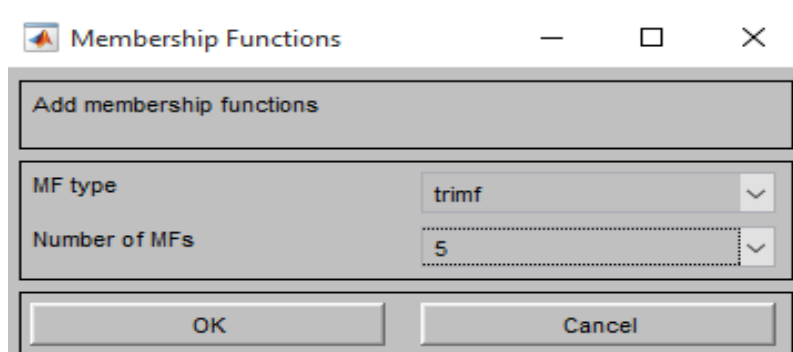
۲- روی پنجره‌ی ویرایشگر تابع عضویت فیلد Display Range و Range را برابر [۰ ۱۰۰] قرار دهید.

۳- تابع عضویت مربوط به متغیر ورودی Dirty را طی مراحل زیر ایجاد نمایید.

الف. با انتخاب منوی Edit> Remove All MFs کلیه توابع عضویت پیش فرض مربوط به متغیر Dirty را حذف کنید.

ب. با انتخاب منوی Edit> Add MFs پنجره‌ی مربوط به توابع عضویت را باز کنید.

ج. در پنجره انتخاب تابع عضویت، trimf را به عنوان مقدار فیلد MF Type وارد نمایید.



شکل ۲-۱۱: اضافه کردن یک تابع عضویت

د. دقت کنید که عدد ۵ به عنوان تعداد توابع عضویت تعیین شده باشد.

ن. روی کلید OK کلیک کنید تا ۵ منحنی مثلثی به متغیر Dirty اضافه شوند.

۴- توابع عضویت مربوط به متغیر Dirty را تغییر نام داده و پارامترهای مربوط به آن را تعیین نمایید.

در این راستا مطابق با مراحل زیر عمل نمایید.

الف. برای انتخاب منحنی mf1 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر D1 و ناحیه‌ی Params را برابر [-0.5 10 20] قرار دهید. سه مقدار

تعیین شده در فیلد Params معین کننده انحراف معیار و مرکز منحنی مثلثی هستند.

ب. برای انتخاب منحنی mf2 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر D2 و ناحیه‌ی Params را برابر [19 30 40] قرار دهید.

پ. برای انتخاب منحنی mf3 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر D3 و ناحیه‌ی Params را برابر [39 50 60] قرار دهید.

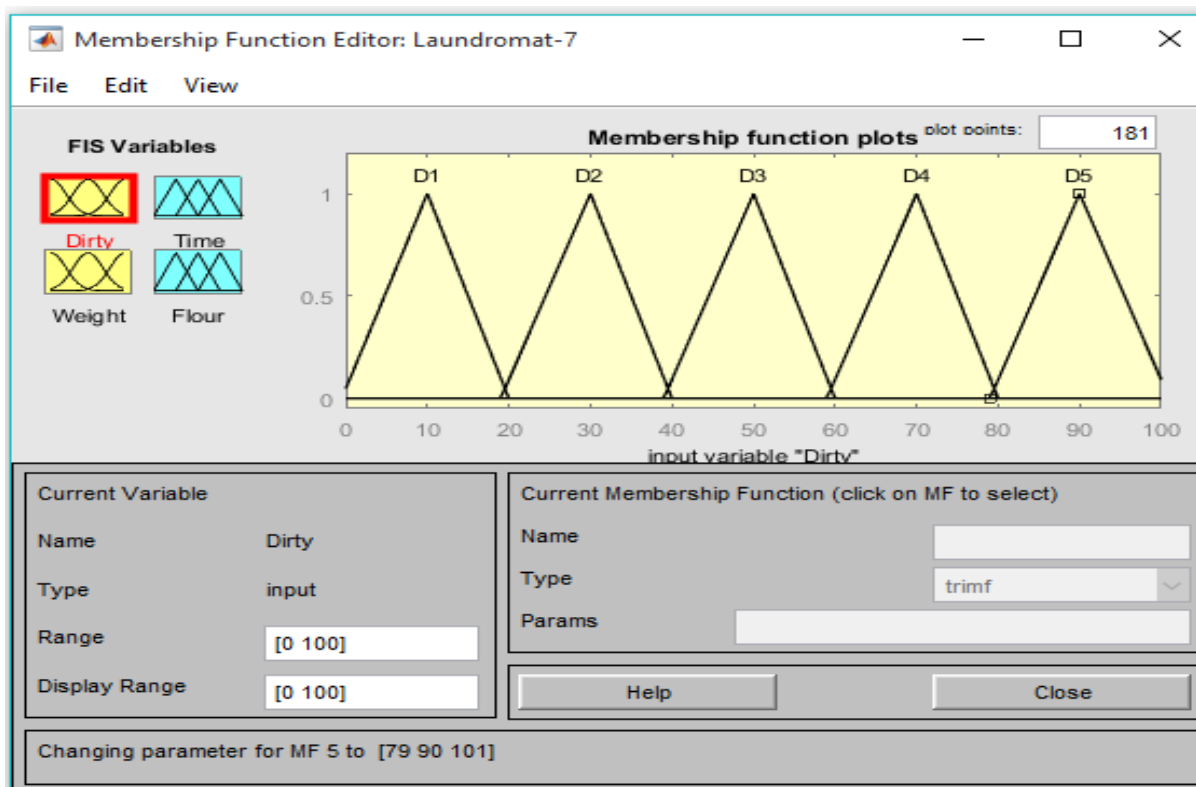
ت. برای انتخاب منحنی mf4 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر D4 و ناحیه‌ی Params را برابر [59 70 80] قرار دهید.

ث. برای انتخاب منحنی mf5 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر D5 و ناحیه‌ی Params را برابر [79 90 101] قرار دهید.

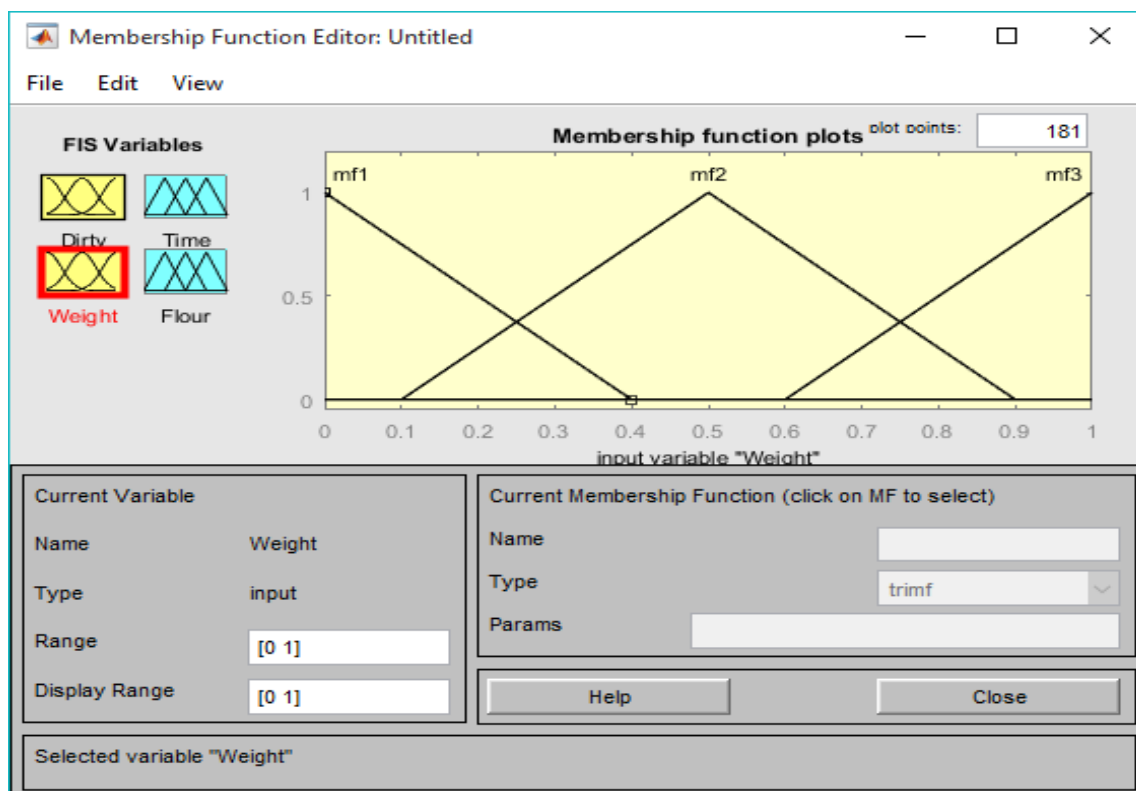
به این ترتیب پنجره‌ی ویرایشگر توابع عضویت ورودی اول به صورت شکل زیر در خواهد آمد.



شکل ۲-۱۲: پنجره ویرایشگر توابع عضویت مربوط به ورودی Dirty

ب) تعیین توابع عضویت مربوط به ورودی دوم (Weight)

۱- روی متغیر ورودی Weight دو بار کلیک کنید تا پنجره ویرایشگر تابع عضویت باز شود. مطابق شکل زیر.



شکل ۲-۱۳: پنجره ویرایشگر توابع عضویت

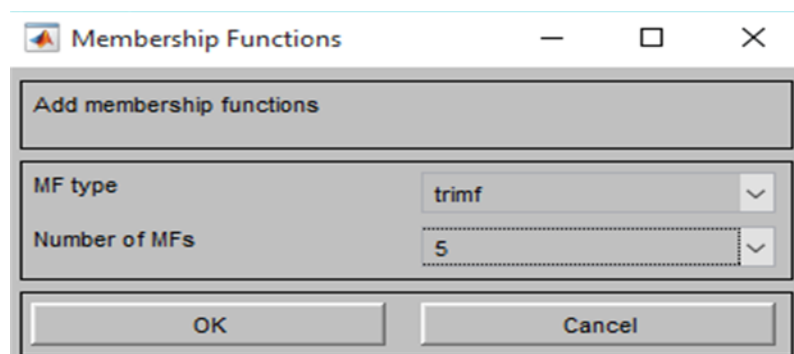
۲- روی پنجره ویرایشگر تابع عضویت فیلد Display Range و Range را برابر [۰ ۷] قرار دهید.

