

## مسیریابی ربات با استفاده از یک روش تغییر یافته میدان نیروی مجازی، ضمن شناسائی و فرار از مینیم‌های محلی

صادق سلیمان پور، سعید شیرینی قیداری، حجت اله واهب

دانشگاه صنعتی امیر کبیر (پلی تکنیک تهران)، دانشکده مهندسی کامپیوتر و فناوری اطلاعات

E-mail: S.Soleimanpour@cic.aut.ac.ir

چکیده - یک راه حل مناسب برای یافتن یک مسیر مناسب برای ربات، استفاده از روش‌های میدان نیروی مجازی می‌باشد. مساله‌ای که در استفاده از این روش‌ها مطرح می‌شود، این است که امکان گیر کردن ربات در مینیم‌های محلی وجود دارد. در این حالت لازم است که ربات بتواند گیرافتادن در مینیم محلی را شناسائی و در ادامه برای رهائی از این موقعیت تلاش کند. راه حلی که برای رهائی ربات از این موقعیت ارائه شده به این صورت است که در مواردی که حرکت ربات در مدت مشخصی از یک حد آستانه‌ای کمتر باشد، گیر افتادن ربات در مینیم محلی شناسائی و با تلاش به دور کردن ربات از موانع سعی کنیم که موقتاً هدف را در نظر نگرفته و از مینیم رهائی پیدا کنیم. ایده‌ای که در این مقاله برای این مساله پیشنهاد شده به این صورت است که در مسیر حرکت ربات موانع کم ارتفاعی بصورت مجازی و در نقشه ربات قرار می‌دهیم و در مرورهای بعدی ربات از این نقاط ارتفاع این موانع را افزایش می‌دهیم. به این ترتیب حتی اگر ربات در مینیم‌های محلی گیر کند، خیلی سریع می‌تواند از این مینیم‌ها رهائی پیدا کند

کلید واژه- مسیریابی ربات- میدان نیروی مجازی - مینیم محلی

### ۱- مقدمه

ربات‌ها در طول مسیر خود، تنها می‌توانند با استفاده از حسگرهایشان موانع و رباتهایی که در محدوده دید آنها هستند، را تشخیص دهند، لذا از مسیر و مکان رباتهای دیگر و موانع موجود در محیط اطلاع کافی ندارند.

مسیریابی برای محیط شبه ایستا پیاده سازی شده است، هرچند که راه حل ارائه شده برای محیط پویا نیز قابل پیاده‌سازی است.

یکی از مسائل تحت بررسی دیگر در مسیریابی ربات‌ها مساله مینیم محلی است. این مساله شامل دو بخش است، بخش اول اینکه تشخیص داده شود ربات در مینیم محلی قرار گرفته است. و بخش دوم آن نیز این است که خروج از وضعیت بن بست بررسی شده و ربات در اولین فرصت ممکن به صورت عادی بسمت هدف خود حرکت کند.

در این طرح برای جلوگیری از قرار گرفتن ربات در مینیم محلی، در هنگام طی مسیر، در نقشه ربات، یک سری موانع

یکی از مسائلی که در کار با ربات‌های سیار مطرح است، مسیریابی ربات در محیط‌های پیچیده می‌باشد، اگر در محیط موانع متعدد وجود داشته باشد، ربات نمی‌تواند به خوبی مسیر خود را پیدا کند، این مشکل در صورتی که در محیط ربات‌های سیار دیگر و موانع متحرک وجود داشته باشد، بیشتر نمایان می‌شود. در این گزارش یک روش مسیریابی مبتنی بر میدان نیروی مجازی<sup>1</sup> برای رباتهای سیار در چنین محیط‌هایی ارائه کرده و با استفاده از راه حل‌های ارائه شده مشکلات ناشی از گیر افتادن ربات در مینیم‌های محلی را حل خواهیم نمود.

در این روش فرض بر این است که ربات‌ها در محیطی شروع به حرکت به سمت هدف می‌کنند که هیچ شناختی نسبت به آن ندارند.

<sup>1</sup> Potential field

با ارتفاع و شیب کم قرار می‌دهیم. به این ترتیب در صورتی که ربات در یک محل گیر کند، پس از مدتی، محل مورد نظر به عنوان یک مانع برای ربات مطرح شده و بصورت اتوماتیک از مینی‌مم محلی مورد نظر خارج خواهد شد.

