ELECTRONICS AND ELECTRICAL ENGINEERING
ISSN 1392 – 1215 2008. No. 4(84)
ELEKTRONIKA IR ELEKTROTECHNIKA

T125 AUTOMATION, ROBOTICS
AUTOMATIZAVIMAS, ROBOTECHNIKA

Creation of vector marks for robot navigation

V. Baranauskas, K. Šarkauskas
Department of Control Engineering, Kaunas University of Technology,
Studentų 48-319, LT-51367 Kaunas, Lithuania; phone +370 – 37 – 300292. E-mail: virgisbar@yahoo.com,
Studentų 48-107, LT-51367 Kaunas, Lithuania; phone +370 61025585, E-mail: Kastytis.Sarkauskas@ktu.lt
S. Bartkevičius
Department of Theoretical Electric Engineering, Kaunas University of Technology,
Studentų 48-230, LT-51367 Kaunas, Lithuania; ph +370 – 37 – 300253. E-mail. Stanislovas.Bartkevicius@ktu.lt

Introduction

Real time navigation of mobile robots requires knowing of necessary path or representation of the environment, robots move. Planned path usually concludes by a sequence of points in the environment. Representation of an environment is more complicated, but allows flexible navigation – a robot calculates its path dynamically. Popular representation of an environment is usage of means of electrostatic field or potential artificial functions [1].

Usage of electrostatic field, when all obstacles have the same potential and only a goal point has another one, is very attractive, because all gradient lines leads to the goal and a robot moving along a gradient line always reaches the goal, but trajectories of movement are far from optimal path, because in some cases movement is executed receding from the goal. So it is necessary to find new methods for trajectories planning.

The weaknesses of these methods stimulate to search for new methods of an environment representation [1]. Requirements for these methods:

1. Ability to define a possible path to the goal or detect absence of it from any point of represented space;
2. The found path is near to optimal;
3. Representation of an environment do not requires large amount of data.
4. An algorithm to calculate current step of robot is rather simple.

The purpose of the article is to introduce to algorithm of vector marks, which are used for systems learning in coloured Petri nets, formation. A set of virtual vectors oriented along the nearest fragment of a path (bitangential line section) with head in vertex or touch point are dominated as a vector mark. In other words, it is set of vectors, oriented along the shortest path to the target, in the turning points of the path.

Authors of the article refer to previous works, where they analysed problems of systems training and learning [2], and related problems of trajectories planning and calculations.

Requirements and constraints

Let us designate the space occupied by obstacles as a set of points S_{obst} and space free of obstacles - S_{free}, a subspace, all points of which are visible from a vertex or touch point, belonging to a path, $-S_v$ [1]. An environment is fully defined by k subspaces, associated with k points, if the following condition is fulfilled:

$$S_{free} = \bigcup_{n=1}^{k} S_{vn} . \quad (1)$$

To define properties of vector marks let us define "visibility" – a point p_d is visible from point p_0 if a segment of line connecting these points exists and all points of this segment belongs to S_{free}:

$$\forall p \in l(p_0, p_d) \in S_{free}, \quad (2)$$

$l(p_0, p_d)$ – denotes line connecting points p_0 and p_d

Properties of vector marks and visibility:

1. All vector marks lie in the space S_{free}.
2. Any vector mark is elongation of bitangential linear fragment of a path.
3. A mark is visible if there is at least one visible point of the mark.
4. From any point of S_{free} at least one of vector marks is visible. This property follows from condition (1) [1].

Visibility problem is relevant not only in process of vector marks formation. The known problem of art gallery [3] essentially deals with the same question. It is shown [4], that a space may be divided into convex polygons, subspaces. The requirement (1) is fulfilled, if any subspace contains, at least, one vector mark. The number and form of convex polygons depends on configuration of the space S_{free} - corners, doors and other. The fact that a part of S_{free} is not visible from a point p_0 can be detected scanning space from this point and fixing discontinuities of visibility – jumps in length of $l(p_0, p_d)$. Let's call these jumps break points.

