

# مسیریابی برای ناوبری یک روبات پایه متحرک، براساس گراف ورونوئی تعمیم یافته

یاسین زمانی<sup>۱</sup>، محمد علی منتظری<sup>۲</sup>

<sup>۱</sup> دانشجوی کارشناسی ارشد دانشکده برق و کامپیوتر، دانشگاه صنعتی اصفهان، y.zamani@ec.iut.ac.ir

<sup>۲</sup> استادیار دانشکده برق و کامپیوتر، دانشگاه صنعتی اصفهان، montazeri@cc.iut.ac.ir

چکیده - مسیریابی در محیط‌های پیچیده، همواره یکی از چالش‌های پیش روی روبات‌های متحرک بوده است. در این مقاله شیوه‌ی نوینی برای مسیریابی روبات‌های پایه متحرک در محیط‌های پیچیده، بیان گردیده است. در گام نخست مسیریابی امن محیط، به وسیله استخراج گراف ورونوئی مشخص می‌شود. سپس در گام دوم، کوتاهترین مسیر رسیدن به هدف با استفاده از الگوریتم  $A^*$  مشخص می‌گردد. در این مقاله گراف ورونوئی تعمیم یافته در محیط‌های پیچیده، با نگرشی نو ترسیم شده است. کلید واژگان - الگوریتم  $A^*$ ، گراف ورونوئی تعمیم یافته، مسیریابی روبات متحرک

## ۱- مقدمه

که نقاط تولید کننده هر دو طرف آن‌ها متعلق به شیء یکسانی هستند حذف می‌شوند. در [۷]، سودها نمودار ورونوئی را با استفاده از تصویر دودویی فضای کاری که به وسیله‌ی یک دوربین دیجیتال به دست آمده می‌سازد.

یک خانواده از الگوریتم‌ها بر مبنای رویه‌های ساخت افزایشی، بر اساس اکتشاف محیط‌های ناشناخته عمل می‌کنند. مواردی از این رویکرد به وسیله‌ی ناگاتانی، چاست و ترون در [۵، ۸، ۹] پیشنهاد شده‌اند. این الگوریتم‌ها خطاهای رمزنگار را در نظر نگرفته و نمی‌توانند در محیط‌های بزرگ مورد استفاده قرار گیرند.

برخی از الگوریتم‌های مسیریابی، از مفاهیم مربوط به نمودارهای ورونوئی الهام گرفته‌اند. روش MAPRM [۱۰]، که پیکربندی نمونه‌برداری شده را در محور میانی (یا نمودار ورونوئی تعمیم یافته) مجتمع می‌کند، از جمله این الگوریتم‌ها است. همچنین مبنای نمودار EDD [۱۱]، ایده‌ی نقشه‌ی راه ورونوئی می‌باشد. این نمودار یک نقشه راه بوده، که به وسیله‌ی وصل کردن مقادیر بیشینه‌ی محلی یک تابع اجتناب از مانع تعریف شده با استفاده از یک تابع فاصله ایجاد می‌شود. مهم‌ترین نقطه‌ی ضعف این روش‌ها این است که، نقشه‌ی راه آن‌ها به صورت برون خط ساخته شده و اطلاعات محیطی باید از قبل در اختیار آن‌ها قرار داده شود.

برو و همکاران [۱۲] یک الگوریتم خطی (از این رو به

مفهوم ورونوئی به مدت چهار قرن استفاده می‌شده است. در *Traité de la Lumière* منتشر شده در سال ۱۶۴۴، دکارت از نمودارهایی شبیه به ورونوئی برای نشان دادن حالات سیستم‌های خورشیدی و محیط پیرامون آن، استفاده کرده است. الگوریتم‌های نمودار ورونوئی از دهه‌ی ۱۹۷۰ مورد بررسی قرار گرفته‌اند، که می‌توان بررسی‌های انجام گرفته توسط آرن‌هارمر [۲، ۱]، دی‌برگ [۳]، و اوکاب [۴] را روی الگوریتم‌ها، برنامه‌های کاربردی، و نمودارهای ورونوئی توسعه یافته مشاهده نمود.

