



## هدایت روبات موبایل توسط کنترل کننده فازی با هدف مانع گریزی و هدف گرایی در محیط های پیچیده

آیدین تیهویی ، دکتر ایرج حسن زاده

دانشگاه تبریز - دانشکده ی مهندسی برق

E-mail: teyhooei@yahoo.com

چکیده - بسیاری از الگوریتم های ارائه شده در مورد هدایت روباتهای موبایل که بر پایه مدل محیطی می باشند، کارکرد مناسبی از خود در محیط های دینامیک نشان نمی دهند. در این مقاله استراتژی جدیدی بر پایه منطق فازی ارائه شده است. اطلاعات مربوط به محل هدف و همچنین اطلاعات حاصل از دریافت سنسورها از محل موانع توسط الگوریتم خاصی برای تولید زیر - هدف ها به کار می روند و منجر به پیمایش روبات در یکی از بهینه ترین مسیرها می شود. به دلیل سادگی و قابلیت اجرای *real-time*، از منطق فازی برای الگوریتم پیشنهادی استفاده شده است. این الگوریتم توسط نرم افزار *MATLAB* شبیه سازی شده و نتایج بدست آمده از قابلیت بالا و موثر بودن استراتژی به کار رفته حکایت می کند.

کلید واژه- روبات موبایل ، مسیریابی (path-planning) ، منطق فازی

### 1- مقدمه

time روبات با تکیه بر اطلاعات سنسورها و تنها اشکال آنها احتمال بالای گم شدن روبات به دلیل محدودیت سنسورها می باشد.

با توجه به موارد مطرح شده در هر دو استراتژی هدایتی، به نظر می رسد که توجه جدی به موارد ذیل برای توسعه الگوریتم های هدایتی روبات های موبایل لازم و ضروری است:

1- معمولا مدل ریاضی از محیط در دست نیست.

2- اطلاعات سنسوری به دلیل وجود نویز، غیر دقیق و غیر قابل اطمینان هستند.

3- هدایت *real-time* به دلیل دینامیک بودن محیط کاملا ضروری است.

بدین منظور، الگوریتم هایی بر پایه منطق فازی در طراحی کنترل کننده های مقاوم پیشنهاد شده اند که کارایی مناسبی در برخورد با نوسانات پارامترها و همچنین نویز از خود نشان داده اند. از طرف دیگر، به جهت سادگی ساخت و پیاده سازی، کنترل فازی برای روبات های موبایل کاملا

روشهای متداول هدایت روباتهای موبایل معروف به روشهای *model-based*، از یک مدل محیطی برای تولید مسیر استفاده می کنند. الگوریتم هایی چون الگوریتم میدان پتانسیل [2] و گراف روئیت [3] از این گونه اند و در این میان روش دوم از سابقه ی طولانی تری برخوردار است. در این روش که نمونه ای از کاربرد مساله ی فروشنده دوره گرد در بحث هدایت روبات می باشد، سعی بر این است که از میان تمام مسیرهای ممکنه، کوتاهترین آنها بین دو نقطه شروع و پایان (هدف) انتخاب شود.

با وجود ویژگی های مناسب این الگوریتم، این روش نیز همانند دیگر روش های *model-based*، با بکار گرفته شدن در محیط های دینامیک، کارکرد مناسب خود را از دست می دهد. برای چنین محیط هایی که هرگونه اطلاع دقیق در مورد موقعیت موانع به دلیل پیچیدگی و غیرقابل پیش بینی بودن دینامیک آنها، کاملا محدود و غیرقابل اطمینان است، روشهایی که برپایه اطلاعات سنسوری عمل می کنند (*sensor-based*)، پاسخ های قابل قبول تری از خود نشان می دهند. مزیت عمده ی این روش ها، هدایت *real-*