همانطور که می‌دانید، یکی از روش‌هایی که برای مسیریابی استفاده می‌شود، روش میدان نیروی مجازی است. حسن این روش این است که یک مسیر نسبتاً بهینه، کوتاه و بدون برخورد به سمت هدف به ما می‌دهد. ولی این روش دارای معایبی نیز هست. از محدودیت‌هایی که می‌توان برای این روش پیش‌بینی کرد، عکس‌العمل این روش در مقابل موانع نعلی شکل است.

### ۱-۱- روشهای فرار از کمینه محلی

برای بررسی روشهای فرار از بن بست دو نکته زیر را بایستی در نظر گرفت:

تشخیص واقع شدن در کمینه محلی  
تشخیص خارج شدن از کمینه محلی

اول اینکه ربات تشخیص دهد که در این وضعیت قرار دارد و این تشخیص جامع باشد یعنی اینکه کلیه حالاتی که باعث پیش آمدن این وضعیت شده را شامل شود. دوم آنکه خروج از وضعیت کمینه تشخیص داده شود و در اولین فرصت ممکن به مسیری که به سمت هدف اصلی می‌باشد برگردانده شود و و دیگر وارد وضعیت قبل نشود.

روشهای موجود برای خروج از بن بست :

- تغییر الگوریتم ناوبری
- در نظر گرفتن یک هدف مجازی
- تغییر پارامترهای الگوریتم
- دنبال کردن دیوار
- تعریف موانع مجازی

### ۲- الگوریتم پیشنهادی

کارهای انجام شده در این مقاله را می‌توان به سه گروه اصلی تقسیم کرد:

مسیریابی برای هر ربات در جهت رسیدن به هدف با حداکثر سرعت و اجتناب از برخورد با موانع دوری از برخورد احتمالی بین ربات‌ها  
قرار نگرفتن در مینی‌مم محلی  
الگوریتم مربوط به مسیر یابی ارائه شده به صورت زیر می‌باشد:

**گام اول:** برای هر ربات نقطه شروع و پایان وهمینطور اولویت آن ربات مشخص می‌شود. در این روش یک شمارنده برای مقاصد مثل خروج از بن بست استفاده شده است، در این گام شمارنده صفر می‌شود. همچنین در اینجا یک شعاع دید بعنوان محدوده دید حسگری مشخص می‌شود.

**گام دوم:** ربات بررسی می‌کند که به هدف رسیده است یا خیر؟ اگر نرسیده بود به گام بعدی می‌رود. اگر هم رسیده بود پایان مسیر یابی را اعلام می‌کند.

**گام سوم:** در این گام ربات موانع را در محدوده دید حسگری خود می‌بیند.

**گام چهارم:** در این گام ربات مسیر پیموده شده قبلی را بعنوان مانع به مجموعه موانع مشاهده شده اضافه می‌کند، تا به این ترتیب بتواند از ماندن ربات در یک محل جلوگیری کند. اگر مسیر قبلاً پیموده شده است، ارتفاع مانع افزایش می‌یابد.

**گام پنجم:** در اینجا ربات بررسی می‌کند که آیا در محدوده دید حسگری برخوردی با رباتهای دیگر وجود دارد یا خیر؟ اگر وجود داشت، اولویتها را بررسی می‌کند اگر اولویت با او بود به مرحله بعدی می‌رود و در غیر اینصورت به گام قبلی بر میگردد. اگر برخورد وجود نداشت به مرحله بعد می‌رود.

**گام ششم:** در این گام مساله فرار از مینی‌مم محلی برای ربات‌ها بررسی و حل می‌شود.

**گام هفتم:** در این مرحله بمنظور پیدا کردن بهترین جهت طوری که ربات کمترین فاصله تا هدف و کمترین میزان برخورد با موانع را داشته باشد یک تابع هزینه وزندار مینی‌مم می‌شود. این تابع هزینه وزندار شامل دو قسمت

**گام نهم:** با توجه به سرعت ربات موقعیت ربات در جهت تعیین شده تغییر می کند. یعنی ربات با توجه به سرعت و جهتی که برای خود تعیین می کند حرکت کرده و به جای بعدی می رسد.

