Real Time Path Planning for Multiple Robots and Multiple Goals

Waldez Azevedo Gomes Junior

Abstract — **This paper addresses the problem of path planning for multiple robots and multiple goals by using artificial potential fields, APF. It explains the classic artificial potential field approach for a single robot, a single steady goal, possibly multiple steady obstacles, later, it goes through its limitations.**

More up-to-date approaches are mentioned, including evolutionary artificial potential fields, an improved tangent BUG method, ITB, different potential functions used to solve classical problems of the APF.

A simulation of a classical APF that uses the K-means algorithm to solve path planning for multiple robots and goals is performed in a player-stage scenario to demonstrate possibilities and limitations of the APF using a differential-drive robot.

Index Terms **— Evolutionary Artificial Potential Field, Artificial Potential Field, K-means, Improved Tangent Bug method , differential-drive robot**

I. Introduction

THE problem of solving a path for multiple robots in an environment where position and velocities of goals, robots and obstacles are well known is a very relevant problem that could be applied in many fields of industry or even for commercial products. For that matter many algorithms were developed, some of them are mentioned here, such as evolutionary artificial potential field, which is explained in further detail, and the inclusion of an improved tangent BUG algorithm to help solve known issues of the classic artificial potential field.

The classic APF itself is a very natural approach to the problem of path planning due to its elegance and simplicity [2], this is why so many researchers try to overcome its known issues in order to benefit from its advantages [3-7].

In previous studies, APF methods have been used to solve robot path planning in environments where only the robots would move, but this is not the case in most real-world applications where even objects that shouldn't move may be subject to variations that could lead to inefficient path planning or generate a local minima where the robot would then be trapped [3].

Moving robots, goals and obstacles are taken into account in the computations of the evolutionary artificial potential field, EAPF, algorithm. The EAPF algorithm constantly tries to guess the obstacles threats to the robot by taking into account its movements towards the robot and the goal.[3]

The problems that arise from using APF as a control law to an omnidirectional robot were qualitatively and quantitatively analyzed in [2]. Some of these issues are shown in experimental results by the end of the paper.

Other approaches using artificial potential fields use the potential functions just to solve a higher level path planning problem and develop solutions on their own to plan the path locally, that is decompose the control process by function. Low-level processes provide simple functions that are grouped together by higher-level processes in order to provide overall robot control. Local path planning should be performed in real time, and it takes priority over the high level planning performed by APF, it can also be called real time obstacle avoidance and it usually uses local range sensors to plan locally [5].

APF's can also be improved without a major modification in its framework. Small improvements that might come a long way include extended potential fields that keep track of the direction of the robot and direction of the force field generated by the APF, and modifications on the potential functions, like in [7], where the potential function were chosen in order to guarantee a local minimum in the goal despite situations where the obstacles are set very near to the goal.

In regard to multi robot applications one should be concerned with local minima due to robots acting as obstacles to one another. For more than one robot it is not as straightforward as it may seem to efficiently control all of them using APF's. However, in [6] a solution is proposed that takes into account movements from groups of robots that are guided by a leader, where all of them don't follow a fixed geometric form, and try to constantly be at the same distance from one another, but that not being possible they adapt themselves to the environment with the help of local sensors, again the local path planning associated to the high level planning in order to increase efficiency in real world applications.

The next sections of the paper consist of a brief explanation of the artificial potential field, the k-means algorithm, followed by a description of the kinematics of a differential drive robot used in the experiments with the implemented player-stage scenario that uses a modified version of the classic APF approach to attack the problem of path planning with multiple robots and multiple goals, using the k-means algorithm.

II. Artificial Potential Fields

The potential field method treats the robot as a point under the influence of an artificial potential field ($U(q)$) throughout the robot's map. The robot moves by following the field just as ball would roll downhill. The goal acts as an attractive force while the obstacles act as repulsive forces.

The forces are calculated based on its potential functions, eq. 1. , where q is the robot's position at the map.

$$F = -\nabla U(q) \quad (1)$$

The sum of the repulsive forces and attractive forces will define the velocity of the robot, v_r;

$$v_r \propto \sum_i^{\#\,obstacles} F_{rep_i} + F_{att} \quad (2)$$

Thus, it is important to note this method is not only a path planning algorithm; it is also a control law for the robot. Assuming it can localize its position in the map every time and the potential field as well, the robot will always know what to do next.