نمودار ورونوئی تعمیم یافته، مجموعه‌ای از نقاط می‌باشد که فاصله‌ی آن‌ها از دو نزدیکترین شیء به یک اندازه است [۵]. برای ساخت نمودارهای ورونوئی تعمیم یافته، الگوریتم‌های بسیاری با استفاده از اطلاعات مسافتی فراهم شده به وسیله حسگرهای خارجی متفاوت نظیر سونار، پویشرگر لیزری و دوربین‌های استریو پیشنهاد شده است. روش ساخت نمودار ورونوئی محلی تعمیم یافته<sup>۱</sup> با استفاده از اندازه‌گیری‌های یک پویشرگر لیزری، توسط ماکوویک در [۶] بررسی گردیده است. در ابتدا، نقاط مربوط به یک شیء خوشه‌بندی می‌شوند، سپس نمودار ورونوئی تولید شده و در آخر، لبه‌های خارجی محیط‌های مرئی و آن‌هایی

<sup>۱</sup> Generalized local voronoi diagram

در ساده‌ترین و مشهورترین حالت (که در شکل ۱ قابل مشاهده است)، ما مجموعه محدود  $\{p_1, \dots, p_n\}$  نقاط را در فضای اقلیدوسی داریم. در این مورد هر مکان  $p_k$  به سادگی یک نقطه است، و سلول ورونوئی متناظر با آن (که معمولاً ناحیه ورونوئی نیز نامیده می‌شود)  $R_k$  شامل تمامی نقاطی است که فاصله آن‌ها تا  $p_k$  بزرگتر از فاصله آن‌ها تا هر یک از مکان‌های دیگر نیست. هر سلول مشابه از برخورد نیم-فضاها به وجود آمده است، و از این رو یک چندضلعی محدب می‌باشد. قسمت‌های یک نمودار ورونوئی تمامی نقاطی در صفحه است که از دو نزدیکترین مکان به یک فاصله هستند. رأس‌های ورونوئی نقاطی هستند که از سه یا بیشتر مکان‌ها به یک فاصله می‌باشند.

این مقاله به رسم گراف ورونوئی برای مکان‌هایی می‌پردازد که به صورت اشایی چند ضلعی مدل می‌شوند.

## ۲-۱- یافتن مسیر

با داشتن نقشه و موقعیت هدف، مسیریابی<sup>۵</sup> به فرایند پیدا کردن مسیری<sup>۶</sup> گفته می‌شود که بعد از پیمایش آن به هدف می‌رسیم. در مسیریابی، روبات باید در مورد اینکه در طولانی مدت چه کارهایی را بایستی انجام دهد تصمیم‌گیری کند.

## ۲-۲- الگوریتم $A^*$

معروفترین شکل جست‌وجوی اول-بهترین، الگوریتم  $A^*$  نامیده می‌شود. این روش رأس‌ها را با ترکیب  $g(n)$  و  $h(n)$  ارزیابی می‌کند. یعنی  $f(n) = g(n) + h(n)$  در نظر می‌گیرد. از آن‌جا که  $g(n)$  هزینه مسیر از رأس شروع تا رأس  $n$  و  $h(n)$  هزینه تخمینی ارزانترین مسیر از  $n$  تا هدف است، بنابراین  $f(n)$  هزینه تخمینی ارزانترین راه حل از طریق رأس  $n$  خواهد بود. از این رو رأس با کمترین مقدار  $f$  برای گسترش انتخاب می‌شود.

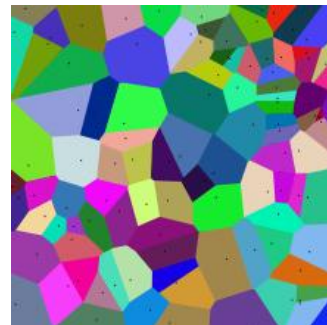
از آن‌جا که در این مسئله تابع هیورستیک، فاصله اقلیدوسی مابین دو نقطه است، جست‌وجوی  $A^*$  کامل و بهینه می‌باشد. نمونه‌ای از به کار بردن الگوریتم  $A^*$  در شکل ۲ قابل مشاهده است. در این مسئله هدف یافتن کوتاهترین مسیر ما بین دو نقطه مفروض در صفحه است. هیورستیک مورد استفاده اندازه خط مستقیم واصل بین نقطه مفروض تا نقطه هدف می‌باشد.