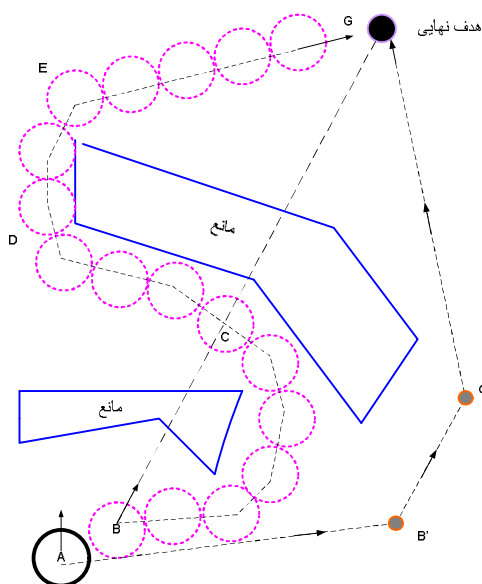
تولید می کند که من حیث برخورد نکردن با مانع، جهتی ایمن و از نقطه نظر هدف گرایی، جهتی قابل قبول و مناسب می باشد.

شایان ذکر است که علاوه بر قانون فازی مربوط به جهت گیری روبات، قانونی نیز تحت عنوان قانون فازی سرعت قابل بیان است که به جهت استفاده نکردن از این قانون در برنامه شبیه سازی شده، از ذکر آن در این مجال چشم می پوشیم. استفاده از این قانون می تواند حرکات نرم تری را برای روبات به ارمغان آورد.

### 3- استراتژی هدایت فازی هدف گرا

در الگوریتم فازی به کار رفته، موقعیت هدف که می تواند منجر به کوتاه شدن مسیر پیمایش روبات شود، به عنوان عاملی هدایتی جهت گم نشدن روبات به کار گرفته شده است. البته لازم به ذکر است که الگوریتم پیشنهادی هیچگونه تضمینی را از بابت طی شدن کوتاهترین مسیر به ما نمی دهد.

همچنان که در تصویر شماره (1) مشاهده می شود، می توان دو مسیر  $AB'C'G$  و  $ABCDEG$  را برای رسیدن به هدف متصور شد. به نظر مسیر دوم بهینه تراز اولی می باشد اما همچنان که گفته شد، به دلیل محدودیت سنسورها تضمینی از جهت حرکت روبات از مسیر دوم وجود ندارد.



شکل 1. رفتار غیر هدف گرا در مقایسه با رفتار هدف گرا

مناسب می باشد. بیش از دو دهه از اولین کاربردهای منطق فازی در هدایت و ناوبری سیستم ها در روبات های موبایل نمی گذرد اما در همین اندک مدت کاربرد های متنوعی از این تئوری ارائه گردیده که گام های بسیار موثری در نوع خود به شمار می روند.

در این مقاله، الگوریتمی با هدف مانع گریزی و هدف-گرایی بر مبنای منطق فازی جهت هدایت روبات موبایل پیشنهاد می شود. اطلاعات مربوط به موقعیت هدف و اطلاعات سنسوری از موقعیت موانع توسط کنترل کننده ترکیب و زیر-هدف های میانی جهت نیل به هدف نهایی تولید می گردند. مسیر بدست آمده توسط این روش شباهت زیادی به مسیر منتهی از الگوریتم گراف روئیت دارد، با این تفاوت که در روش ما هیچگونه فرض یا اطلاع قبلی در مورد محیط لازم نیست. این استراتژی هدایتی، در واقع، هیبریدی از روش های  $model-based$  و  $sensor-based$  می باشد.

این مقاله به شکل زیر ترتیب یافته است. در بخش 2، ما بطور خلاصه، مروری بر ایده های اساسی الگوریتم هدایتی فازی برای روبات های موبایل انجام می دهیم. بخش 3 به تشریح الگوریتم پیشنهادی پرداخته و در بخش 4، نتایج شبیه سازی این الگوریتم در محیط MATLAB نشان داده می شود. در بخش 5 نیز خلاصه ای از مزایای این الگوریتم را ارائه می کنیم.

### 2- منطق فازی در هدایت روبات موبایل

یک مجموعه ی فازی توسط رابطه ای ریاضی تحت عنوان تابع عضویت، توصیف می شود. این تابع هر عضو از مجموعه معین مانند  $X$  را به درون مجموعه ی  $[0,1]$  می نگارد.