۳- تابع عضویت مربوط به متغیر ورودی Weight را طی مراحل زیر ایجاد نمایید.

الف. با انتخاب منوی Edit> Remove All MFs کلیه توابع عضویت پیش فرض مربوط به متغیر Weight را حذف کنید.

ب. با انتخاب منوی Edit> Add MFs پنجره ویرایشگر تابع عضویت را باز کنید.

ج. در پنجره انتخاب تابع عضویت، trimf را به عنوان مقدار فیلد MF Type وارد نمایید.



شکل ۲-۱۴: اضافه کردن یک تابع عضویت

د. دقت کنید که عدد ۵ به عنوان تعداد توابع عضویت تعیین شده باشد.

ن. روی کلید OK کلیک کنید تا ۵ منحنی مثلثی به متغیر Weight اضافه شوند.

۴- توابع عضویت مربوط به متغیر Weight را تغییر نام داده و پارامترهای مربوط به آن را تعیین نمایید.

در این راستا مطابق با مراحل زیر عمل نمایید.

الف. برای انتخاب منحنی mf1 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر W0 و ناحیه‌ی Params را برابر $[-0.5 \ 0 \ 0.6]$ قرار دهید. سه مقدار

تعیین شده در فیلد Params معین کننده انحراف معیار و مرکز منحنی مثلثی هستند.

ب. برای انتخاب منحنی mf2 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر W1 و ناحیه‌ی Params را برابر $[0.5 \ 1.35 \ 2.2]$ قرار دهید.

پ. برای انتخاب منحنی mf3 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر W2 و ناحیه‌ی Params را برابر $[2.1 \ 2.95 \ 3.8]$ قرار دهید.

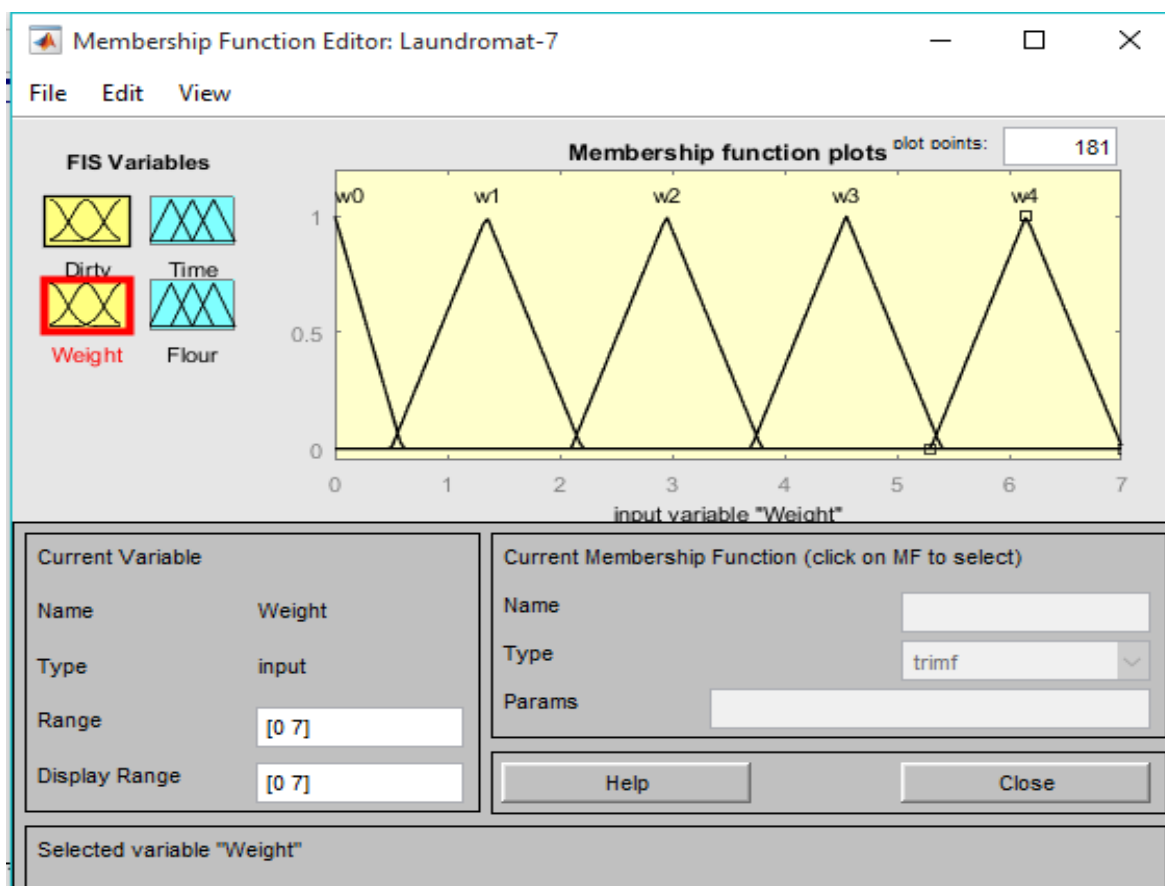
ت. برای انتخاب منحنی mf4 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر W3 و ناحیه‌ی Params را برابر $[3.7 \ 4.55 \ 5.4]$ قرار دهید.

ث. برای انتخاب منحنی mf5 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر W4 و ناحیه‌ی Params را برابر [5.3 6.15 7.01] قرار دهید.

به این ترتیب پنجره‌ی ویرایشگر توابع عضویت ورودی دوم به صورت شکل زیر در خواهد آمد.



شکل ۳-۱۳: پنجره ویرایشگر توابع عضویت مربوط به ورودی Weight

فرآیند تعیین توابع عضویت مربوط به خروجی‌های پروژه به صورت ذیل می‌باشد:

الف) تعیین توابع عضویت مربوط به خروجی اول (Time)

۱- در ناحیه مربوط به FIS variables متغیر خروجی Time را انتخاب کنید.

۲- [۰ ۹۵] را به عنوان ناحیه‌های Range و Display Range وارد نمایید.

۳- توابع عضویت مربوط به متغیر Time را طی مراحل زیر تعیین نمایید.

الف. منوی **Add MFs > Edit** را برای حذف توابع عضویت پیش فرض مربوط به متغیر خروجی **Time** انتخاب نمایید.

ب. با انتخاب منوی **Edit > Remove All MFs** پنجره‌ی مربوط به انتخاب تابع عضویت را باز کنید.

پ. روی پنجره انتخاب تابع عضویت، **trimf** (مثلثی) را به عنوان **MF Type** انتخاب کنید.

ت. مقدار ۹ را به عنوان **Number of MFs** تعیین کنید.

ث. با کلیک روی **OK**، نه منحنی مثلثی به عنوان توابع عضویت متغیر **Time** انتخاب می‌شوند.

۴- تغییر نام توابع عضویت مربوط به متغیر خروجی **Time** و تعیین پارامترهای مربوط به آن را طی مراحل زیر انجام

دهید:

الف. در ناحیه‌ی مربوط به **FIS Variables** روی متغیر خروجی **Time** کلیک کرده و آن را انتخاب نمایید.

ب. برای انتخاب منحنی **mf1** روی آن کلیک نموده و در قسمت **Current Membership Function** ناحیه‌ی **Name** را برابر **Error** و ناحیه‌ی **Params** را برابر **[0.5 0 -0.5]** قرار دهید. سه مقدار تعیین شده در فیلد **Params** معین کننده انحراف معیار و مرکز منحنی مثلثی هستند.

پ. برای انتخاب منحنی **mf2** روی آن کلیک نموده و در قسمت **Current Membership Function** ناحیه‌ی **Name** را برابر **T1** و ناحیه‌ی **Params** را برابر **[0.4 6 12]** قرار دهید.

ت. برای انتخاب منحنی **mf3** روی آن کلیک نموده و در قسمت **Current Membership Function** ناحیه‌ی **Name** را برابر **T2** و ناحیه‌ی **Params** را برابر **[11 18 24]** قرار دهید.

ث. برای انتخاب منحنی **mf4** روی آن کلیک نموده و در قسمت **Current Membership Function** ناحیه‌ی **Name** را برابر **T3** و ناحیه‌ی **Params** را برابر **[23 30 36]** قرار دهید.

ج. برای انتخاب منحنی **mf5** روی آن کلیک نموده و در قسمت **Current Membership Function** ناحیه‌ی **Name** را برابر **T4** و ناحیه‌ی **Params** را برابر **[35 42 48]** قرار دهید.

پ. برای انتخاب منحنی mf2 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر T5 و ناحیه‌ی Params را برابر [47 54 60] قرار دهید.

ت. برای انتخاب منحنی mf3 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر T6 و ناحیه‌ی Params را برابر [59 66 72] قرار دهید.

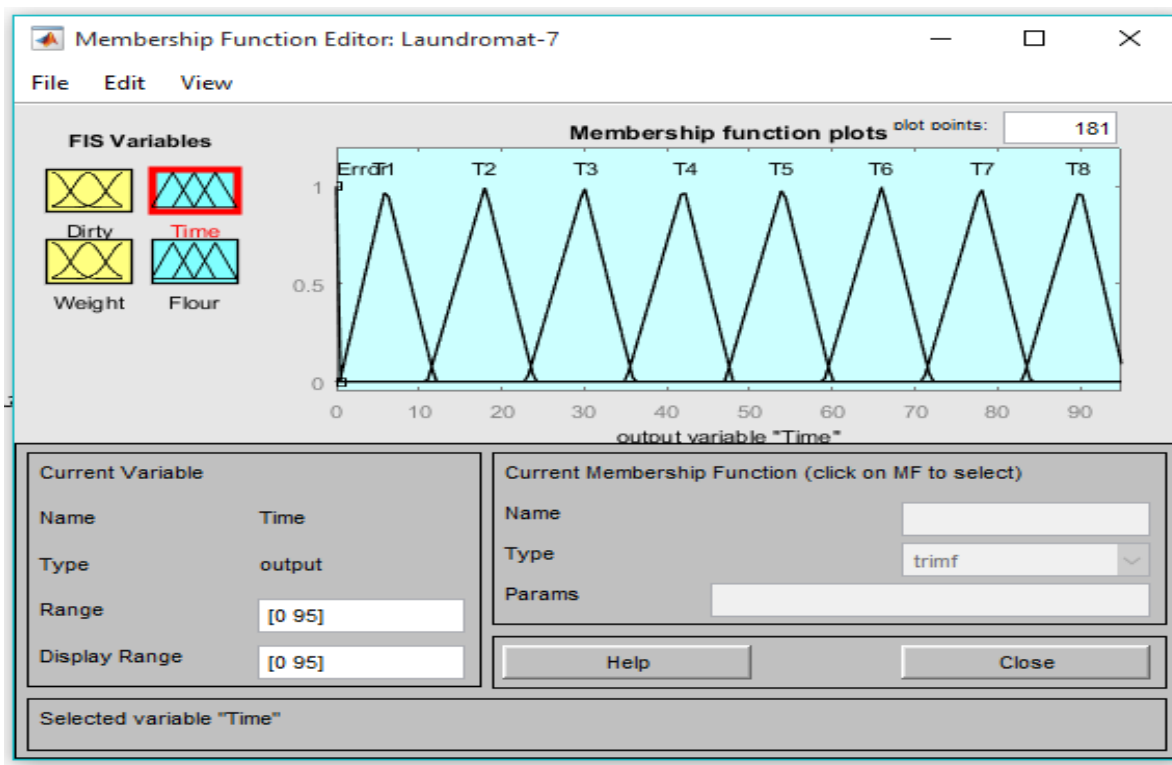
ث. برای انتخاب منحنی mf4 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر T7 و ناحیه‌ی Params را برابر [71 78 84] قرار دهید.