**گام دهم:** بعد از ثبت محل جدید ربات، برای اینکه در مراحل بعدی بعنوان مانع در مسیر ربات مطرح شود، یکی به مقدار شمارنده اضافه شده و به مرحله دوم بر می گردیم.

### ۳- کنترل فازی سرعت ربات

یک کنترل کننده فازی برای ربات به منظور جلوگیری از برخورد ربات با موانع در نظر گرفته شده است. این کنترل کننده فازی دو ورودی یکی میزان فاصله تا مانع و دیگری فاصله تا هدف و یک خروجی بعنوان سرعت دارد و از توابع عضویت مثلثی برای ورودی و خروجی آن استفاده شده است. همچنین از متوسط وزنی مراکز برای غیر فازی سازی و سیستم استنتاج ممدانی برای استنتاج استفاده نموده ایم.

کنترل کننده دیگری نیز به منظور افزایش سرعت ربات در رسیدن به هدف استفاده شده است. ورودیهای آن نیز میزان مانع و فاصله تا هدف و خروجی آن سرعت ربات است ساختار آن نیز مانند کنترل کننده فازی ارائه شده برای پرهیز از موانع بوده است.

### ۴- فرار از مینیمم محلی

قبل از بحث در مورد چگونگی مواجهه ما با این مساله در این پروژه به ذکر دو موضوع زیر می پردازیم.

تشخیص فرار گرفتن در مینیمم محلی

تلاش برای خارج شدن از مینیمم محلی

نحوه تشخیص فرار گرفتن در مینیمم محلی با توجه به تغییرات فاصله ربات نسبت به هدف و میزان فاصله تا مانع در جلو ربات می باشد. اگر تغییرات فاصله ربات تا هدف بطور مداوم در چند مرتبه از یک عدد معین که از سعی و خطا بدست می آید کمتر بود و میزان مانع نیز بطور مداوم در همان چند بار از یک عدد معینی که آن هم سعی و خطا بدست می آید بیشتر بود ربات در مینیمم محلی قرار گرفته است. با استفاده از یک کنترل کننده فازی نسبت وزنها بطوری عوض می شود که اهمیت دور شدن از مانع خیلی بیشتر از رسیدن به هدف باشد بعد از دور شدن ربات از مانع

است، یکی مربوط به فاصله از هدف و دیگری مربوط به میزان برخورد با موانع. تابع هزینه مربوط به میزان برخورد با موانع بصورت زیر است:

$$J_1 = \max [e^{-\alpha(x-x_{i1})^2}, \dots, e^{-\alpha(x-x_{in})^2}, e^{-\alpha(x-x_{i1})^2}, \dots, e^{-\alpha(x-x_{i+1})^2}] \quad (1-2)$$

$x_0, x_o, x_r$  بترتیب موقعیت رباتهای دیگر، موقعیت موانع و موقعیت فعلی ربات است.

همانطور که دیده می شود تابع هزینه مربوط به دوری از موانع به صورت نمایی انتخاب شده است. این انتخاب باعث می شود تا ربات نه زیاد از موانع دوری کند نه خیلی به موانع نزدیک شود. البته این فواصل بصورت سعی و خطا با  $\alpha$  قابل تنظیم است.

تابع هزینه مربوط به فاصله ربات تا هدف نیز به صورت زیر است:

$$J_g = (x - x_g)^2 \quad (2-2)$$

بطوری که  $x, x_g$  بترتیب موقعیت هدف و موقعیت فعلی ربات می باشند.

علاوه بر دو تابع هزینه فوق، یک تابع هزینه جدید نیز در نظر گرفته شده است تا بتوانیم ربات را به راحتی از گیر کردن در مینیمم محلی برحذر داریم. روشی که برای انجام این کار پیشنهاد شده است به این صورت است که در مسیری که ربات طی می کند، یک سری مانع با تابع هزینه ای به صورت زیر قرار می دهیم.

$$J_g = (x - x_g)^2 \quad (3-2)$$

این دو تابع هزینه را به صورت وزندار باهم جمع می کنیم و در هر راستا که مجموع وزندار آنها مینیمم شد ربات جهت را در آن سمت انتخاب می کند.

$$J = W_1 J_o + W_2 J_g \quad (4-2)$$

**گام هشتم:** حال با توجه به میزان مانعی که در راستای انتخاب شده وجود دارد و فاصله موجود تا هدف، سرعت ربات تنظیم می شود. در بخش بعدی به تفصیل توضیح داده می شود.