The most used attractive and repulsive force functions follows:

$$F_{att} = -k_{att}(q_{robot} - q_{goal}) \quad (3)$$

$$F_{rep} = \left(\frac{1}{\rho(q)} - \frac{1}{\rho_0}\right)\frac{1}{\rho^2(q)}\frac{q - q_{obstacle}}{\rho(q)}\ , if\ \rho(q) \le \rho_0 a$$
$$0, if\ otherwise$$

Where, $\rho(q)$ is the minimal distance from the robot to the obstacle, and ρ_0 is a threshold distance that dictates how much further the repulsive field should influentiate the robot.

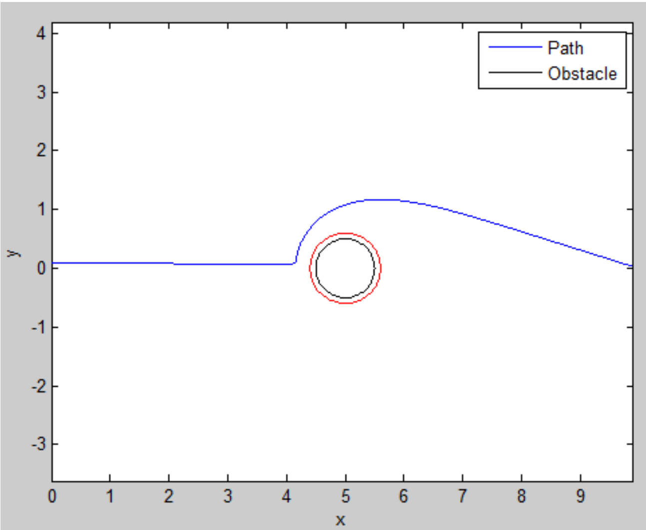

Fig. 1 - Path planning using a classical Artificial Potential Field approach

Fig.1 shows a robot that driving from the origin of the axis to a certain goal at the point (10,0) using the control law/path planning of an APF.

Later on this paper our simulation uses a slight modification of this classic method.

In order to smooth the path of the robot while turning around the obstacle the artificial forces from the obstacles are rotated depending on the location of the goal and the robot, so that the forces acting on the robot because of the goal direct the robot towards the goal instead of only deviating the robot from the obstacle, this is clarified in Fig. 2 and equations 5 and 6.

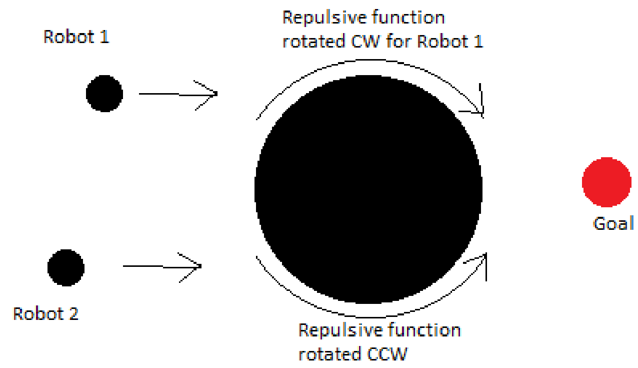

Fig 2 – Figure illustrates the rotation of the repulsive function

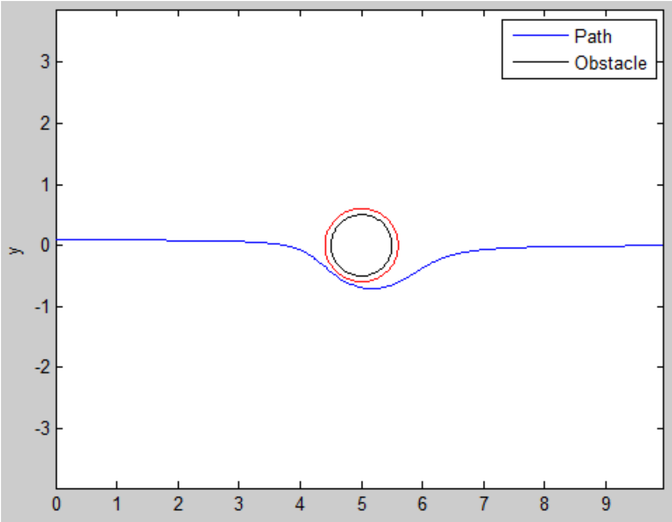

Fig 3 – Smooth path of the robot accomplished by rotating the repulsive function CW or CCW