ج. برای انتخاب منحنی mf5 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر T8 و ناحیه‌ی Params را برابر [83 90 95.5] قرار دهید

به این ترتیب پنجره‌ی ویرایشگر توابع عضویت خروجی (Time) به صورت شکل زیر در خواهد آمد.



شکل ۲-۱۴: پنجره مربوط به توابع عضویت خروجی Time

ب) تعیین توابع عضویت مربوط به خروجی دوم (Flour)

۱- در ناحیه مربوط به FIS variables متغیر خروجی Flour را انتخاب کنید.

۲- [۰ ۴۲۵] را به عنوان ناحیه‌های Display Range و Range وارد نمایید.

۳- توابع عضویت مربوط به متغیر Flour را طی مراحل زیر تعیین نمایید.

الف. منوی Edit> Add MFs را برای حذف توابع عضویت پیش فرض مربوط به متغیر خروجی Flour انتخاب نمایید.

ب. با انتخاب منوی Edit> Remove All MFs پنجره‌ی مربوط به انتخاب تابع عضویت را باز کنید.

پ. روی پنجره انتخاب تابع عضویت، trimf (مثلثی) را به عنوان MF Type انتخاب کنید.

ت. مقدار ۹ را به عنوان Number of MFs تعیین کنید.

ث. با کلیک روی OK، نه منحنی مثلثی به عنوان توابع عضویت متغیر Flour انتخاب می‌شوند.

۴- تغییر نام توابع عضویت مربوط به متغیر خروجی Flour و تعیین پارامترهای مربوط به آن را طی مراحل زیر انجام دهید:

الف. در ناحیه‌ی مربوط به FIS Variables روی متغیر خروجی Flour کلیک کرده و آن را انتخاب نمایید.

ب. برای انتخاب منحنی mf1 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر Error و ناحیه‌ی Params را برابر [0.5 0 -0.5] قرار دهید. سه مقدار

تعیین شده در فیلد Params معین کننده انحراف معیار و مرکز منحنی مثلثی هستند.

پ. برای انتخاب منحنی mf2 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر F1 و ناحیه‌ی Params را برابر [0.4 26.5 53] قرار دهید.

ت. برای انتخاب منحنی mf3 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر F2 و ناحیه‌ی Params را برابر [50 79.5 106] قرار دهید.

ث. برای انتخاب منحنی mf4 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر F3 و ناحیه‌ی Params را برابر [103 132.5 159] قرار دهید.

ج. برای انتخاب منحنی mf5 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر F4 و ناحیه‌ی Params را برابر [156 185.5 212] قرار دهید.

پ. برای انتخاب منحنی mf2 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر F5 و ناحیه‌ی Params را برابر [209 238.5 265] قرار دهید.

ت. برای انتخاب منحنی mf3 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر F6 و ناحیه‌ی Params را برابر [262 291.5 318] قرار دهید.

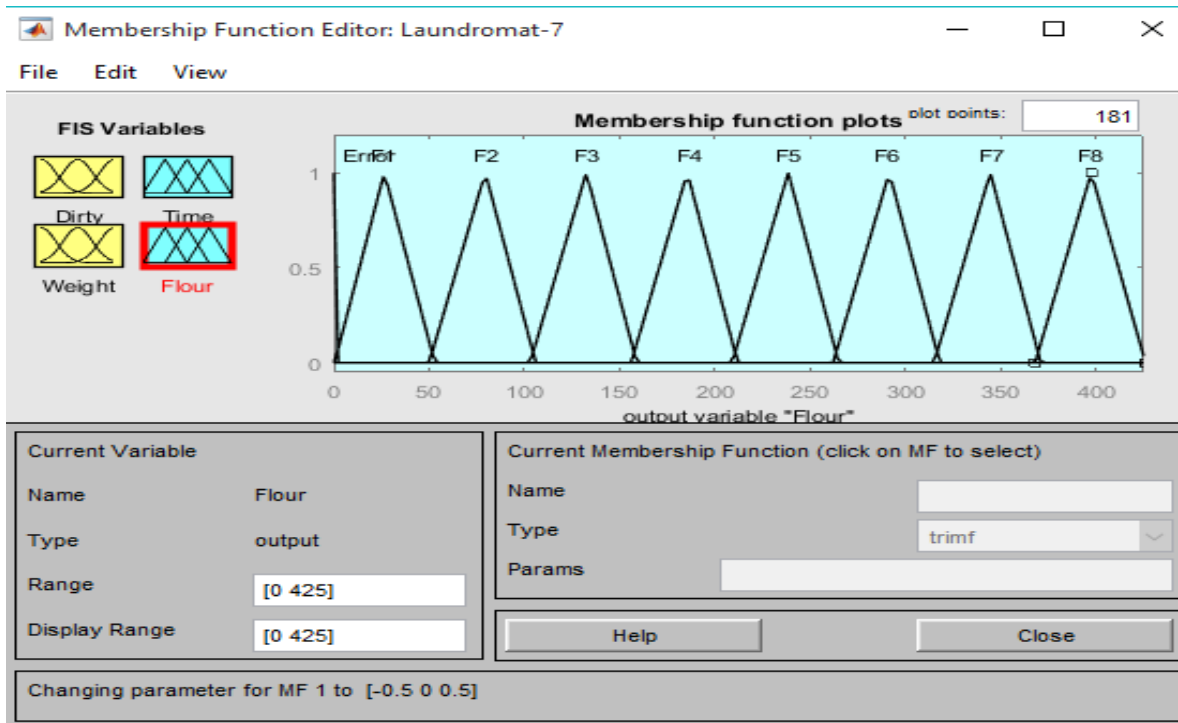
ث. برای انتخاب منحنی mf4 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر F7 و ناحیه‌ی Params را برابر [315 344.5 371] قرار دهید.

ج. برای انتخاب منحنی mf5 روی آن کلیک نموده و در قسمت Current Membership Function

ناحیه‌ی Name را برابر F8 و ناحیه‌ی Params را برابر [368 397.5 425.5] قرار دهید

به این ترتیب پنجره‌ی ویرایشگر توابع عضویت خروجی (Flour) به صورت شکل زیر در خواهد آمد.

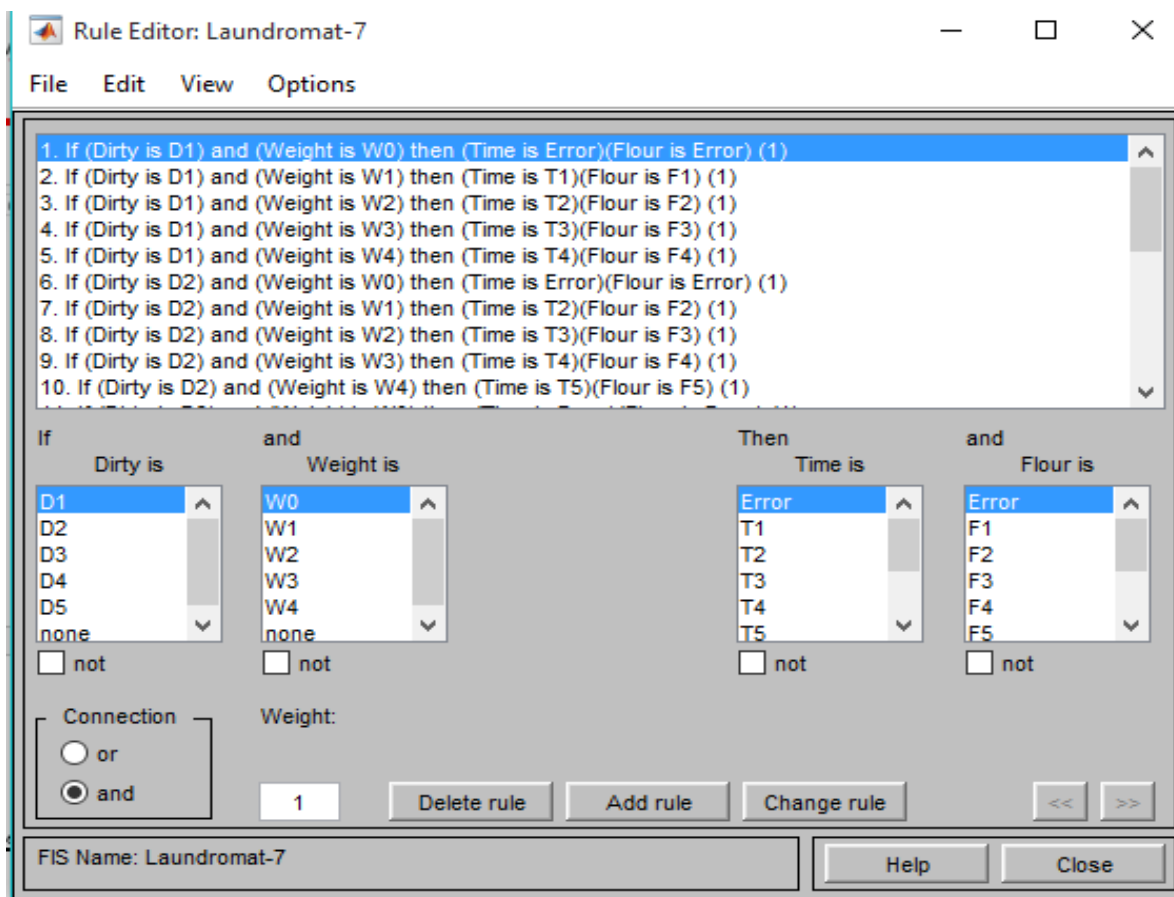


شکل ۲-۱۵: پنجره مربوط به توابع عضویت خروجی Flour

حال متغیرها نام گذاری شده و توابع عضویت تنظیم شده اند و نوبت به ورود قواعد پروژه رسیده است. برای باز کردن ویرایشگر قواعد به منوی Edit رفته و Rules را انتخاب کنید. همچنین می‌توانید از دستور rulesedit روی خط فرمان در نرم افزار MATLAB استفاده نمایید.