## ۵- شبیه سازی

### ۵-۱- طرح شبیه سازی

همانطوری که عنوان شد، ایده طرح فرض کردن مسیر حرکت ربات به عنوان مانع می‌باشد. اما اینکه شکل این موانع به چه صورتی باشد و از چه تابعی به عنوان مانع استفاده کنیم، خود یک مساله اساسی است.

ما برای انجام کار سه فرض اساسی را مطرح نموده‌ایم:

۱. مسیر ربات به عنوان یک مانع، مانند موانع موجود در محیط فرض شود.

۲. مسیر طی شده بوسیله ربات به عنوان یک مانع با ارتفاع کم و شیب ملایم در نظر گرفته شود.

برای هر کدام از موارد فوق، بایستی حالت‌های زیر مورد بررسی قرار بگیرد:

۱. طی مسیر عادی بوسیله ربات بدون برخورد با موانع ثابت

۲. طی مسیر بوسیله ربات در حالتی که در محیط موانع متحرک و احتمال برخورد با سایر ربات‌ها وجود دارد

۳. طی مسیرهای پیچیده و دارای بن‌بست

شبیه‌سازی‌های متعددی برای حالت‌های مختلف فوق انجام شده است. روش ارائه شده توانسته است که کارایی خود را برای محیط‌های با شرایط فوق به خوبی نشان دهد.

در شبیه‌سازی‌ها و نتایج ارائه شده، مفروضات عبارتند از:

- نقطه شروع با دایره نشان داده شده است
- هدف ربات‌ها با مربع نمایش داده شده اند.
- سرعت ربات بین ۱ تا ۱۰ متر بر ثانیه تغییر می‌کند
- محدوده دید هر ربات ۱ متر می‌باشد.

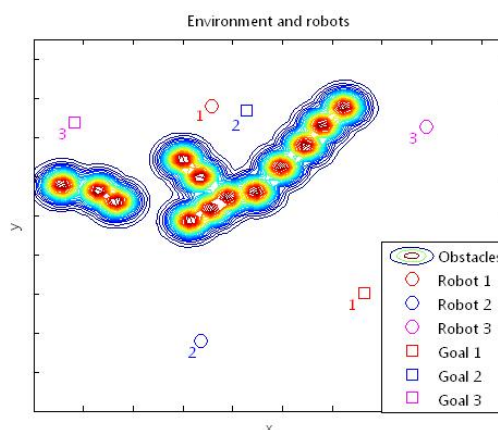
### ۵-۲- طی مسیر بدون برخورد با موانع ثابت

شکل زیر سه ربات را نشان می‌دهد که از مبدا شروع به حرکت می‌کند و پس از عبور کردن از موانع تعیین شده و با

وضعیت به حالت عادی بر می‌گردد و وزن‌ها به حالت گذشته بر می‌گردند. اگر دوباره به مینی‌مم محلی برگردد تغییر وزن‌ها بطور تصاعدی و بیشتر از گذشته خواهند بود. و این روند تا خروج ربات از مینی‌مم محلی ادامه خواهد داشت. اگر ربات دچار مینی‌مم محلی نبود به گام دوم بر می‌گردد.

بعد از خارج شدن از مینی‌مم محلی یکی به مقدار شمارنده اضافه شده و به مرحله دوم بر می‌گردیم. علاوه بر موارد فوق برای تکمیل کار و بهینه شدن حرکات ربات، در طول مسیری که ربات طی می‌کند، در نقشه ربات یک سری مانع با ارتفاع کم قرار داده می‌شود. با توجه به اینکه حضور ربات در یک محل باعث افزایش ارتفاع این موانع می‌شود، این عامل باعث می‌شود که ربات تمایلی به ماندن در محل قبلی نداشته باشد و لذا به راحتی از مینی‌مم محلی خارج می‌شود.