Algorithm to determine vector marks

If the surrounding is divided into subspaces, it is important to find break points and interconnect subspaces with vector marks.

When all properties of vector marks and their visibility are defined, let's describe and compose algorithm of vector marks formation.

Let us form most important parts of this algorithm firstly:

1. There is target point in the system. Scanning must start from this point.
2. Vector marks are fixing during process of scanning and detecting break points.
3. Scanning is continuing in order to form vector marks of junior generations.
4. Scanning is continuing as long as there is at leas point one in the surrounding, from which at least one vector mark could be visible.

The scanning of the surrounding starts from the target point. The tree structure starts from the target point. It is shown in Fig. 1:

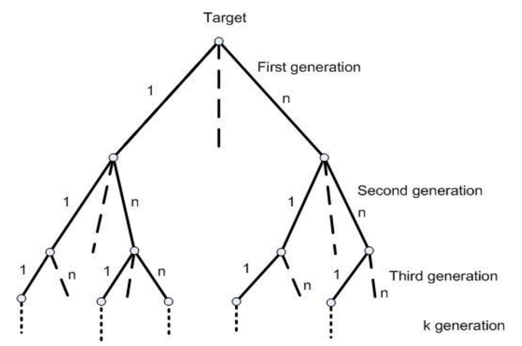

Fig. 1. Structure of vector marks tree.

As we see from the Fig. 1, there could be k generations; amount of them depends on the complexity of the scanning surrounding, and n branches in the structure of vector marks tree. Hatched line in Fig. 1 means possible branches of the tree. If we replace hatched lines in the structure of vector marks tree with solid lines, Fig. 1 would show structure of vector marks tree for the specific surrounding, designed and calculated only for that special case. In this case, Fig. 1 shows the common case of vector marks tree.

Formation of vector marks begins from the target point. Firstly, radar scans at a 360^0 angle, at a fixed step of radar angle. Found break points are fixed. All vectors of the first generation are formed. These vectors are directed to the target point.

If break points repeats in every step and they lay in the straight line – it is parasitical serie (radar beam goes by parallel wall). If parasitical serie is fixed, node, of vector marks tree, is generated. It lies in the parallel line of "seeming wall" near the last point of that serie. All other points of the serie are ignored.

First generation vector marks array *possible[n]* is composed in the next step. Weight coefficients of the vector marks are assigned. Weight coefficient of the vector mark is directly proportional to the distance from target point to the end of the mark.

Further, scanning must start not from the found break point, because, there could be cases, when the point, you want to reach, is inside of the obstacle (point 1 in Fig. 2) or in the opening between obstacles, which do not depends on the width of the opening. This case is shown in Fig. 2.

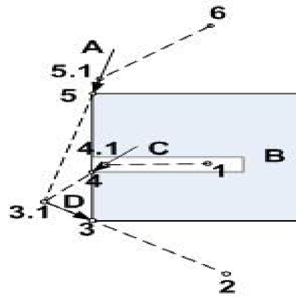

Fig. 2. 6 – the start point, 2 – the target point, B – obstacle, 1 – the possible invisible point inside the obstacle B

Let us analyse Fig. 2. According to theoretical approach, when break point 3 is fixed and first generation vector mark D is formed, further scanning must start from point 3. But, if you start further scanning, according to theoretical approach, from the point 3, point 4 could not be found, because the opening is not in the visible zone. So, in this case, point 1 could not be found. According to the algorithm, when break point 3 is found, further scanning starts from point 3.1. It means, that vector D is elongated at length, equal to the distance from point 3 to point 3.1. This elongation is not fixed length and it depends on the deployment of the obstacles and spaces between obstacles. Authors of the article uses simple heuristic algorithm of the elongation calculation:

```
if possible[n].jump > 4*RadarGap then
        dd := 2*RadarGap
 else
        dd := 0.25*possible[n].jump;
end.
```

Where, possible[n].jump – length of n – th jump, RadarGap – neutral zone in jump length of radar, dd – elongation of the vector.