ویرایشگر قواعد

ایجاد قواعد با استفاده از ویرایشگر قواعد گرافیکی قواعد، بسیار ساده می‌باشد. بر مبنای نحوه‌ی توصیفات انجام شده برای متغیرهای ورودی و خروجی در ویرایشگر استنتاج فازی، ویرایشگر قواعد به شما اجازه‌ی خودکار قواعد را می‌دهد. شمایی از این ویرایشگر در شکل زیر نشان داده شده است.



شکل ۲-۱۶: ویرایشگر قواعد

شما با استفاده از این رابط گرافیکی می‌توانید:

- با انتخاب یک آیتم از جعبه‌ی متغیرهای ورودی و خروجی و انتخاب یک نوع اتصال و کلیک روی **Add Rule** یک قاعده به مجموعه قواعد اضافه نمایید. می‌توانید با انتخاب **none** به عنوان مقدار یک متغیر آن را از قاعده حذف نمایید. همچنین با انتخاب جعبه‌ی **not** می‌توانید مقدار یک متغیر را قرینه نمایید.
- برای حذف یک قاعده باید آن را انتخاب نموده و بر روی **Delete Rule** کلیک کنید.
- برای ویرایش یک قاعده آن را انتخاب نموده و پس از انجام تغییرات بر روی **Chenge Rule** کلیک کنید.
- برای تعیین وزن هر قاعده، می‌توانید وزن مورد نظر را به صورت یک عدد بین 0 و 1 در ناحیه‌ی **Weight** وارد نمایید. در صورت عدم تعیین، وزن آن به صورت پیش فرض برابر ۱ در نظر گرفته می‌شود.

ویرایشگر قواعد دارای منو و خط وضعیت مشابه با ویرایشگرهای استنتاج فازی و توابع عضویت می باشد. می توانید با استفاده از گزینه های موجود در منو، یک سیستم فازی را باز، ذخیره سازی و ویرایش نموده و در نهایت آن را ببندید. همچنین به کمک این منو می توانید:

- فرمت نمایش را با انتخاب منوی Options > Format تنظیم نمایید.

- زبان را با استفاده از منوی Options > Language تنظیم کنید.

همچنین در پنجره ی مربوط به ویرایشگر قواعد مانند سایر ویرایشگرها دو کلید Help و Close وجود دارد که می توانید از آنها برای دریافت راهنمایی و بستن پنجره استفاده کنید. برای درج اولین قاعده مراحل زیر را انجام دهید:

- برای متغیر ورودی Dirty مقدار D1 را انتخاب کنید.

- برای متغیر ورودی Weight مقدار W0 را انتخاب کنید.

- برای متغیر خروجی Time مقدار Error را انتخاب کنید.

- در بلاک Connection دستور and را انتخاب کنید.

- برای متغیر خروجی Flour مقدار Error را انتخاب کنید.

قاعده حاصل به صورت زیر خواهد بود:

1. If (Dirty is D1) and (Weight is W0) then (Time is Error)(Flour is Error) (1)

عدد داخل پرانتز مقدار وزن مربوط به قاعده را مشخص می کند.

مطابق مراحل قبل تمام قواعد را نیز به مجموعه قواعد اضافه نمایید. به این ترتیب مجموعه ی قواعد به

صورت زیر در خواهند آمد:

- 1. If (Dirty is D1) and (Weight is W0) then (Time is Error)(Flour is Error) (1)
- 2. If (Dirty is D1) and (Weight is W1) then (Time is T1)(Flour is F1) (1)
- 3. If (Dirty is D1) and (Weight is W2) then (Time is T2)(Flour is F2) (1)
- 4. If (Dirty is D1) and (Weight is W3) then (Time is T3)(Flour is F3) (1)
- 5. If (Dirty is D1) and (Weight is W4) then (Time is T4)(Flour is F4) (1)
- 6. If (Dirty is D2) and (Weight is W0) then (Time is Error)(Flour is Error) (1)
- 7. If (Dirty is D2) and (Weight is W1) then (Time is T2)(Flour is F2) (1)
- 8. If (Dirty is D2) and (Weight is W2) then (Time is T3)(Flour is F3) (1)
- 9. If (Dirty is D2) and (Weight is W3) then (Time is T4)(Flour is F4) (1)
- 10. If (Dirty is D2) and (Weight is W4) then (Time is T5)(Flour is F5) (1)
- 11. If (Dirty is D3) and (Weight is W0) then (Time is Error)(Flour is Error) (1)
- 12. If (Dirty is D3) and (Weight is W1) then (Time is T3)(Flour is F3) (1)
- 13. If (Dirty is D3) and (Weight is W2) then (Time is T4)(Flour is F4) (1)
- 14. If (Dirty is D3) and (Weight is W3) then (Time is T5)(Flour is F5) (1)
- 15. If (Dirty is D3) and (Weight is W4) then (Time is T6)(Flour is F6) (1)
- 16. If (Dirty is D4) and (Weight is W0) then (Time is Error)(Flour is Error) (1)
- 17. If (Dirty is D4) and (Weight is W1) then (Time is T4)(Flour is F4) (1)
- 18. If (Dirty is D4) and (Weight is W2) then (Time is T5)(Flour is F5) (1)
- 19. If (Dirty is D4) and (Weight is W3) then (Time is T6)(Flour is F6) (1)
- 20. If (Dirty is D4) and (Weight is W4) then (Time is T7)(Flour is F7) (1)
- 21. If (Dirty is D5) and (Weight is W0) then (Time is Error)(Flour is Error) (1)
- 22. If (Dirty is D5) and (Weight is W1) then (Time is T5)(Flour is F5) (1)
- 23. If (Dirty is D5) and (Weight is W2) then (Time is T6)(Flour is F6) (1)
- 24. If (Dirty is D5) and (Weight is W3) then (Time is T7)(Flour is F7) (1)
- 25. If (Dirty is D5) and (Weight is W4) then (Time is T8)(Flour is F8) (1)

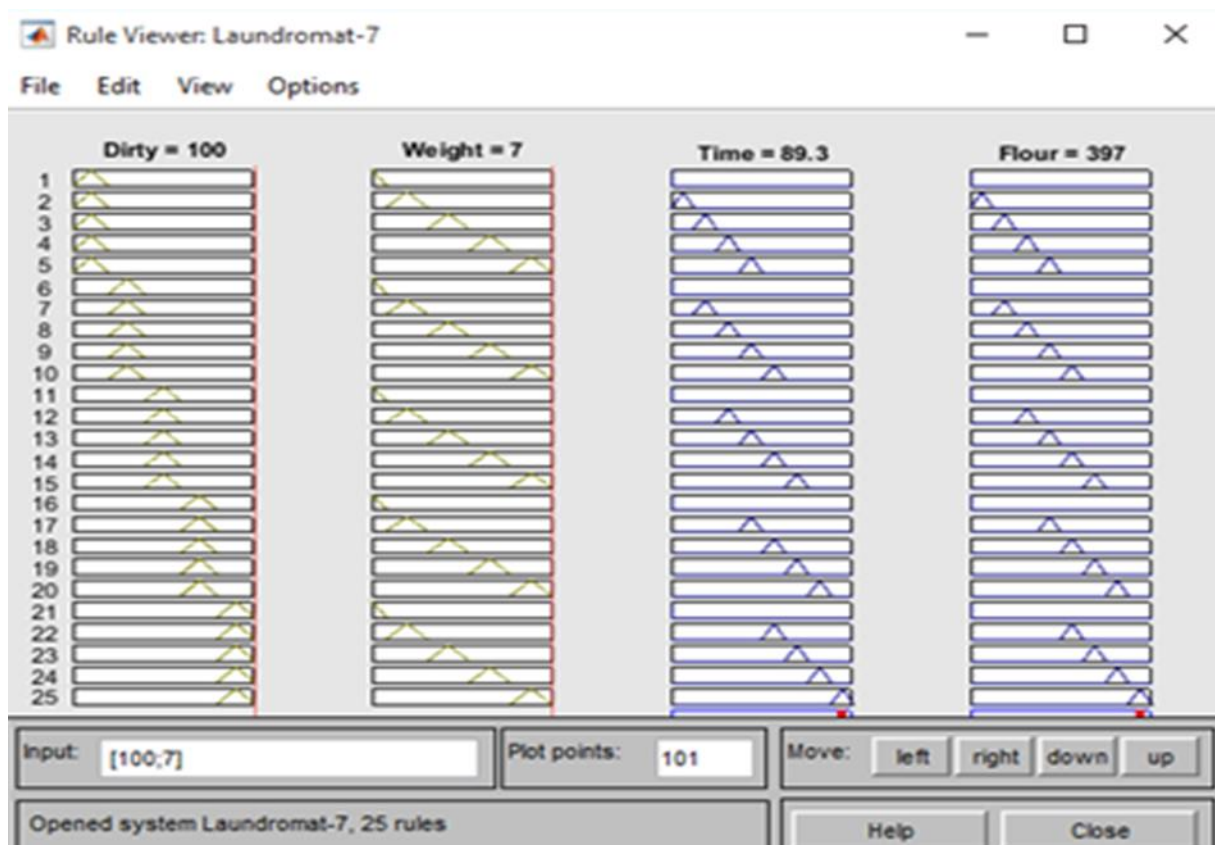
برای تغییر یک قاعده ابتدا روی قاعده‌ای که باید تغییر داده شود کلیک نموده و سپس تغییرات مورد نظر را روی

آن اعمال کنید و در نهایت روی کلید Change rule کلیک نمایید.

نمایشگر قواعد فازی

نمایشگر قواعد یک نقشه کلی از فرایند استنتاج فازی را نشان می‌دهد. در مورد پروژه با بازکردن رابط گرافیکی

مربوط به نمایشگر قواعد، نمودار زیر نمایش داده می‌شوند.