در شکل (۴-۱) یک نمونه دشوار از استقرار موانع نمایش داده شده است. با توجه به اینکه محل سه مانع سمت چپ به گونه‌ای است که باعث می‌شود ربات به درون حفره تمایل پیدا کرده و در بن‌بست مذکور گیر کند، این موانع برای ربات‌های بدون استفاده از روش‌های شناسایی و خروج از بن‌بست، یک مشکل اساسی محسوب می‌شوند.



شکل (۴-۱) یک نمونه دشوار از استقرار موانع

در این شکل، ربات‌ها با دایره و اهداف ربات‌ها با مربع نمایش داده شده است. موانع نیز با میدان‌های دفعی نزدیک هر کدام نمایش داده شده‌اند. در شبیه‌سازی انجام شده، هر دور کامل از گام دوم تا گام هشتم را یک تکرار نام‌گذاری می‌کنیم و فرض می‌کنیم که ربات در هر تکرار بسته به سرعتی که بخش تنظیم سرعت برای آن تعیین می‌کند، بین یک سانتیمتر تا یک متر حرکت می‌کند.

احساس می‌کند، اما به دلیل وجود مانع در سمت راستش (چپ تصویر)، مجبور شده است که مقداری از مسیر را موازی با دو ربات دیگر طی کند، به محض اینکه دو ربات از مسیر حرکتش کنار رفتند، ربات مسیر حرکتش را آزاد احساس می‌کند و بصورت مستقیم و خیلی سریع به سمت هدف حرکت می‌کند.

#### ۴-۵- خروج از بن بست با قوانین فازی

در روش پیشنهادی با توجه به تغییرات فاصله ربات نسبت به هدف و میزان مانع در جلو ربات، ربات تشخیص می‌دهد در مینیمم محلی قرار گرفته است. اگر تغییرات فاصله ربات تا هدف بطور مداوم در چند بار از یک عدد معین که از سعی و خطا بدست می‌آید کمتر بود و میزان مانع نیز بطور مداوم در همان چند بار از یک عدد معینی که آن هم از سعی و خطا بدست می‌آید بیشتر بود ربات در مینیمم محلی قرار گرفته است.

با استفاده از یک کنترل کننده فازی نسبت وزنها بطوری عوض می‌شود که اهمیت دور شدن از مانع خیلی بیشتر از رسیدن به هدف باشد بعد از دور شدن ربات از مانع و کم شدن میزان مانع وضعیت به حالت عادی بر می‌گردد و وزنها به حالت گذشته تغییر می‌کنند. و اگر دوباره به مینیمم محلی برگردد تغییر وزنها بطور تصاعدی و بیشتر از گذشته خواهند بود. و این روند تا خروج ربات از مینیمم محلی ادامه خواهد داشت.

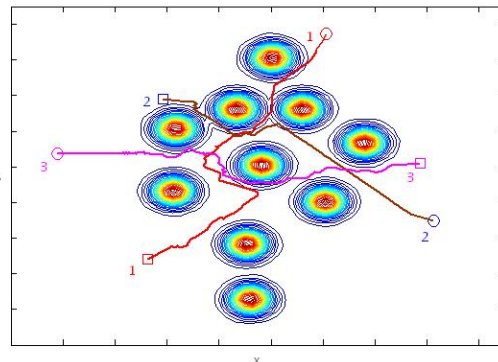
مواردی که در بالا ذکر شد، با استفاده از روش فرض کردن مسیر به عنوان مانع، ترکیب شده و ربات راحت‌تر و سریعتر می‌تواند از بن بست خارج شود

#### ۵-۵- طی مسیره‌های پیچیده و دارای بن بست

نقشه‌ای که برای این حالت در نظر گرفته شده است، به گونه‌ای است که احتمال قرار گرفتن ربات‌ها در بن بست وجود دارد (شکل (۵-۵)).

این مورد و شبیه‌سازی‌های مشابه دیگر نشان دادند که استفاده از این روش این مزیت را نسبت به کارهایی که سایرین انجام داده‌اند دارد که خیلی سریع از بن بست خارج می‌شود ولی معایب زیر باعث می‌شوند که نتوان مسیر ربات را به عنوان یک مانع با ارتفاع بلند در نظر گرفت:

در نظر گرفتن جلوگیری از برخورد با یکدیگر به سمت هدف می‌روند. با توجه به فرم چیدمان موانع در این مساله، مشکلات برخورد با موانع بوجود نیامده است.