Numerical values, used in this algorithm, are set from simulation experience. More exact development of this algorithm is the further part of vector marks formation algorithm.

So, how explained above, that further scanning must start not from the found break point, scanner scans at a fixed angle, which must be lesser than 360^0, by a fixed step of radar angle. Scanner scans at an angle, which is lesser than 360^0, due to avoid going back.

Found break points are fixed; next, younger generation vector marks are formed. Next, younger generation, vector marks array is composed. Weight coefficients of vector marks are assigned.

Vector marks of the first generation are the oldest generation vector marks. Vector marks of the second and further generations are junior generation's vector marks. The weight coefficient of junior generation vector marks will be the total of the older generation vector marks weight coefficients. It is:

$$
\begin{aligned}
&D_1 = D_{v1} + dd_1, \\
&D_2 = D_1 + D_{v2} + dd_2, \\
&D_3 = D_2 + D_{v3} + dd_3, \\
&D_i = D_1 + \sum_{l=2}^{i} \left(D_{vn} + dd_n \right), \text{ i=2, 3, ...; n=2, 3, ...;}
\end{aligned} \tag{3}
$$

Where, D_1 – weight coefficient of the first vector, D_{v1} – length of the first vector mark (distance from point 2 to point 3 (Fig. 2)), dd_1 – elongation of the first vector, D_{vn} – length of the nth vector mark, dd_n – elongation of the nth vector mark, D_i- weight coefficient of the i - th vector.

All vectors of each generation are checked among each other do they see vectors of the same generation, or older generations vectors. These vectors, which see vectors of older generations and its weight coefficients are bigger, are eliminated form the further calculations. During processes of scanning and calculating there is requirement, that there were no point in the scanned system, from which at least one vector mark could be seen. This requirement interconnects 4 condition of vector marks visibility, (2) mathematical expression and algorithm of vector marks formation.

So, algorithm of vector marks formation is described. Let us present it in scheme, shown in Fig. 3:

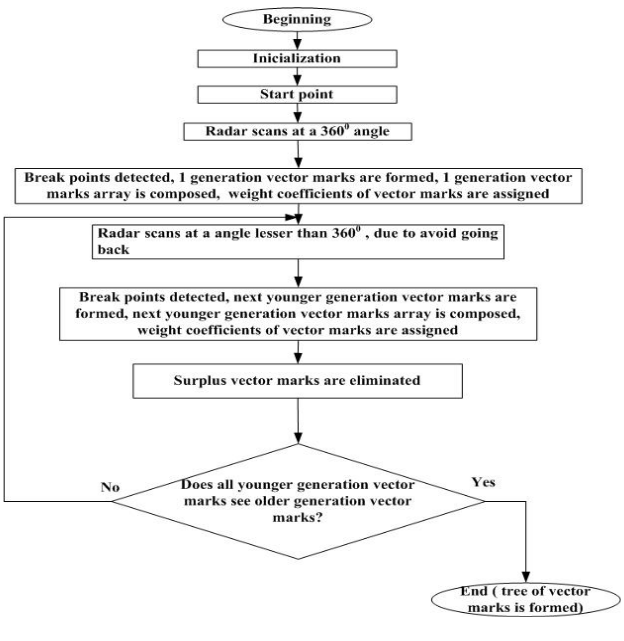

Fig. 3. Algorithm of vector marks formation.

Theoretical example is composed according to this algorithm. Let us analyse it. Let us say we have system, as shown in Fig. 4.