شکل ۲-۱۷: نمایشگر قواعد

نمودار ستون اول و دوم از شکل فوق (نمودار زرد رنگ) نشان دهنده‌ی توابع عضویت مربوط به فرض قواعد (قسمت if قواعد) هست. نمودارهای ستون سوم و چهارم (نمودارهای آبی) نشان دهنده‌ی توابع عضویت مربوط به نتیجه قواعد (قسمت then) هستند. در نهایت نمودار بیست‌وششم از ستون دوم و سوم، ارائه دهنده‌ی نمودار مربوط به تجمیع تصمیم‌های وزن دار شده می‌باشد. تصمیم اتخاذ شده نهایی بستگی به مقادیر ورودی دارد. خروجی غیر فازی شده به صورت یک خط عمودی قطور روی این نمودار نشان داده شده است.

متغیرها و مقادیر فعلی آنها در بالای هریک از ستون‌ها نمایش داده شده‌اند. در قسمت پایین و چپ پنجره‌ی یک ناحیه‌ی متنی با نام Input وجود دارد که شما می‌توانید مقادیر ورودی خاص را در آن وارد نمایید. و سپس Enter کنید. همچنین می‌توانید با کلیک کردن روی هریک از نمودارها و تغییر مکان خط عمودی قرمز رنگ مقدار هر متغیر را تنظیم کنید. با جابجایی این خط محاسبات جدید صورت گرفته و نتایج به نمایش در می‌آیند. با استفاده از این خط می‌توانید

کلیه خصوصیات مربوط به هریک از متغیرها را تنظیم نمایید. به کمک این رابط گرافیکی می‌توان روال دلالت قواعد و تجمیع آنها را به صورت گرافیکی مشاهده نمود.

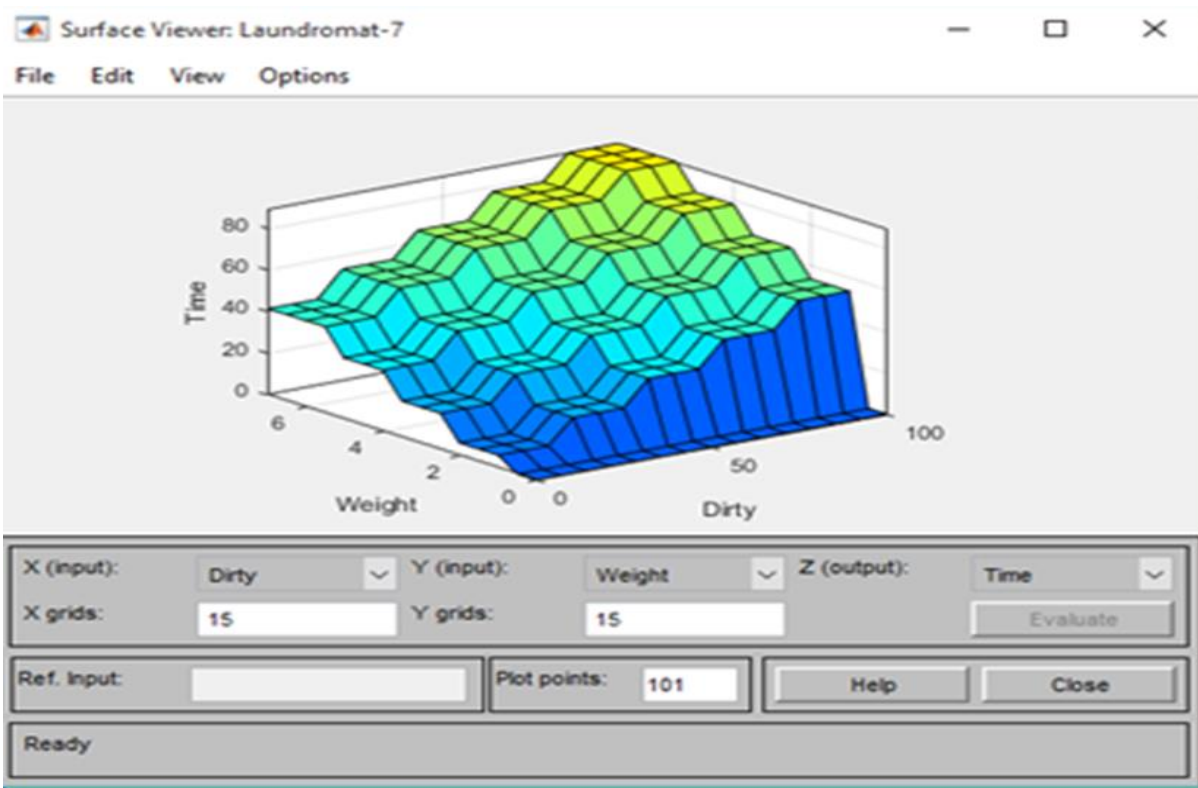
همچنین می‌توانید به کمک چهار کلید left, right, down, up نمودارها را در چهار جهت انتقال دهید. همچنین گزینه‌های موجود در منو به شما اجازه‌ی ذخیره سازی، بازکردن و یا ویرایش یک سیستم فازی را به کمک هر یک از ۵ ابزار گرافیکی می‌دهند.

نمایشگر قواعد، قابلیت تفسیر کل فرایند استنتاج فازی را فراهم می‌آورد. این نمایشگر، نحوه‌ی تاثیر هر یک از توابع عضویت روی نتیجه نهایی را نشان می‌دهد. البته در مورد سیستم‌های بزرگ استفاده از این رابط گرافیکی اندکی مشکل می‌شود، اما به هر حال این رابط در مورد سیستم‌های کم حجم تر با حدود ۳۰ قاعده و ۶ یا ۷ متغیر می‌تواند مفید واقع گردد.

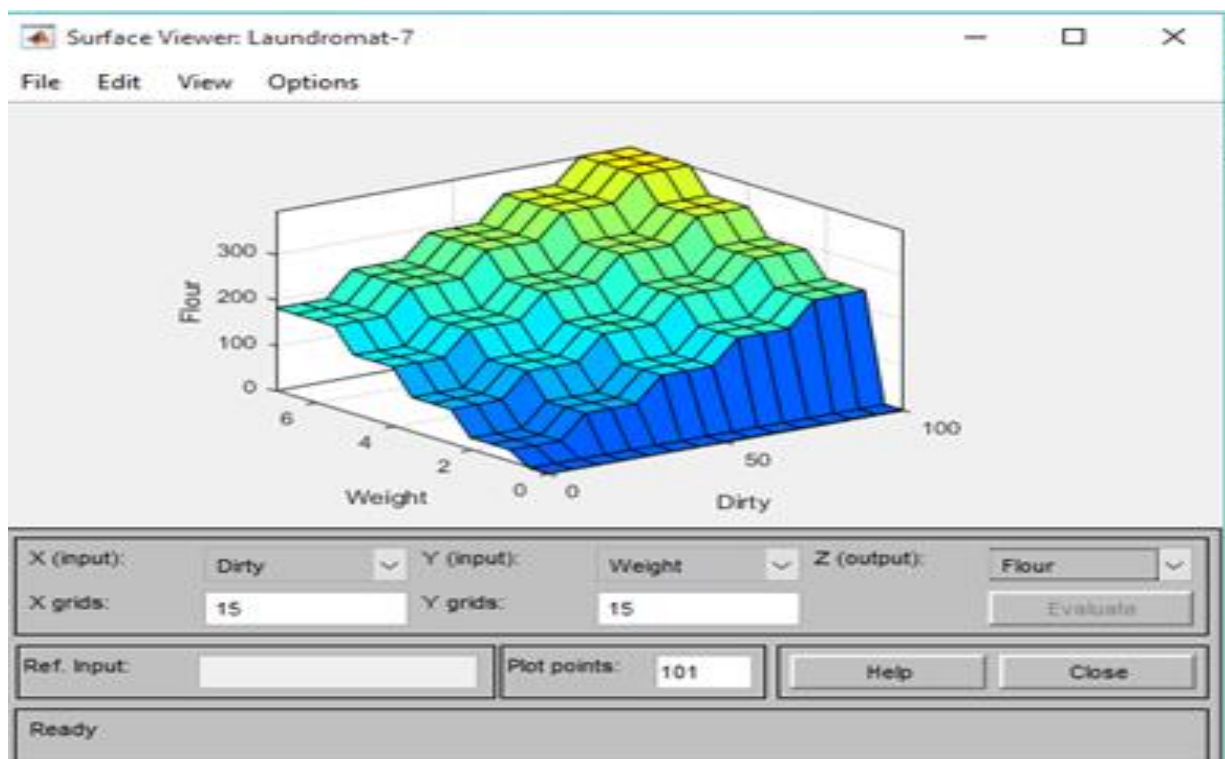
در صورتی که بخواهید سطوح کلی مربوط به خروجی‌های سیستم که بر مبنای ورودی‌های سیستم ترسیم شده‌اند را ببینید، می‌توانید از نمایشگر سطوح استفاده کنید. برای بازکردن این نمایشگر باید گزینه‌ی Surface را از منوی View انتخاب کنید. در بخش بعد به مفصل به این رابط گرافیکی خواهیم پرداخت.

نمایشگر سطوح

پس از بازکردن نمایشگر سطوح مربوط به پروژه، دو نمودار سه بعدی که یک نمودار نسبت تغییرات کثیفی و وزن البسه را نسبت به تغییرات زمان شستشو را نمایش می‌دهد و نمودار دیگر نسبت تغییرات کثیفی و وزن البسه را نسبت به تغییرات پودر شستشو را نمایش می‌دهد. مطابق شکل‌های زیر



شکل ۲-۱۸: تغییرات کثیفی و وزن البسه نسبت به زمان



شکل ۲-۱۹: تغییرات کثیفی و وزن البسه نسبت به پودر

رابط گرافیکی نمایشگر سطوح دارای سه منوی بازشونده $X(input)$, $Y(input)$, $Z(output)$ می باشد. که شما را قادر می سازد تا یکی از ورودی ها و یا خروجی ها را برای رسم در قالب نمودار انتخاب نمایید. در زیر قسمت یاد شده دو ناحیه متنی با نام های $Y grids$ و $X grids$ وجود دارند که امکان تعیین تعداد خطوط تورانه ای محور y ها و x ها را در اختیار شما قرار می دهند. با استفاده از این امکان، می توانید زمان محاسباتی را در مورد مسائل پیچیده، معقول تر نمایید. در صورتی که قصد ایجاد یک نمودار نرم تر را داشته باشید، می توانید از ناحیه $Plot points$ برای تعیین تعداد نقاطی که در محدوده ورودی و خروجی تابع عضویت ارزیابی می شود، استفاده نمایید. مقدار این ناحیه به صورت پیش فرض ۱۰۱ می باشد.