صورت مجانبی مطلوب) برای محاسبه‌ی فاصله‌ی اقلیدوسی دو تصویر دودویی دو بعدی، ارائه کرده‌اند. مبنای این الگوریتم، ساخت و نمونه‌برداری منظم از نمودار ورونوئی، که مکان‌های آن شامل پیکسل‌های (ویژگی‌های) تصویر است؛ می‌باشد. الگوریتم آن‌ها نمودار ورونوئی را جایی می‌داند که، خطوط افقی عبور کننده از میان مراکز پیکسل‌های تصویر، یکدیگر را قطع کنند. آن‌ها همچنین در مورد توسعه کار خود به تصاویر با ابعاد بزرگتر و نیز استفاده از توابع فاصله دیگر، بحث کرده‌اند. الگوریتم پیشنهاد شده در این مقاله تا حدودی شبیه به قسمت اول این مقاله می‌باشد.

از آنجایی که نیاز به تعریف برخی از مفاهیم احساس می‌شود در ادامه به معرفی نمودار ورونوئی تعمیم یافته می‌پردازیم. در بخش سوم شیوه‌ی ساخت گراف ورونوئی مبتنی بر الگوریتم پیشنهادی خود را شرح می‌دهیم. چگونگی یافتن کوتاهترین مسیر بر مبنای نمودار ورونوئی حاصله در بخش چهارم بررسی خواهد شد. در آخر، در فصل پنجم نتیجه‌گیری ارائه گردیده است.

## ۲- نمودار ورونوئی

در ریاضیات، نمودار ورونوئی<sup>۲</sup> نوعی خاص از تجزیه فضای داده شده (به عنوان نمونه فضای متریک) می‌باشد، که به وسیله فاصله تا خانواده خاصی از اشیاء در فضا مشخص می‌شود. این اشیاء معمولاً مکان<sup>۳</sup> نامیده می‌شوند، و متناظر با هر کدام از این مکان‌ها یک سلول ورونوئی<sup>۴</sup> نسبت داده می‌شود. به این معنا که، مجموعه تمامی نقاط در فضای داده شده، که فاصله آن‌ها تا شیء مورد نظر بزرگتر از فاصله آن‌ها تا سایر اشیاء نیست.



شکل ۱: نمودار ورونوئی مجموعه‌ای از نقاط تصادفی در صفحه

<sup>5</sup> Path Planning

<sup>6</sup> Trajectory

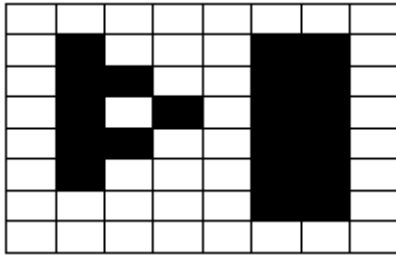
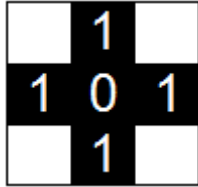
<sup>2</sup> Voronoi Diagram

<sup>3</sup> Site

<sup>4</sup> Voronoi Cell

### ۲-۳- استخراج مرز نواحی

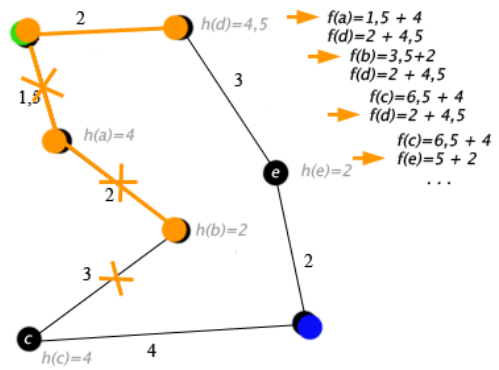
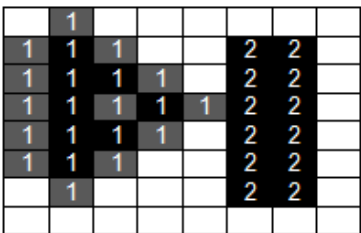
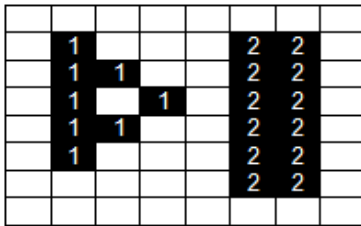
خانه‌ای را خانه‌ی مرزی برای یک شکل قلمداد می‌کنیم که یکی از چهار همسایگی آن جزء زمینه باشد. در این مرحله مرز تمامی نواحی استخراج خواهد شد. شکل ۴ مرز نواحی استخراج شده‌ی موانع شکل ۳ را نشان می‌دهد.



