

بسم الله الرحمن الرحيم



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

گزارشکار پروژه ۲ درس هوش محاسباتی

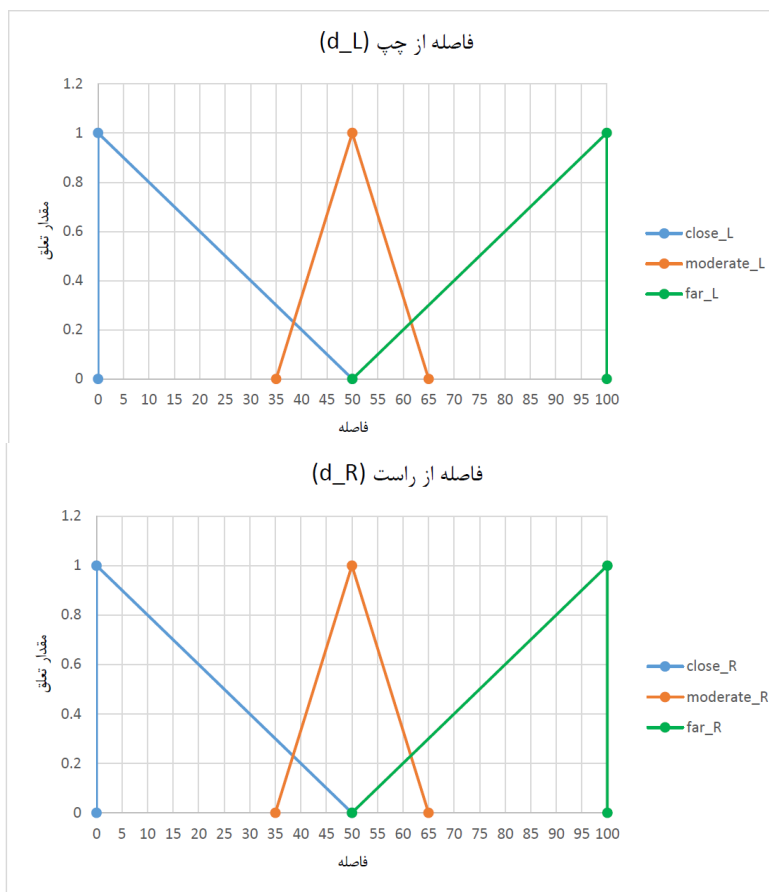
استاد عبادزاده

سید امیرمهدی میرشریفی

۹۸۳۱۱۰۵

فاز ۱

در این فاز سه بخش فازی سازی ، محاسبه مقدار فازی چرخش فرمان و غیر فازی سازی مقدار مرحله قبل را پیاده سازی کرده ایم . در مرحله یک مقادیر فاصله راست و چپ را دریافت میکنیم و مطابق با میزان تعلق که برای هر کدام به صورت نمودار داده شده است آن ها را مقدار دهی میکنیم . این کار به وسیله به دست آوردن معادله خط نمودار های زیر انجام میشود:



در ادامه در دو تابع `membership_right_dist` و `membership_left_dist` مقدار تعلق فازی را به ازای مقدار فاصله ای که داده میشود و با توجه به معادله خط های شکل بالا به دست می آوریم.

در مرحله بعد می بایست قوانینی که بر اساس ورودی های فازی فعال شده اند را شناسایی کنیم. قوانین در فایل `rules.txt` موجود هستند و از آنها در تابع `get_active_rules` که ورودی آن مقدار تعلقات در مرحله قبلی است ، استفاده میکنیم و مقدار

تعلق قوانین را بر میگردانیم. طبیعتاً قوانینی که مقدار تعلق آنها بیشتر از صفر باشد فعال شده اند. تصویر تابع در شکل زیر آمده است:

```
"""this function returns active rules based on rules and memberships """
def get_active_rules(self, membership_r, membership_l):
    membership_rotate = {}
    # rule 1 :IF (d_L IS close_L ) AND (d_R IS moderate_R) THEN Rotate IS low_right
    # rule 2 :IF (d_L IS cclose_L ) AND (d_R IS far_R) THEN Rotate IS high_right
    # rule 3: IF (d_L IS moderate_L ) AND (d_R IS close_R) THEN Rotate IS low_left
    # rule 4: IF (d_L IS far_L ) AND (d_R IS cclose_R) THEN Rotate IS high_left
    # rule 5: IF (d_L IS moderate_L ) AND (d_R IS moderate_R) THEN Rotate IS nothing
    membership_rotate["u_low_right"] = min(membership_l["close_L"], membership_r["moderate_R"])
    membership_rotate["u_high_right"] = min(membership_l["cclose_L"], membership_r["far_R"])
    membership_rotate["u_low_left"] = min(membership_l["moderate_L"], membership_r["cclose_R"])
    membership_rotate["u_high_left"] = min(membership_l["far_L"], membership_r["cclose_R"])
    membership_rotate["u_nothing"] = min(membership_l["moderate_L"], membership_r["moderate_R"])

    return membership_rotate
```

در ادامه با استفاده از قوانین فعال شده و سطح زیر نمودار آنها مقدار نهایی را می بایست به دست آوریم.

در این پروژه با به دست آوردن نقطه ثقل مقدار نهایی چرخش غیر فازی میشود. به این روش که با گام های مشخص شده از طرف کاربر مقادیر زیر نمودار در صورت و مخرجی که حاصل تقسیم آنها مقدار نهایی است ، اعمال میشود. این عملیات در تابع defuzz پیاده سازی شده است . همچنین ابتدا در این تابع سطح زیر نمودار به دست آمده با توجه به مقدار تعلقات و معادله خط های داده شده به دست می آید.

در نهایت مقدار نهایی این تابع به عنوان مقدار چرخش از طریق تابع تصمیم گیری برگردانده میشود.