با کلیک بر روی Evaluate محاسبات آغاز شده و پس از تکمیل محاسبات، نمودار رسم می گردد. برای تغییر تعداد تورانه های هر یک از محور X یا Y ناحیه مربوط را تغییر داده Enter نمایید تا نمودار به روز رسانی شود.

رابط گرافیکی نمایشگر سطوح، دارای قابلیت های بسیار مفیدی در مورد مسائلی با دو ورودی (یا بیشتر) و یک خروجی می باشد. می توانید با جابه جایی محورها به کمک ماوس از زاویه جدیدی نمودار را مورد بررسی قرار دهید. می توان از ناحیه $Ref. Input$ برای ویرایش ورودی هایی که در رسم نمودار مورد استفاده قرار می گیرند، استفاده نمود.

بدین ترتیب با ۵ رابط گرافیکی مربوط به جعبه ابزار منطق فازی آشنا شده اید. اما شاید بپرسید "چرا اینقدر دردسر؟ به راحتی می توان با ایجاد یک جدول جستجو، مسائلی مانند پروژه و مسائل مشابه تصمیم گیری را حل نمود". اما باید گفت راه حل فازی مسائل، ضمن سهولت در پیاده سازی، امکان ایجاد تغییرات سریع در سیستم را تسهیل می نماید. همچنین این روش ضمن تولید پاسخ مناسب از انعطاف پذیری بالایی برخوردار است.

ورود و خروج داده ها در رابط های گرافیکی

زمانی که شما یک سیستم فازی را در یک فایل ذخیره می کنید. اطلاعات در قالب یک فایل متنی ASCII و با پسوند .fis ذخیره می شوند. این فایل متنی به راحتی قابل ویرایش بوده و بررسی آن ساده می باشد. زمانی که شما

سیستم فازی را در فضای کاری MATLAB ذخیره می‌کنید، مشخصات سیستم در قالب یک ساختار FIS ذخیره می‌شود. در واقع می‌توان گفت فایل و ساختار FIS دارای عملکرد مشابهی می‌باشند. فقط باید توجه داشته باشید که با ذخیره سازی سیستم در فضای کاری و عدم ذخیره سازی آن در قالب فایل، با بستن و اجرای دوباره نرم افزار MATLAB امکان بازیابی سیستم وجود نخواهد داشت.

برنامه نویسی در محیط m.file نرم افزار MATLAB

```
Clear
Clc
newfis('Laundromat-7');
a = readfis('Laundromat-7.fis');
fuzzy(a);
mfedit(a);
a.input(1).name='Dirty';
a.input(1).range=[0 100];
a.input(1).mf(1).name='D1';
a.input(1).mf(1).type='gaussmf';
a.input(1).mf(1).params=[4.268 9.975];
a.input(1).mf(2).name='D2';
a.input(1).mf(2).type='gaussmf';
a.input(1).mf(2).params=[4.289 29.95];
a.input(1).mf(3).name='D3';
a.input(1).mf(3).type='gaussmf';
a.input(1).mf(3).params=[4.289 49.95];
a.input(1).mf(4).name='D4';
a.input(1).mf(4).type='gaussmf';
a.input(1).mf(4).params=[4.289 69.95];
a.input(1).mf(5).name='D5';
a.input(1).mf(5).type='gaussmf';
a.input(1).mf(5).params=[4.48 89.97];

a.input(2).name='Weight';
a.input(2).range=[0 7];
a.input(2).mf(1).name='W0';
a.input(2).mf(1).type='gaussmf';
```

```
a.input(2).mf(1).params=[0.2123 0];
a.input(2).mf(2).name='W1';
a.input(2).mf(2).type='gaussmf';
a.input(2).mf(2).params=[0.261 1.201];
a.input(2).mf(3).name='W2';
a.input(2).mf(3).type='gaussmf';
a.input(2).mf(3).params=[0.4671 2.638];
a.input(2).mf(4).name='W3';
a.input(2).mf(4).type='gaussmf';
a.input(2).mf(4).params=[0.4671 4.387];
a.input(2).mf(5).name='W4';
a.input(2).mf(5).type='gaussmf';
a.input(2).mf(5).params=[0.4671 6.137];

a.output(1).name='Time';
a.output(1).reng=[0 90];
a.output(1).mf(1).name='Error';
a.output(1).mf(1).type='trimf';
a.output(1).mf(1).params=[-0.5 0 0.5];
a.output(1).mf(2).name='T1';
a.output(1).mf(2).type='trimf';
a.output(1).mf(2).params=[0 5.625 12.25];
a.output(1).mf(3).name='T2';
a.output(1).mf(3).type='trimf';
a.output(1).mf(3).params=[11.25 16.8 23.5];
a.output(1).mf(4).name='T3';
a.output(1).mf(4).type='trimf';
a.output(1).mf(4).params=[22.5 28.125 34.75];
a.output(1).mf(5).name='T4';
a.output(1).mf(5).type='trimf';
a.output(1).mf(5).params=[33.75 39.375 46];
a.output(1).mf(6).name='T5';
a.output(1).mf(6).type='trimf';
a.output(1).mf(6).params=[45 50.625 57.25];
a.output(1).mf(7).name='T6';
a.output(1).mf(7).type='trimf';
a.output(1).mf(7).params=[56.25 61.875 68.5];
a.output(1).mf(8).name='T7';
a.output(1).mf(8).type='trimf';
a.output(1).mf(8).params=[67.5 73.125 79.75];
a.output(1).mf(9).name='T8';
```



```
a.output(1).mf(9).type='trimf';  
a.output(1).mf(9).params=[78.75 84.375 91];
```

```
a.output(2).name='Flour';  
a.output(2).reng=[0 400];  
a.output(2).mf(1).name='Error';  
a.output(2).mf(1).type='trimf';  
a.output(2).mf(1).params=[-0.5 0 0.5];  
a.output(2).mf(2).name='F1';  
a.output(2).mf(2).type='trimf';  
a.output(2).mf(2).params=[0 25 51];  
a.output(2).mf(3).name='F2';  
a.output(2).mf(3).type='trimf';  
a.output(2).mf(3).params=[50 75 101];  
a.output(2).mf(4).name='F3';  
a.output(2).mf(4).type='trimf';  
a.output(2).mf(4).params=[100 125 151];  
a.output(2).mf(5).name='F4';  
a.output(2).mf(5).type='trimf';  
a.output(2).mf(5).params=[150 175 201];  
a.output(2).mf(6).name='F5';  
a.output(2).mf(6).type='trimf';  
a.output(2).mf(6).params=[200 225 251];  
a.output(2).mf(7).name='F6';  
a.output(2).mf(7).type='trimf';  
a.output(2).mf(7).params=[250 275 301];  
a.output(2).mf(8).name='F7';  
a.output(2).mf(8).type='trimf';  
a.output(2).mf(8).params=[300 325 351];  
a.output(2).mf(9).name='F8';  
a.output(2).mf(9).type='trimf';  
a.output(2).mf(9).params=[350 375 401];
```

```
a.rule(1).antecedent=[1 1];  
a.rule(1).consequent=[1 1];  
a.rule(1).weight=1;  
a.rule(1).connection=1;  
a.rule(2).antecedent=[1 2];  
a.rule(2).consequent=[2 2];  
a.rule(2).weight=1;  
a.rule(2).connection=1;
```

```
a.rule(3).antecedent=[1 3];
a.rule(3).consequent=[3 3];
a.rule(3).weight=1;
a.rule(3).connection=1;
a.rule(4).antecedent=[1 4];
a.rule(4).consequent=[4 4];
a.rule(4).weight=1;
a.rule(4).connection=1;
a.rule(5).antecedent=[1 5];
a.rule(5).consequent=[5 5];
a.rule(5).weight=1;
a.rule(5).connection=1;
a.rule(6).antecedent=[2 1];
a.rule(6).consequent=[1 1];
a.rule(6).weight=1;
a.rule(6).connection=1;
a.rule(7).antecedent=[2 2];
a.rule(7).consequent=[3 3];
a.rule(7).weight=1;
a.rule(7).connection=1;
a.rule(8).antecedent=[2 3];
a.rule(8).consequent=[4 4];
a.rule(8).weight=1;
a.rule(8).connection=1;
a.rule(9).antecedent=[2 4];
a.rule(9).consequent=[5 5];
a.rule(9).weight=1;
a.rule(9).connection=1;
a.rule(10).antecedent=[2 5];
a.rule(10).consequent=[6 6];
a.rule(10).weight=1;
a.rule(10).connection=1;
a.rule(11).antecedent=[3 1];
a.rule(11).consequent=[1 1];
a.rule(11).weight=1;
a.rule(11).connection=1;
a.rule(12).antecedent=[3 2];
a.rule(12).consequent=[4 4];
a.rule(12).weight=1;
a.rule(12).connection=1;
a.rule(13).antecedent=[3 3];
```

```
a.rule(13).consequent=[5 5];
a.rule(13).weight=1;
a.rule(13).connection=1;
a.rule(14).antecedent=[3 4];
a.rule(14).consequent=[6 6];
a.rule(14).weight=1;
a.rule(14).connection=1;
a.rule(15).antecedent=[3 5];
a.rule(15).consequent=[7 7];
a.rule(15).weight=1;
a.rule(15).connection=1;
a.rule(16).antecedent=[4 1];
a.rule(16).consequent=[1 1];
a.rule(16).weight=1;
a.rule(16).connection=1;
a.rule(17).antecedent=[4 2];
a.rule(17).consequent=[5 5];
a.rule(17).weight=1;
a.rule(17).connection=1;
a.rule(18).antecedent=[4 3];
a.rule(18).consequent=[6 6];
a.rule(18).weight=1;
a.rule(18).connection=1;
a.rule(19).antecedent=[4 4];
a.rule(19).consequent=[7 7];
a.rule(19).weight=1;
a.rule(19).connection=1;
a.rule(20).antecedent=[4 5];
a.rule(20).consequent=[8 8];
a.rule(20).weight=1;
a.rule(20).connection=1;
a.rule(21).antecedent=[5 1];
a.rule(21).consequent=[1 1];
a.rule(21).weight=1;
a.rule(21).connection=1;
a.rule(22).antecedent=[5 2];
a.rule(22).consequent=[6 6];
a.rule(22).weight=1;
a.rule(22).connection=1;
a.rule(23).antecedent=[5 3];
a.rule(23).consequent=[7 7];
```