شکل ۴: مشخص کردن مرز نواحی موانع استخراج شده در مرحله قبل

### ۳-۳- رسم نمودار ورونوئی

در هر مرحله مرز هر یک از نواحی مشخص شده در مرحله‌ی قبل را یک واحد گسترش می‌دهیم. این کار را تا زمانی ادامه خواهیم داد که به مرز مشترک برسیم. مرز مشترک نمودار ورونوئی خواهد بود. مراحل رسم نمودار ورونوئی برای مرز موانع استخراج شده در مرحله‌ی قبل در شکل ۵ قابل مشاهده می‌باشد.



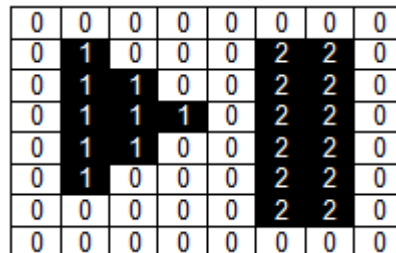
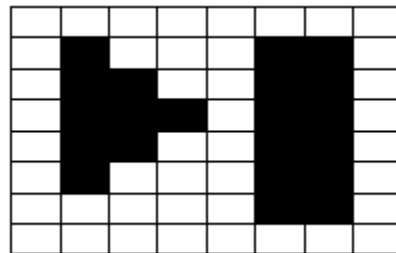
شکل ۲: مثالی برای الگوریتم A\* که در آن رأس‌ها، مکان‌هایی هستند که به وسیله مسیریابی به هم وصل شده‌اند.  $h(x)$  مسافت خط مستقیم تا هدف است. رأس سبز، شروع، رأس آبی، هدف و رأس نارنجی رأس ملاقات شده می‌باشد.

### ۳- استخراج نمودار ورونوئی

این کار در سه مرحله انجام می‌شود. در گام نخست تصاویر موجود در شکل تشخیص داده می‌شود. پس از آن مرز شکل‌ها استخراج شده، و در آخر با بسط هر یک از شکل‌ها گراف ورونوئی قابل محاسبه است. در ادامه هر یک از مراحل به تفصیل بیان شده است.

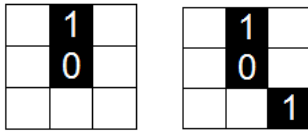
### ۱-۳- استخراج شکل‌ها

در ابتدا تمامی مجموعه‌های همبند در تصویر را با اعداد 1 تا تعداد شکل‌ها، برچسب‌گذاری می‌کنیم. خانه‌های مربوط به زمینه با عدد 0 مقداردهی می‌شوند (شکل ۳).

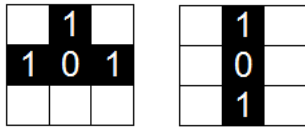


شکل ۳: نحوه‌ی استخراج شکل‌های توصیف‌کننده‌ی موانع در محیط

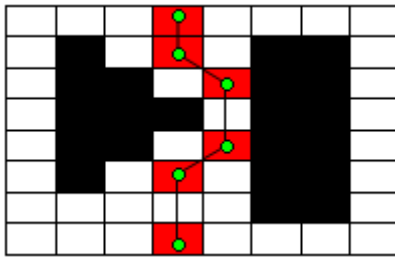
خطوط سیاه رنگ واصل رأس‌ها مشخص شده‌اند.



شکل ۶: نمونه‌هایی از نقاطی که می‌توانند رأسی برای گراف ورونوئی باشند.