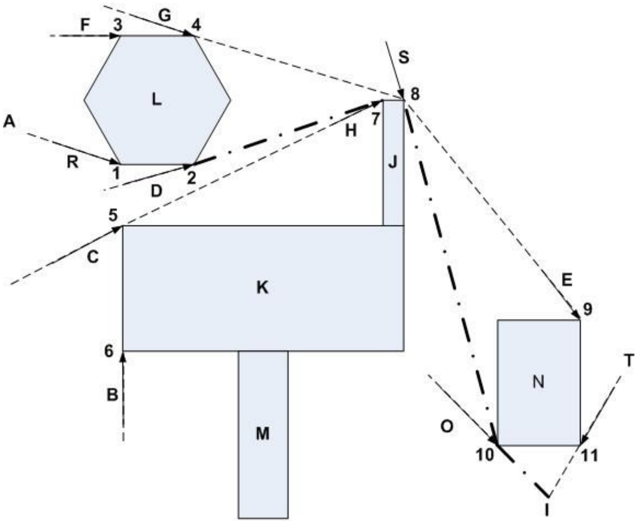

Fig. 4. Theoretical approach of vector marks formation. Areas (L, J, K, M and N) – static objects, vectors (F, G, R, D, C, B, H, S, E, T and O) – theoretical vector marks, A – the start point, I – the target point, points (1-11) – possible collision points with static objects.

Point I is the target point in Fig. 5. Theoretically scanner could find two break points (10 and 11) from it. These points are the vertexes of the vector marks. Oldest (first) generation vector marks O and T are formed. Vector marks are directed to the target point I. Further, during process of scanning, break points 9 and 8 could be detected. Vector marks E and S could be formed. The vertex of vector mark E is in point 9, and the vertex of the vector mark S is in point 8. These vector marks are the second generation vector marks. They are directed not to the target point, as first generation vector mark, but to the visible first generation vector mark. So, we can consider that junior generation vector marks are directed to the visible older generation vector mark. During further scanning from the point 8, break points 4 and 7 could be found. Vector marks G and H could be formed. Further, from the break point 4, only break point 3 could be found. Vector mark F could be formed. From the break point 7, it is possible to find break points 2 and 5. Vector marks D and C could be formed. Break point 6 could be found from the break point 5, vector mark B could be formed. Break point 1 could be found from the break point 2, vector mark R could be formed. Point A is the start point in the Fig. 4. It is important to mention again, that vector marks formation starts not from the start point in the surrounding (point A in the Fig. 4), but from the target point. This theoretical approach of vector marks formation is realised practically and described in further subsection.

Simulated example

Theoretical approach, shown in Fig. 4 is realised in specific software CENTAURUS CPN. It is shown in Fig. 5.

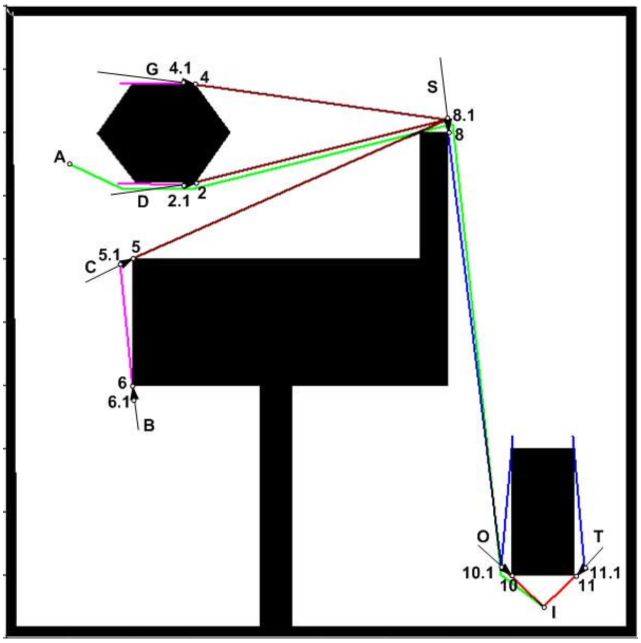

Fig. 5. Formed vector marks in package CENTAURUS CPN. Filled areas are static objects, vectors (B, C, D, G, S, O, T) – vector marks, A – the start point, I – the target point.

As we can see from the Fig. 5, vector marks F, R, H and E, shown in Fig. 4, were eliminated, because other, older generations vector marks sees those zones. When break point is found, further scanning starts not from the found break point. For example, from the target point I, scanner finds break point 10, but further scanning starts from the point 10.1 in Fig. 5. It is obviously, that proposed algorithm of vector marks formation works. Realisation of theoretical model in CENTAURUS CPN proved it.