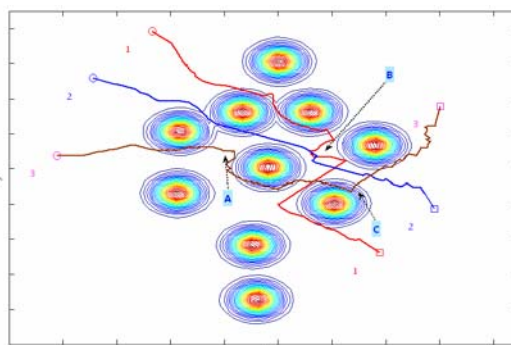


شکل (۱-۵) طی مسیر بدون برخورد با موانع ثابت

#### ۳-۵- طی مسیر بدون برخورد با موانع متحرک

چیدمان ربات‌ها و اهداف آنها را به گونه‌ای تغییر داده‌ایم که در حین طی مسیر، بالاجبار در مسیر همدیگر قرار بگیرند.

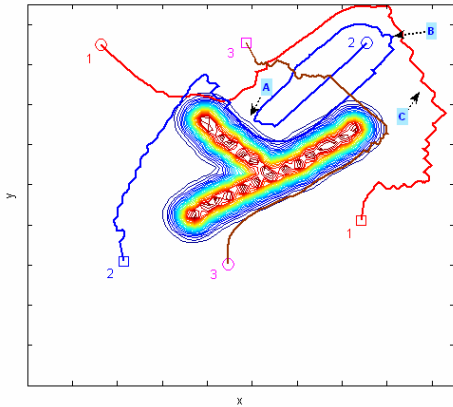
همانطوری که در شکل (۲-۵) مشاهده می‌کنید، ربات شماره ۳، در نقطه A مشاهده نموده است که ادامه دادن مسیر، منجر به برخورد با ربات ۲ یا مانع کنار آن خواهد شد. لذا تغییر مسیر داده و با دور زدن مانع مسیر دیگری را برای رسیدن به هدف انتخاب نموده است. همین ربات بعد از دور زدن مانع، این بار به دلیل وجود ربات ۱ در مسیر حرکتش، و از طرفی تاثیر وجود ربات ۲ در جلو آن از نزدیک شدن به هدف خودداری کرده و بصورت موازی با ربات ۱ و ۲ تا نقطه C حرکت نموده است. در این نقطه به دلیل دور شدن ربات ۲ و کم شدن تاثیر آن، توانسته است که به سمت هدف حرکت کند.



شکل (۲-۵) طی مسیر بدون برخورد با موانع ثابت و متحرک

از طرف دیگر ربات ۱ در نقطه B حضور دو ربات ۲ و ۳ را

شکل (۴-۵) تابع موانع برای این محیط



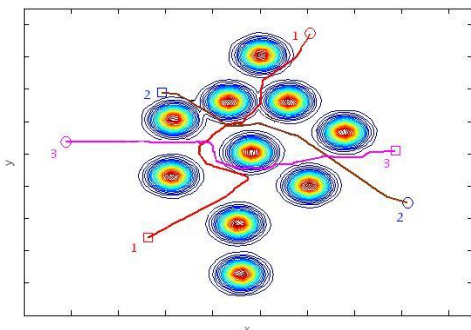
شکل (۵-۵) طی مسیرهای پیچیده و دارای بن‌بست

### ۵-۶- شبیه سازی با در نظر گرفتن مسیر حرکت به عنوان یک مانع با ارتفاع و شیب کم

این مساله باعث می‌شود که ربات کمتر به اطراف منحرف شود، ضمن اینکه در صورتی که در یک بن‌بست گیر کرد، بتواند از روی مسیر قبلی خود عبور کرده و به ادامه مسیر بپردازد.

### ۱-۱-۱- طی مسیر بدون برخورد با موانع ثابت

شکل زیر همان شبیه سازی عنوان شده در بخش ۵-۲- می‌باشد، با این تفاوت که از موانع با ارتفاع و شیب کم استفاده شده است، همانطور که می‌بینید علاوه بر اینکه مشکل برخورد با موانع وجود ندارد، ربات مسیر هموارتری را نیز پیموده است.