قانون فازی برپایه جهت به شکل زیر قابل بیان است:

اگر جهت ممنوعه  $A$  و جهت مناسب  $B$  می باشد، آنگاه جهت چرخش  $C$  خواهد بود.

که در آن،  $A$ ،  $B$  و  $C$  همه با مجموعه های فازی بیان شده و  $C = (1-A)*B$ . (عملگر  $*$  یک عملگر  $t$ -norm می باشد که در تئوری مجموعه های فازی مطرح می گردد).

این شکل از قانون فازی، اطلاعات مربوط به مانع و هدف را با یکدیگر آمیخته و جهت مناسب برای چرخش روبات را

با مراجعه به تصویر شماره ی (2)، تمامی عباراتی که در این مقاله به کار رفته اند، به شکل زیر قابل بیان اند:

- زاویه هدف اختلاف زاویه ای بین جهت سر روبات و پرتویی است که بین مرکز روبات و هدف قرار دارد. این زاویه با  $f$  نشان داده شده است.

- زیر - هدف نقطه ای است میانی بر روی مسیر که با  $(x_w, y_w, g)$  نمایش داده می شود. عبارت  $g$  زاویه نقطه میانی نسبت به سر روبات می باشد. این زاویه خروجی کنترل کننده فازی ما می باشد.

- زاویه هدایتی،  $q$ ، زاویه چرخش روبات می باشد. در واقع این زاویه، زاویه سر روبات با محور  $X$  از چارچوب کلی سیستم است که متصل به محیط می باشد.

موقعیت توسط دو فاکتور مکان و جهت گیری روبات در چارچوب XOY تعیین می شود. موقعیت در لحظه  $t_i$  با  $(x_i, y_i, q_i)$ ، موقعیت ابتدایی با  $(x_o, y_o, q_o)$ ، موقعیت نقطه میانی با  $(x_w, y_w, q_w)$  و سرانجام هدف نهایی با  $(x_g, y_g, q_g)$  توصیف می شوند.

تمام علامت های مربوط به زوایا مثبت اند اگر نسبت به سر روبات در سمت راست قرار گیرند و منفی اند اگر در سمت چپ باشند. از روی شکل (2) می توان نوشت:

$$x_w = x_i + r \cos(q_i - g) \quad (1)$$

$$y_w = y_i + r \sin(q_i - g) \quad (2)$$

$$q_w = q_i - g \quad (3)$$

که منظور از  $r$  فاصله ای است که روبات باید طی کند تا به نقطه ی میانی بعدی برسد و به شکل زیر به دست می آید:

$$r = L + d$$

$L$  فاصله بین روبات و نزدیکترین مانع به نقطه میانی است، که توسط سنسور تشخیص داده شده است.  $d$  نیز آفستی است که به اندازه روبات بستگی دارد. در برنامه شبیه سازی شده، ما مقدار ثابتی را برای  $r$  در نظر گرفته ایم.

در واقع با یک trade-off مواجه می باشیم. اگر بخواهیم طبق شکل (1) مسیر بهینه انتخاب شود، نیاز داریم که از سنسورهایی با برد بلندتر استفاده نماییم اما از طرفی این امر موجب کاهش حساسیت روبات در تعامل با موانع نزدیک به آن می گردد.