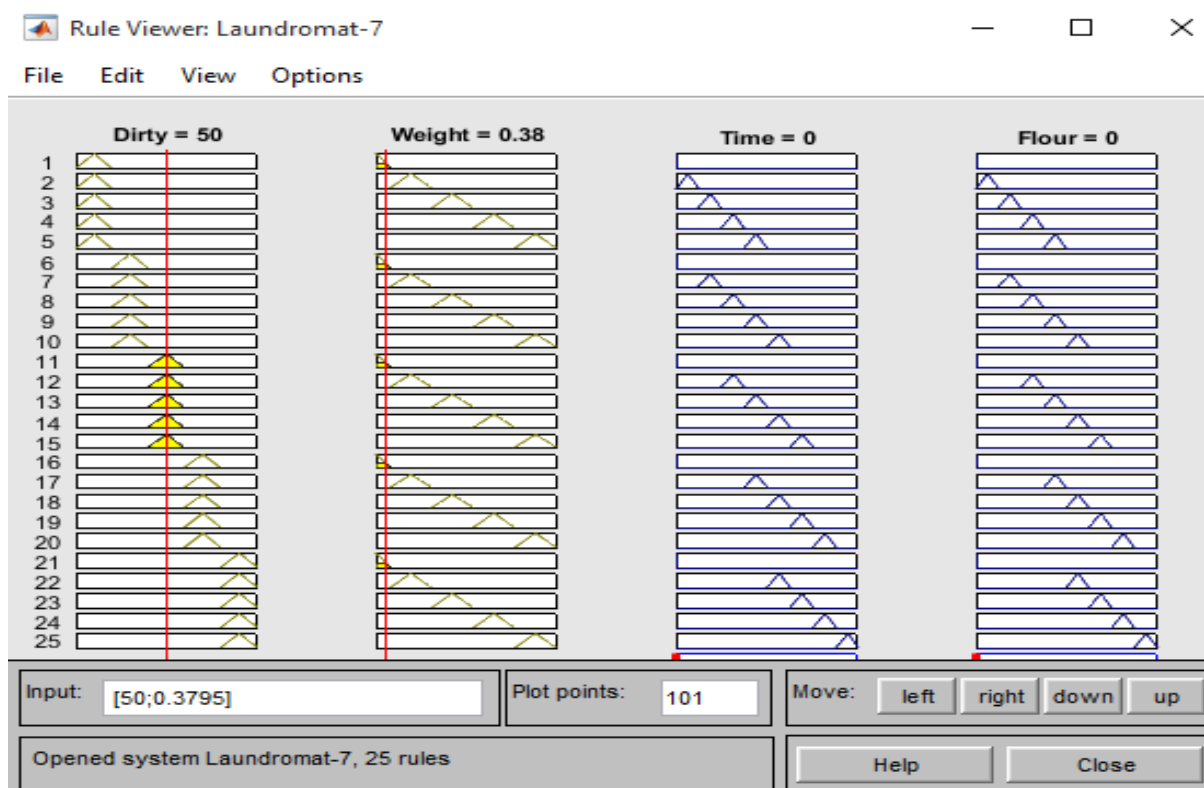
```
a.rule(23).weight=1;  
a.rule(23).connection=1;  
a.rule(24).antecedent=[5 4];  
a.rule(24).consequent=[8 8];  
a.rule(24).weight=1;  
a.rule(24).connection=1;  
a.rule(25).antecedent=[5 5];  
a.rule(25).consequent=[9 9];  
a.rule(25).weight=1;  
a.rule(25).connection=1;
```

چند مثال از پروژه ماشین شستشو فازی

مثال اول

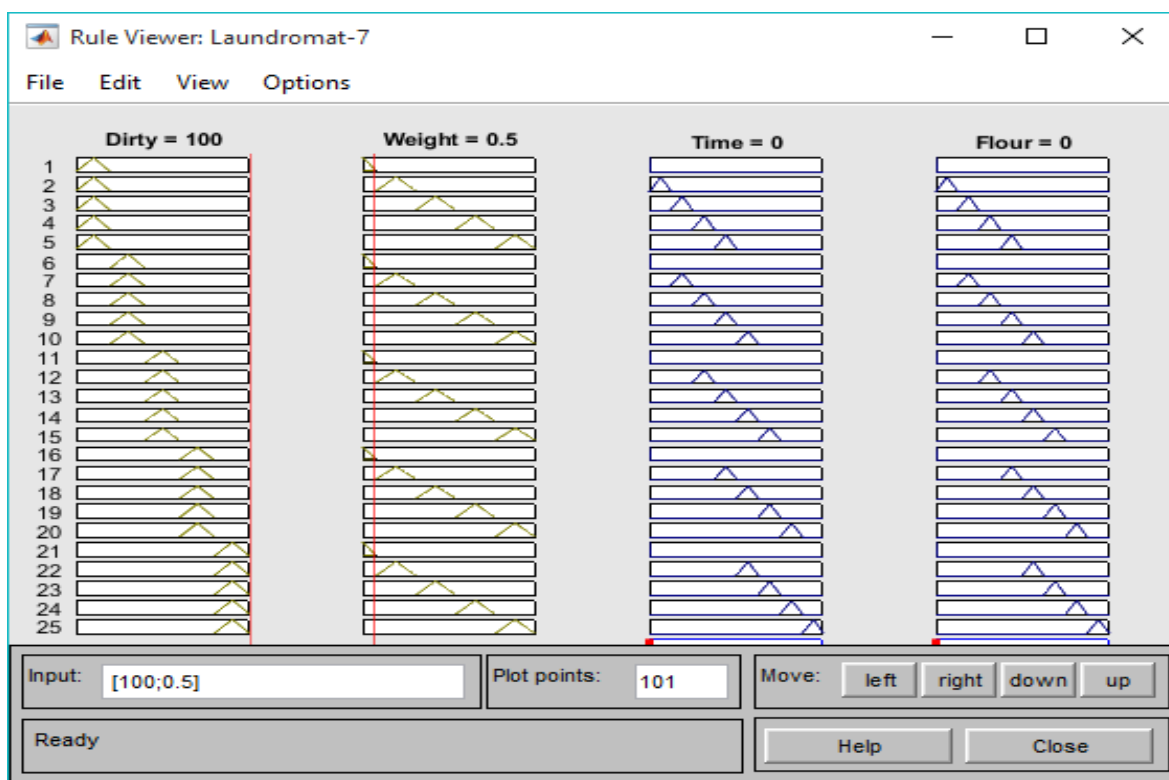
زمانی که سنسور بار میزان وزن لباس‌ها را کمتر از نیم کیلو گرم تشخیص داده است و میزان اندازه گیری شده

چرک لباس‌ها توسط سنسور نوری برابر ۵۰ درصد است.



شکل ۲-۲۰: نمایشگر قواعد مثال اول (الف)

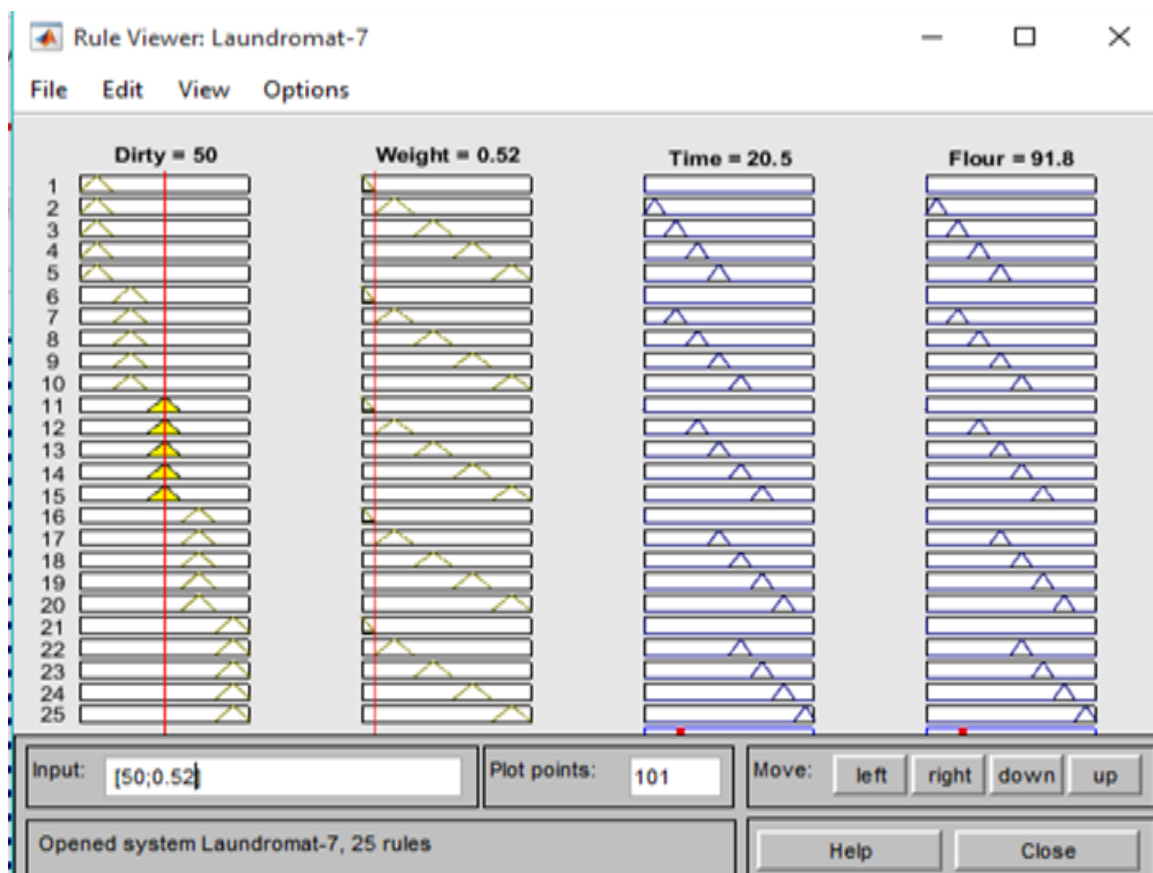
همانگونه که از قبل نیز انتظار داشتیم؛ در این حالت (زمانی که وزن لباس‌ها زیر نیم کیلوگرم است) اصلاً از لحاظ اقتصادی به صرفه نیست که ماشین شستشو روشن شود در این حالت ماشین لباسشویی Error کاهش وزن می‌دهد. و همانگونه که مشاهده می‌کنید در این حالت زمان شستشو برابر صفر و میزان پودر برابر صفر گرم است. توجه شما را به مثالی دیگر جلب می‌کنم. در این مثال سنسور بار میزان وزن لباس‌ها را نیم کیلو گرم تشخیص داده است و میزان اندازه گیری شده چرک لباس‌ها توسط سنسور نوری برابر ۱۰۰ درصد است.