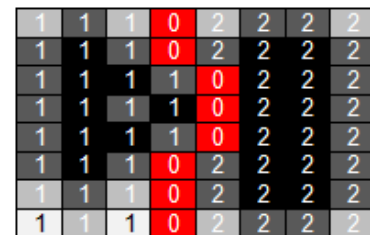
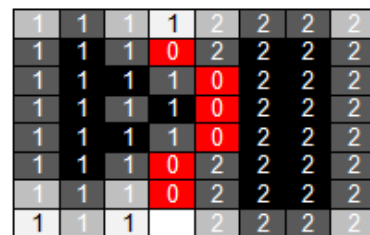
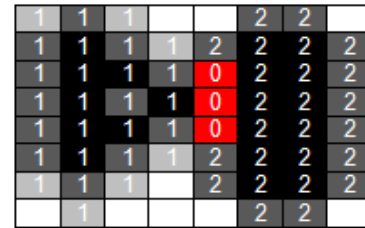
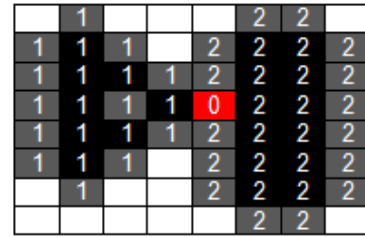
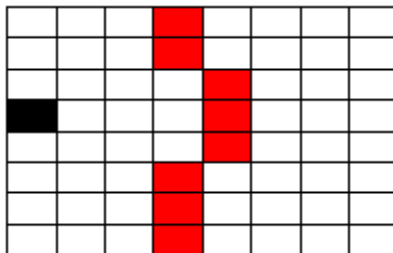
شکل ۷: نمونه‌هایی از نقاطی که نمی‌توانند در مجموعه رأس‌های گراف ورونوئی قرار گیرند.



شکل ۸: نمونه‌ای از گراف ورونوئی استخراج شده. رأس‌ها دایره‌های سبز رنگ و یال‌ها خطوط سیاه واصل رأس‌ها می‌باشند.

#### ۴- یافتن کوتاهترین مسیر

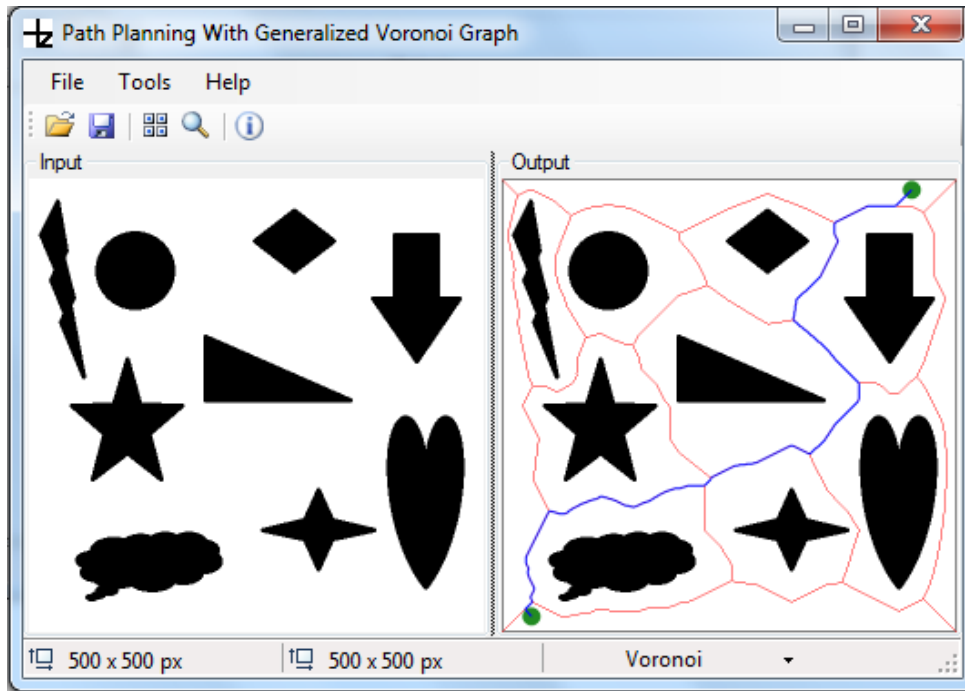
ابتدا نقاط شروع و هدف را مشخص کرده. سپس نزدیکترین نقاط بر روی گراف را به آنها می‌یابیم. شکل ۹ مراحل یافتن نزدیکترین نقطه گراف از نقطه‌ای مفروض (سیاه رنگ) را نشان می‌دهد. هر یک از خانه‌ها آبی را می‌توان به عنوان نزدیکترین نقطه‌ی گراف از نقطه مفروض منظور کرد. حال کوتاهترین مسیر بین این نقاط یافته شده را با استفاده از الگوریتم  $A^*$  محاسبه می‌کنیم.