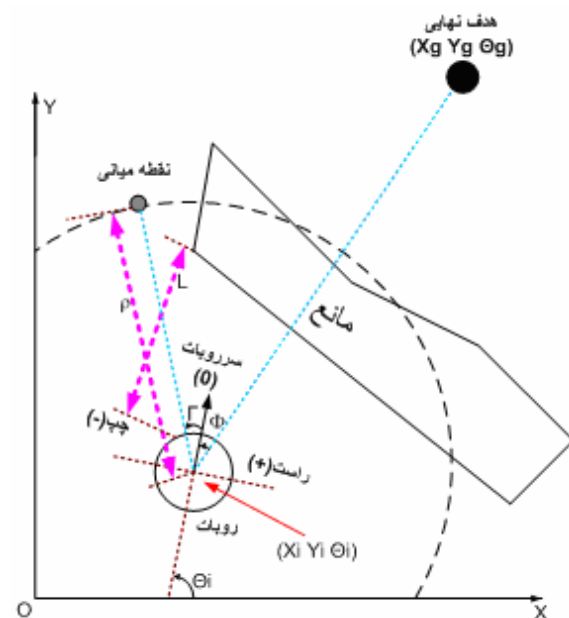
در این مقاله، ما مسئله هدایت روبات موبایل را در محیطی بسته و دو بعدی و مطرح می کنیم اما در صورتی که روبات با سنسورهای مناسب تجهیز شود، این استراتژی در محیط های غیر بسته نیز قابل اجراست. همچنین، شبیه سازی در MATLAB برای محیطی غیر دینامیک بوده است.

### 3-1- اصطلاحات کلیدی

از آنجا که در بحث کنترل روبات موبایل معمولاً از کامپیوتر استفاده می شود، لذا مناسب است که مجموعه ای گسسته مانند  $X$  به شکل زیر تعریف نمائیم:

$$X: [-180, 180]$$

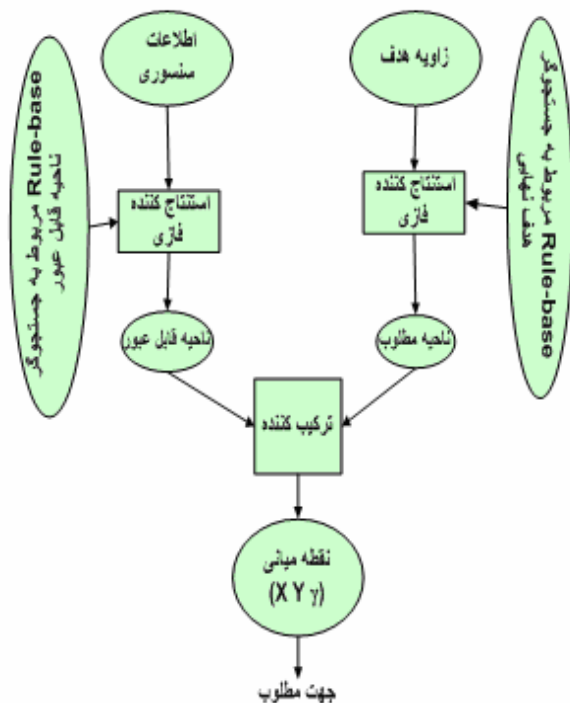
همچنان که می توان دید، توصیف جهت بدان شکلی که در الگوریتم ما به کار رفته است، در تصویر شماره ی (2) آشکار است. سر روبات همواره در جهت صفر درجه گرفته می شود، با منفی گرفتن جهت اگر چرخش به سمت چپ صورت گیرد و مثبت اگر به سمت راست باشد.



شکل 2. توصیف اصطلاحات به کار رفته

### 2-3- الگوریتم به کار رفته در کنترل کننده

کنترل کننده از سه واحد به نام های جستجوگر منطقه قابل عبور، جستجوگر هدف نهایی و ترکیب کننده دستور تشکیل یافته است. در شکل (3) می توان ترکیب کنترل کننده را مشاهده کرد.