شکل (۶-۵) طی مسیر بدون برخورد با موانع ثابت

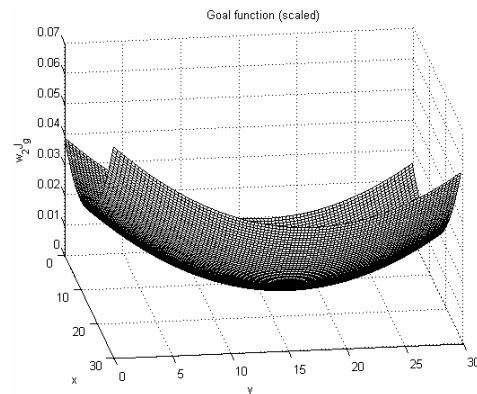
### ۱-۲-۱- طی مسیرهای پیچیده و دارای بن‌بست

در این مورد نیز همانطور که در شکل (۵-۵) مشاهده می-

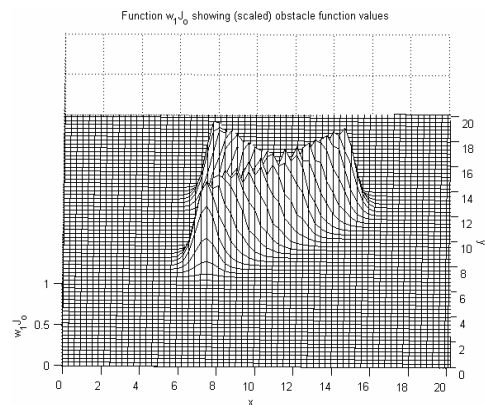
۱. بعلت اینکه ربات ۲ مسیر خودش را بعنوان مانع تعریف نموده است، وقتی به نقطه A رسیده و ربات ۱ را در طول مسیری می‌بیند، به جای اینکه مسیری را تغییر داده و به سمت راست صفحه حرکت کند، مجبور شده است که به موازات مسیر قبلی تا نقطه B حرکت کند و بعد از آن به سمت هدف حرکت کند.

۲. اگر در نقطه A بعلتی مسیر بسته بود، ربات ۲ در یک بن‌بست مجازی که خود در نقشه‌اش ایجاد کرده بود گیر می‌افتاد و نمی‌توانست از بن‌بست ساختگی خودش به هیچ عنوان خارج شود.

۳. در مواردی که سرعت ربات کم می‌باشد، بعلت اینکه ارتفاع موانع مجازی ایجاد شده بوسیله ربات برای خودش زیاد می‌باشد، ربات به صورت کاملاً تصادفی به چپ و راست منحرف می‌شود و باعث می‌شود که حرکت ربات نامنظم بوده و ربات یک حرکت هموار نداشته باشد.



شکل (۳-۵) تابع هدف برای ربات ۱



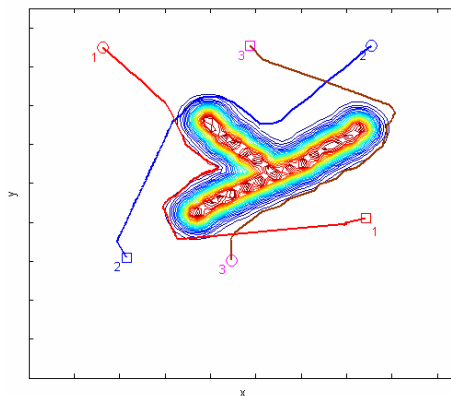
تعدیل نمودیم و به جای آن ارتفاع مسیرهایی را که ربات برای بار دوم می‌پیمود بیشتر نمودیم. این عمل باعث شد که ربات بتواند علاوه بر غلبه بر مشکلات فوق به راحتی و سرعت از مینیم‌های خارج شود، ضمن اینکه حرکت هموارتری داشته باشد.

روش ارائه شده نشان داد که می‌تواند، خیلی سریع از مینیم‌های محلی خارج شده و به سمت هدف اصلی حرکت کند.