شکل ۵: مراحل رسم گراف ورونوئی بر اساس مرز نواحی استخراج شده در مرحله قبل. خانه‌های قرمز رنگ مرز مشترک، یا همان نمودار ورونوئی استخراج شده، خواهند بود.

#### ۳-۴ ساخت گراف ورونوئی

یک خانه را رأس می‌گویند اگر تعداد همسایگی‌های آن ۲ نبوده، یا در صورت ۲ بودن؛ تعداد همسایگی‌ها آن‌ها در یک راستا نباشند (همسایگی‌های در راستای محورهای تقارن را، همسایگی‌های هم‌راستا می‌نامیم). برای ساخت گراف ورونوئی همسایگی‌های مجاور را به یکدیگر وصل می‌کنیم. شکل ۶ نمونه‌هایی از نقاطی است که می‌تواند به عنوان رأس‌های گراف ورونوئی در نظر گرفته شوند. شکل ۷ مثالهایی برای نقاطی است که نمی‌توانند در مجموعه‌ی رأس‌های گراف ورونوئی قرار گیرند. نقطه مورد نظر با عدد ۰ و همسایگی‌های آن با عدد ۱ مشخص شده است. گراف ورونوئی شکل ۳ در شکل ۸ قابل مشاهده است. در این شکل رأس‌های گراف با دایره‌های سبز رنگ و یال‌ها با



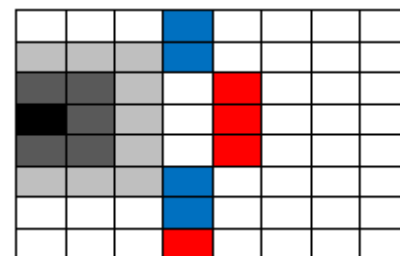
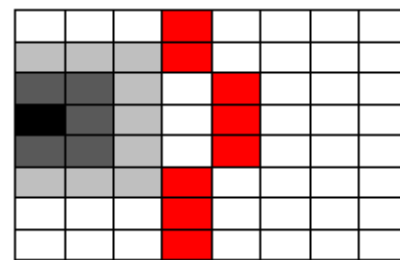
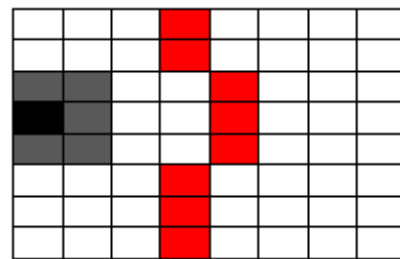
شکل ۱۰: نمونه‌ای از اجرای برنامه، برای مسیریابی ربات پایه متحرک در محیط‌های پیچیده. تصویر سمت چپ نقشه ورودی، دیسک‌های سبز رنگ نقاط ابتدایی و انتهایی مسیر، خطوط قرمز رنگ گراف ورونوئی استخراج شده و خطوط آبی رنگ مسیر طرح‌ریزی شده را نشان می‌دهند.

تصویر ورودی با تفکیک‌پذیری ۵۰۰ د ر ۵۰۰ پیکسل در سمت چپ قرار دارد. نقاط ابتدا و انتهای مسیر به وسیله دیسک‌های سبز رنگ مشخص شده است. گراف ورونوئی تعمیم یافته با رنگ قرمز و مسیر طرح‌ریزی شده با رنگ آبی در تصویر سمت راست قابل مشاهده می‌باشد.