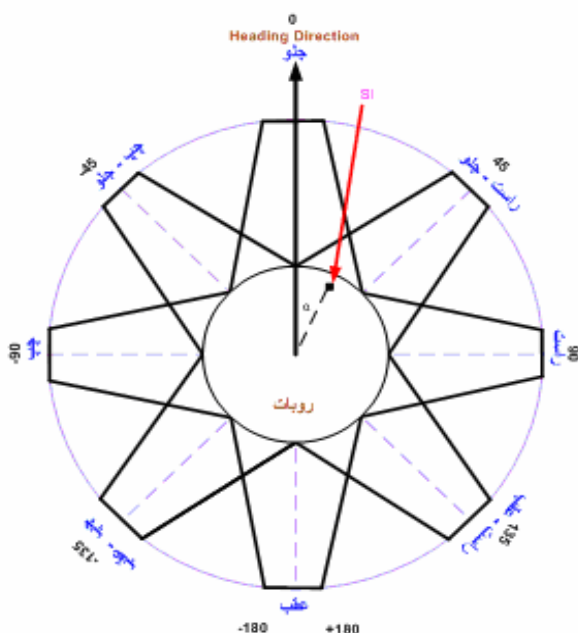


شکل 3. ترکیب کنترل کننده

لازم است که روبات با سنسورهای ultrasonic تجهیز شده باشد. ما در این پروژه از هشت سنسور استفاده کرده ایم که با شروع از زاویه 5 درجه نسبت به سر روبات و با فاصله های 45 درجه ای از هم چیده شده اند. اطلاعات رسیده از سنسورها وارد واحد جستجوگرمانع شده و بر اساس قوانین فازی موجود در Rule-base این واحد، منطقه عاری از مانع مشخص شود. واحد جستجوگر هدف نیز زاویه هدف،  $f$  را به عنوان ورودی گرفته و با توجه به قوانین Rule-base خود، منطقه هدایت کننده روبات به سمت هدف را معین می کند. دست آخرین دو ناحیه که توسط مجموعه های فازی توصیف شده اند، در واحد ترکیب کننده دستور، اشتراک گیری شده و ناحیه ای که هر دو خاصیت مانع گریزی و هدف گرایی روبات در آن ارضا می شود، بدست آمده و از روی آن خروجی این واحد که زاویه  $g$  می باشد، حاصل می شود. با استفاده از این خروجی در رابطه (3) و ترکیب آن با روابط (1) و (2)، موقعیت نقطه میانی  $(x_w, y_w, q_w)$

تعیین می گردد. از آنجا که توابع خطی و تکه ای در رایانه می توانند سریعتر و کارآمدتر مورد تجزیه و تحلیل قرار گیرند، لذا ما در سیستم هدایتی خود از توابع دوزنقه ای و مثلثی استفاده کرده ایم. Rule-base واحد جستجوگر مانع از هشت مجموعه فازی که معادل اصطلاحات زبانی {جلو، راست-جلو، راست، راست-عقب، چپ-جلو، چپ، چپ-عقب و عقب} هستند و در شکل (4) مشاهده می شوند، تشکیل یافته است.

شکل عمومی قانون فازی در اینجا به شکل زیر است: اگر  $s_i$  آتش شده است، آنگاه ناحیه  $i$  ممنوعه  $t_i$  خواهد بود.



شکل 4. Rule-base مربوط به واحد جستجوگر مانع

همچنان که از شکل (4) پیداست، منظور از  $s_i$ ،  $i$  امین سنسور بر روی روبات است که تحت زاویه  $a$  نسبت به سر آن قرار دارد و  $t_i$  مجموعه ای است فازی که بیانگر ناحیه ممنوعه می باشد. نحوه استدلال در این واحد در شکل (5) نشان داده شده است. توابع دوزنقه ای سیاه رنگ و نقطه چین، نمایشی از هشت تابع فازی شکل (4) می باشند. اگر سنسور  $i$  ام با زاویه  $a$  آتش شود، توابع راست-جلو و راست بترتیب با مقادیر  $m_1$  و  $m_2$  که در شکل با رنگ آبی مشخص اند، تحریک می شوند. سپس توسط عملگر  $\oplus$  و طبق رابطه ی زیر این دو تابع با یکدیگر جمع و ناحیه  $t_i$  با رنگ قرمز در شکل حاصل می گردد:

در اینجا نیز  $m_1, m_2$  میزان آتش شدن دو تابع فازی مجاور می باشند و این دو تابع در شکل (6) با رنگ آبی دیده می شوند.