شکل ۲-۲۱: نمایشگر قواعد مثال اول (ب)

همانگونه که مشاهده می‌کنید؛ در این حالت زمان شستشو برابر صفر و میزان پودر برابر صفر گرم است. چراکه خود عدد نیم کیلو گرم از این قاعده مستثنی نبوده و طبق تعریف از لحاظ اقتصادی و مصرف انرژی اصلاً به صرفه نیست که ماشین شستشو فازی روشن شود.

در همین رابطه توجه شما را به مثالی دیگر جلب میکنم. در این مثال سنسور بار میزان وزن لباسها را کمی بیشتر از نیم کیلو گرم (5.02) تشخیص داده است و میزان اندازه گیری شده چرک لباسها توسط سنسور نوری برابر ۵۰ درصد است. همانطور که در شکل زیر مشاهده می کنید؛ ماشین شستشو شروع به کار می کند در این حالت زمان شستشو برابر ۲۰ دقیقه و میزان پودر برابر ۹۲ گرم است.

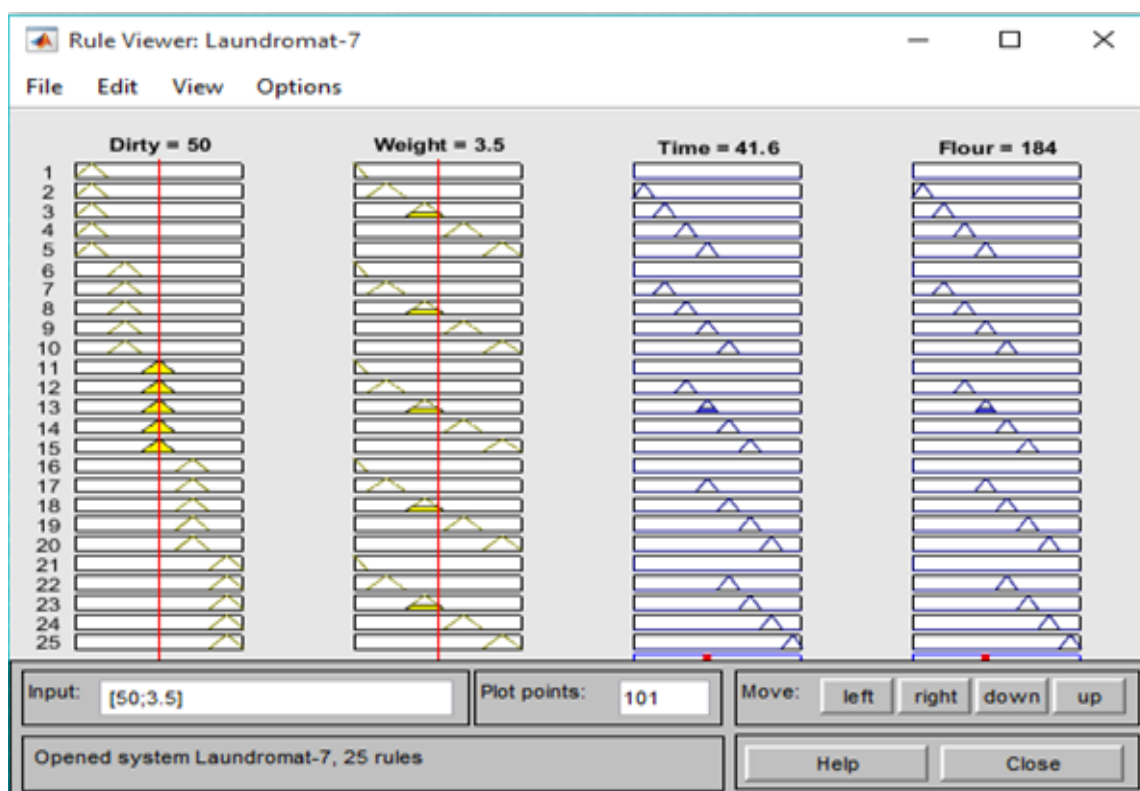


شکل ۲-۲۲: نمایشگر قواعد مثال اول (ج)

مثال دوم

زمانی که سنسور بار میزان وزن لباسها ۳,۵ کیلوگرم تشخیص داده است و میزان اندازه گیری شده چرک لباسها توسط سنسور نوری برابر ۵۰ درصد است.

در این حالت (زمانی که میزان چرک و وزن لباس‌ها برابر با ۵۰ درصد مقدار تعریف شده است) همانگونه که از قبل نیز انتظار داریم؛ در این حالت باید زمان شستشو برابر ۴۵ دقیقه و میزان پودر برابر ۲۰۰ گرم باشد. اما همانطور که در شکل زیر مشاهده می‌کنید؛ زمان و میزان پودر شستشویی که در این حالت توسط سیستم تعریف شده است به ترتیب برابر است با: ۴۱٫۶ دقیقه و ۱۸۴ گرم است و این کمی دور از انتظار ماست. در این باره می‌توان گفت که سیستم طراحی شده مقدار کمی خطا دارد که می‌توان از آن چشم‌پوشی کرد.



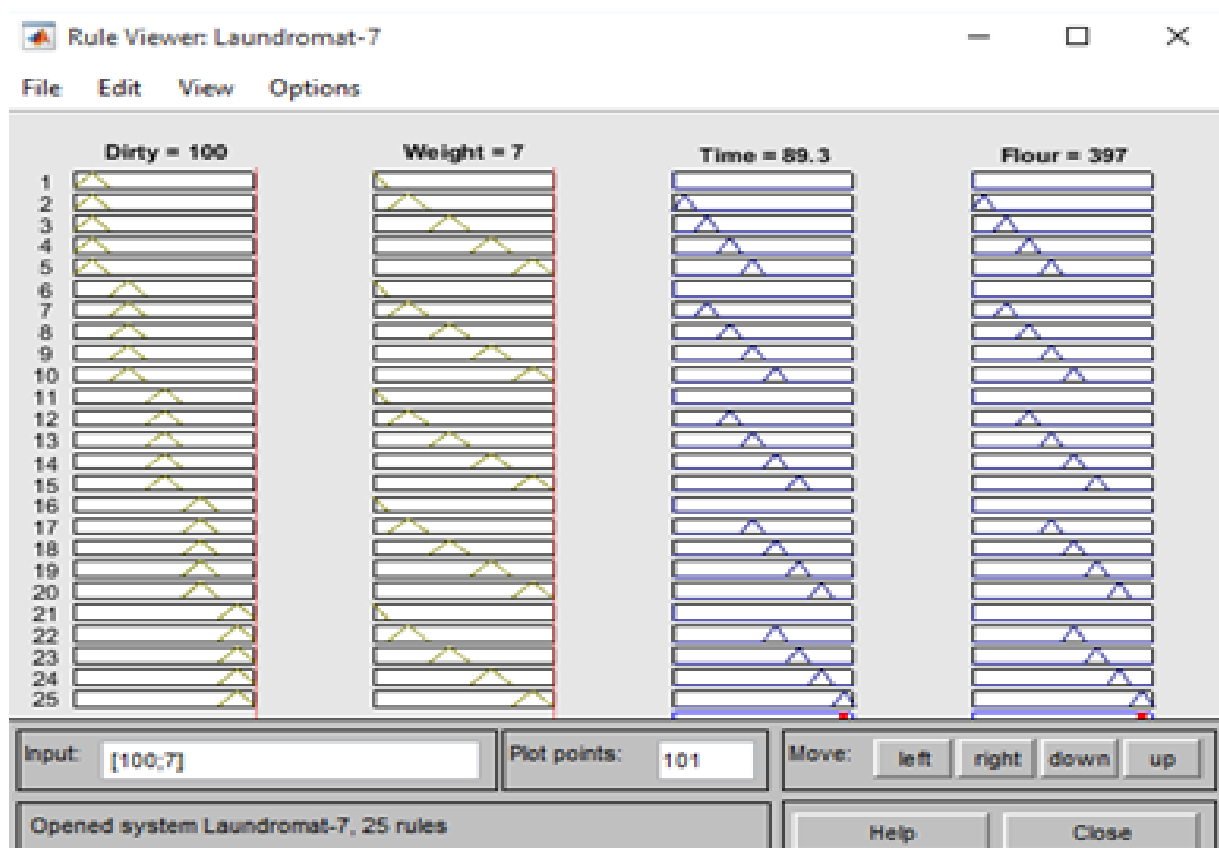
شکل ۲-۲۳: نمایشگر قواعد مثال دوم

مثال سوم

زمانی که سنسور بار میزان وزن لباس‌ها ۷ کیلو گرم تشخیص داده است و میزان اندازه گیری شده چرک لباس‌ها توسط سنسور نوری برابر ۱۰۰ درصد است.

در این حالت (زمانی که میزان چرک و وزن لباس‌ها برابر با ۱۰۰ درصد مقدار تعریف شده است) همانگونه که از قبل نیز انتظار داریم؛ در این حالت باید زمان شستشو برابر ۹۰ دقیقه و میزان پودر برابر ۴۰۰ گرم باشد. اما

همانطور که در شکل زیر مشاهده می‌کنید؛ زمان و میزان پودر شستشویی که در این حالت توسط سیستم تعریف شده است به ترتیب برابر است با: ۸۹,۳ دقیقه و ۳۹۷ گرم است. در این باره می‌توان گفت که سیستم طراحی شده مقدار خیلی کمی خطا دارد که می‌توان از آن چشم‌پوشی کرد.



شکل ۲-۲۴: نمایشگر قواعد مثال سوم

فهرست مراجع

[۱] کیا "سید مصطفی" منطق فازی در MATLAB, انتشارات کیان رایانه سبز، صفحه ۱۸۱-۲۰۲، ۱۳۹۰.

[۲] میرهادی "عبدالرضا" مقاله سیستم های فازی و کاربرد آن در پزشکی، ۱۳۸۶.

[۳] "ابراهیمیان" مقاله کنترل دمای گلخانه توت فرنگی با منطق فازی و میکروکنترلرهای AVR ، ۱۳۹۶.

دوست عزیز من، این فایل از وبسایت **Melce.IR** دانلود شده، برای دانلود بیشتر به وبسایت **Melce.IR** مراجعه کنید.