### ۵- نتیجه‌گیری

از آنجایی که از دهه ۱۹۸۰ مسئله مسیریابی ربات متحرک به یک موضوع تحقیقاتی با اولویت بالا مبدل شد، توجه بسیاری از محققین را برای به دست آوردن روشی مناسب برای حل آن به خود معطوف داشت.

روش گراف ورونوئی به وسیله بسیاری از محققین مورد مطالعه قرار گرفت. اما، بسیاری از روش‌های رسم گراف ورونوئی با دشواری‌های مطالعه خطوط، چندضلعی‌ها، پیدا کردن رئوس و رأس‌ها برای ساخت گرافی که کوتاهترین مسیر را به ما بدهد مواجه است. در این مقاله، روشی نوین برای پیدا کردن گراف ورونوئی، مستقل از پیچیدگی موانع در محیط بیان شد، که تنها وابسته به تفکیک‌پذیری تصویر ورودی است. بر این اساس روشی نیز برای مسیریابی ربات‌های پایه متحرک در مسیرهای پیچیده بیان گردید که در آن، مسیریابی از طریق محیط‌های امنی صورت می‌گیرد که به وسیله گراف ورونوئی استخراج شده به دست می‌آیند.



شکل ۹: مراحل یافتن نزدیک‌ترین نقطه گراف از نقطه‌ای مفروض. خانه‌های گراف قرمز رنگ، خانه‌ی مفروض سیاه، و خانه‌های یافته شده که برای انتخاب آن‌ها مختار هستیم به رنگ آبی نمایش داده شده‌اند.

نمونه‌ای از اجرای برنامه در شکل ۱۰ قابل مشاهده است.

- [1] F. Aurenhammer. "Voronoi Diagrams: A survey of a fundamental Geometric Data Structure," *ACM Computing Surveys*, no. 23, 345-405, (1991).
- [2] F. Aurenhammer and R. Klein, *Chapter 5 in Handbook of Computational Geometry*. Eds. J.R.Sack and J. Urrutia, pp. 201–290, (2000).
- [3] M. de Berg, M. van Krefeld, M. Overmars and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications, 2nd rev..* Springer, (2000).
- [4] A. Okabe, B. Boots, and K. Sugihara, "Spatial Tessellations: Concepts and Applications of Voronoi Diagrams". John Wiley and Sons, Chichester, UK, (1992).
- [5] H. Choset et al., *Principles of Robot Motion: Theory, Algorithms, and Implementations*. The MIT Press, (2005).
- [6] R. Mahkovic and T. Slivnik "Generalized local Voronoi Diagram of visible region. In *Proc. IEEE Int. Conf. On Robotics and Automation*, pp. 349-355, Leuven, Belgium, May (1998).
- [7] N. Sudha, S. Nandi, and K. Sridharan, "A parallel algorithm to construct Voronoi Diagram and its VLSI architecture," *In Proc. IEEE Int. Conf. On Robotics and Automation*, pages 1683-1688, (1999).
- [8] H. Choset. "Sensor Based Motion Planning: The Hierarchical Generalized Voronoi Graph," *PhD thesis, California Institute of Technology, Pasadena, California*, March (1996).
- [9] K. Nagatani, H. Choset, and S. Thrun. "Towards exact localization without explicit localization with the generalized voronoi graph," *In Proc. IEEE Int. Conf. On Robotics and Automation*, pp. 342-348, Leuven, Belgium, May (1998).
- [10] S. A. Wilmarth, N. M. Amato, and P. F. Stiller, "MAPRM: A probabilistic roadmap Planner with sampling on the medial axis of the free space". *In Proc. IEEE Int. Conf. On Robotics and Automation*, pp. 1024-1031, (1999).
- [11] S.S. Keerthi, C.J. Ong, E. Huang, and E.G. Gilbert, "Equidistance diagram- a new roadmap method for path planning," *In Proc. IEEE Int. Conf. On Robotics and Automation*, pp 682-687, (1999).
- [12] H. Breu, J. Gil, D. Kirkpatrick, and M. Werman, "Linear time euclidean distance transform algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 5, pp. 529–533, (1995).