حال با مینیمم گیری از دو ناحیه ی  $\Gamma$  و  $\Omega$  که نوعی اشتراک گیری در منطق فازی به حساب می آید، ناحیه  $\tilde{g}$  حاصل می شود که برای روبات هم بی خطر از نظر مانع و هم مناسب از جهت میل به هدف است.

$$\tilde{g} = \min\{\Gamma, \Omega\}$$

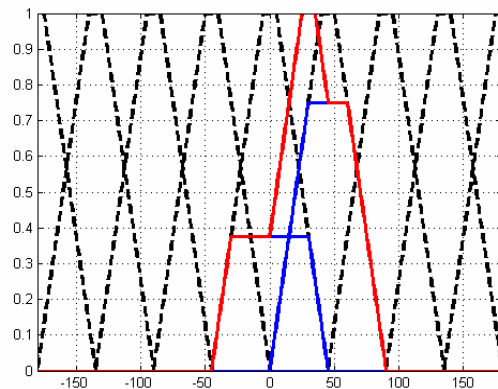
$\tilde{g}$  مجموعه ای فازی محسوب می شود که باید جهت دستیابی به مقدار مطلق  $g$  عمل فازی زدایی بر روی آن انجام گیرد. روش های متنوعی چون centroid، mom و... در بحث فازی زدایی مطرح می شوند که مجالی برای بحث پیرامون فلسفه وجودی این روش ها نیست. ما در این مقاله از روش mom (میانگین ماکسیمم ها) جهت فازی زدایی استفاده کرده ایم که نتایج قابل قبولی از خود داشته است.

#### 4- نتیجه گیری

آنچه که مسلم است، به دلیل سهولت پیاده سازی، این الگوریتم می تواند جایگزینی مناسب برای روش های سنتی پیشین باشد اما باید توجه داشت که می توان گام های بسیاری را در جهت بهینه ساختن آن برداشت. پیدا کردن روشی مناسب جهت خارج ساختن روبات از موانع U شکل و بن بست ها و رفع وقفه (deadlock) از چنین اقداماتی است که هم اکنون توجه ما را به خود معطوف نموده است.

#### مراجع

- [1] X. Yang, M. Moallem, and R. V. Patal, "A layered goal-oriented fuzzy motion planning strategy for mobile robot navigation" in *IEEE Trans. Syst., Man, Cybern. B, Cybern.* vol. 35, no. 6, pp.1214-1224, Dec. 2005
- [2] J. Latombe, *Robot Motion planning*. Norwell, MA: Kluwer, 1991.
- [3] J. Mitchel, "Amobile automation: An application of artificial intelligence techniques," in *Proc. 1<sup>st</sup> Int. Joint Conf. Artificial Intelligence*, Washington, DC, 1969, pp.509-520



شکل 5. ناحیه ممنوعه

$$t_i = m_1 \oplus m_2 = \min\{1, m_1 + m_2\}$$

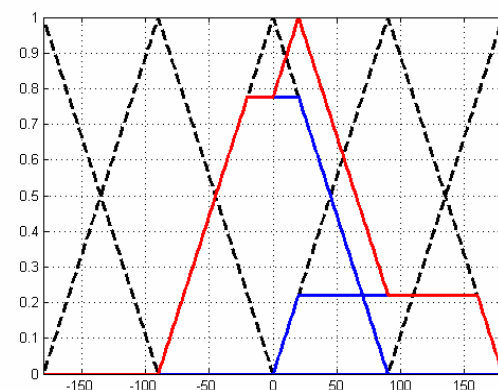
به محض آنکه تمام نواحی ممنوعه توسط هر سنسور تعیین شد، ناحیه قابل عبور و بدون مانع  $\Gamma$  از رابطه زیر به دست می آید:

$$\Gamma = 1 - \sum_{i=1}^n \max\{t_i\}$$

که  $n$  تعداد سنسورهای آتش شده است.

Rule-base مربوط به واحد جستجوگرهدف نیز از پنج مجموعه فازی با توابع مثلثی تشکیل شده است که در شکل (6) با رنگ سیاه مشخص اند. تحت زاویه ی هدف  $f$  این واحد، ناحیه مطلوب یعنی  $\Omega$  (رنگ قرمز در شکل) را که طبق رابطه ی زیر تعریف می شود، ایجاد می نماید:

$$\Omega = \min\{m_1 + m_2, 1\}$$



شکل 6. ناحیه هدف گرا