### مراجع

- 1 J Barraquand, B Langlois & J.C. Latombe, "Numerical Potential Techniques for Robot Path Planning", *IEEE Trans on Syst., Man, and Cyb.*, 22(2):224-241, 1992.
- 2 M Erdmann & T. Lozano-Perez, "On Multiple Moving Objects", *Proc. IEEE Int. Conf. on Robotics and Automation*, 1986
- 3 S.M. LaValle & S.A. Hutchinson, "Optimal Motion Planning for Multiple Mobile Robots having Independent Goals", *IEEE Trans. On Robotics and Automation*, 14:912-925, 1998
- 4 V.J. Lumelsky & K.R. Harinarayan, "Decentralized Motion Planning for Multiple Mobile Robots: The Cocktail Party Model", *Autonomous Robots*, 4:121-135, 1998.
- 5 R. Alami, F. Ingrand, and S. Qutub, "A Scheme for Coordinating Multi-robot Planning Activities and Plans Execution", *Proc 13th European Conference on AI*, 1998
- 6 K. Azarm and G. Schmidt. "Conflict-Free Motion of Multiple Mobile Robots Based on Decentralized Motion Planning and Negotiation", *Proc. IEEE Int. Conf. on Robotics and Automation*, p. 3526-3533, 1997.
- 7 S Carpin and E Pagello, "A Distributed Algorithm for Multi-robot Motion Planning", *In Proceedings of the fourth European Conference on Advanced Mobile Robotics*, 2001.
- 8 Hu, E., Yang, S.Y., Chou, D., and Smith, W.R., "Real time tracking control obstacle with obstacle of multiple mobile robots". *Proceedings of IEEE International Symposium on Intelligent Control*, pages 87-92, Vancouver, Canada, 2002
- 9 R. Siegwart, I. R. Nourbakhsh. "Introduction to Autonomous Mobile Robots". Massachusetts Institute of Technology. 2004.
- 10 M. Ashoorirad, R. Barzamini, A. Afshar, J. Jouzdani "Model Reference Adaptive Path Following for Wheeled Mobile Robots" *Proceedings of the International Conference on Information and Automation*, Colombo, Sri Lanka, IEEE-ICIA2006, pp. 289-294.

کنید، علاوه بر اینکه ربات مسیر هموارتری را پیموده است، بعلاوه نوسانات کمتر، ربات ۱ به جای حرکت به سمت راست صفحه به سمت چپ حرکت نموده و به طور کلی در مسیر ربات ۲ قرار نگرفته است، ضمن اینکه در هنگامی که در بن‌بست قرار گرفته است، بدون تلاش بیهوده، بلافاصله مسیر دیگری را انتخاب نموده و به حرکتش ادامه داده است. لذا این روش جدید توانسته است به خوبی مشکلات روش قبلی را شناسائی و برطرف نماید.



شکل (۵-۷) طی مسیره‌های پیچیده و دارای بن‌بست

### ۶- نتایج

از لحاظ تکنیک مسیر یابی در این پروژه از ترکیب هر دو روش متمرکز و غیر متمرکز استفاده شده است. این تکنیک مساله همگرایی به هدف را که از جمله موارد مهم در مسیر یابی محلی می‌باشد را می‌تواند حل کند.

در این مقاله راه حلی برای حل مشکل بن‌بست و خروج از آن برای ربات‌های سیاری که از روش میدان نیروی مجازی برای مسیریابی استفاده می‌کنند، ارائه شد. در اولین قدم با استفاده از قوانین فازی و بعد از تعیین وقوع بن‌بست، سعی شد که ربات را از موانع دور کنیم تا زمانی که بتواند از بن‌بست خارج شود. با این روش ربات بایستی مدت زمان زیادی را در یک محل صرف کند تا متوجه شود که در بن‌بست گیر کرده است. در مرحله بعد، مسیر حرکت ربات را در نقشه‌اش با استفاده از یک سری موانع پوشاندیم، به این ترتیب، ربات خیلی سریع از بن‌بست بیرون می‌آمد ولی مشکل این روش حرکات غیرعادی ربات و گیر کردن در بن‌بست‌های ناشی از حرکت خود ربات بود. برای حل مشکل تابع هزینه اختصاص داده شده به مسیر حرکت ربات را