Conclusions

1. Proposed and composed algorithm of vector marks formation, which is different from theoretical vector marks approach. The difference is that vector marks are elonged in the calculated length, due to the possibility to find invisible zones in the surrounding.
2. Composed algorithm is realised in specific software CENTAURUS CPN. Algorithm with elonged vector marks operates and matches theoretical search.
3. This algorithm of vector marks formation realises the search of the shortest path between start point and target point.

References

1. **Baranauskas V., Bartkevičius S., Šarkauskas K.** Vector marks versus potential field// Elektronika ir elektrotechnika – Kaunas: Technologija, 2008. – Nr. 1(81). – P. 49-52.
2. **Baranauskas V., Bartkevičius S., Šarkauskas K.** Learning possibilities in CPN// Elektronika ir elektrotechnika – Kaunas: Technologija, 2007. – Nr. 6(78). – P. 71-74.
3. **Michael T. S., Pinciu V.** Art gallery theorems for guarded guards// Computional Geometry – Vol. 26, Issue 3, November 2003. – P. 247-258.
4. **Hachenberger P.** Exact Minkowksi Sums of Polyhedra and Exact and Efficient Decomposition of Polyhedra in Convex Pieces// Book Series Lecture Notes in Computer Science - Berlin: Springer, 2007. Volume 4698/2007, ISSN 1611-3349.

Submitted for publication 2008 03 01

V. Baranauskas, S. Bartkevicius, K. Sarkauskas. Creation of vector marks for robot navigation // Electronics and Electrical Engineering. – Kaunas: Technologija, 2008. – No. 4(84). – P. 27–30.

Usage of electrostatic field or artificial potential functions, trajectories of movement are far from optimal path, because in some cases movement is executed receding from the goal. So it's necessary to find new methods for trajectories planning. Algorithm to determine virtual vector marks, representing environment, which is used for purposes of robot navigation is introduced. Simulated example with generated vector marks is placed. Authors of the article refer to previous works concerning problems of trajectories planning and calculations. Ill. 5, bibl. 4 (in English; summaries in English, Russian and Lithuanian).

В. Баранаускас, С. Барткевичюс, К. Шаркаускас. Создание векторных меток для навигации роботов // Электроника и электротехника. – Каунас: Технология, 2008. – № 4(84). – С. 27–30.

Для навигации роботов, используются принципы функций электростатического или искусственного потенциального полей, притом траектории движения роботов далеки от оптимальных, а в некоторых случаях движение роботов происходить с удалением от цели. По той причине и производится поиск новых методов для определения траектории движения. В статье представляется алгоритм формирования виртуальных векторных меток для окружающей среды, в которой осуществляется движение роботов. В статье представляется и пример применения результатов данного алгоритма. Статья основана на более ранних роботах авторов исследующих навигацию роботов при планировании траекторий их движению Ил. 5, библ. 4 (на английском языке; рефераты на английском, русском и литовском яз.).

V. Baranauskas, S. Bartkevičius, K. Šarkauskas. Vektorinių žymių kūrimas robotų navigacijai // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2008. – Nr. 4(84). – P. 27–30.

Robotų navigacijai naudojant elektrostatinį lauką ar dirbtines potencialo funkcijas, judėjimo trajektorijos būna anaiptol ne optimalios, o kai kuriais atvejais net tolstama nuo tikslo. Todėl reikia naujų trajektorijos paieškos metodų. Straipsnyje autoriai pristato sukurtą virtualių vektorinių žymių, naudojamų aplinkai, kurioje juda robotai, vaizduoti, formavimo algoritmą. Straipsnyje pateiktas ir šio algoritmo taikymo pavyzdys. Autoriai remiasi ankstesniais darbais, kuriuose nagrinėjamos robotų navigacijos ir trajektorijų planavimo bei skaičiavimo problemos. Il. 5, bibl. 4 (anglų k.; santraukos anglų, rusų ir lietuvių k.).