$$Rotate\ CW\ if\ line\ from\ robot\ to\ goal\ is\ above\ center\ of\ goal \quad (5)$$

$$Rotate\ CCW\ if\ line\ from\ robot\ goal\ is\ below\ center\ of\ the\ goal \quad (6)$$

III. K-Means Algorithm

The k-means clustering algorithms was first presented in the paper [8].Given a certain number of clusters and a set of data point it identifies every point in the data set as belonging to a certain cluster. The algorithm is based in two steps:

- Assignment step, where it assigns a certain point to the cluster which mean is closer to the point;
- Update step, recalculate the means of each cluster.

These two steps are iterated until convergence. The algorithm guarantee that it will converge to a solution, however, does not guarantee that it will converge to the best solution. Another clear setback is that it needs to know *a priori* in how many clusters the data set is divided [9].

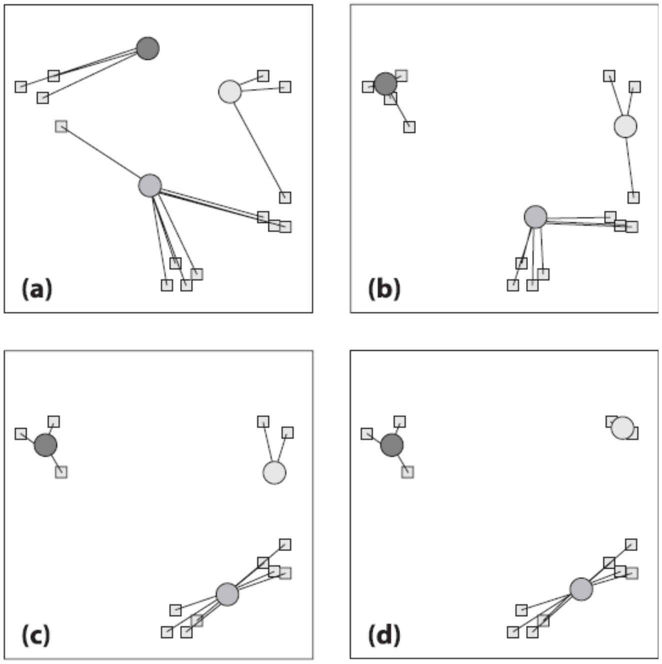

Fig. 4 K-means in action for two iterations: (a) and (c) re assignment steps, while (b) and (d) are update steps [9]

In this paper the K-means algorithm is used to cluster the multiple goals in the map, in such a way that our robots will scatter around the map in order to more efficiently hit all the targets.

IV. Differential Drive Robot

Before talking about how the simulation one should know how our differential drive robot moves. It is not an omnidirectional robot like the one used to make tests in [2], but with the proper set of parameters it can approximate the same path as an omnidirectional robot would take. P in the following equation is the pose of the robot, x, y, and theta describes the 2-D position and the angle between the heading of the robot and the x-axis of the global frame.

$$P = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} \tag{7}$$

$$P' = P + \begin{pmatrix} \Delta s \, cos\left(\theta + \frac{\Delta\theta}{2}\right) \\ \Delta s \, \sin\left(\theta + \frac{\Delta\theta}{2}\right) \\ \Delta\theta \end{pmatrix} \tag{8}$$

$$\Delta\theta = \frac{\Delta s_r - \Delta s_l}{b} \tag{9}$$

$$\Delta s = \frac{\Delta s_r + \Delta s_l}{2} \tag{10}$$

Where Δs_l and Δs_r are the traveled distances for the right and left wheel, those are the only inputs to the robot. The term b is only the distance between the two wheels of the differential drive robot [1].

According to equation (2) the robot should be driven by the forces coming from the APF, to drive this robot with the aid of the APF algorithm the equations 11 and 12 are used to input speeds in each wheel.

$$\Delta s_r = k_f \|\boldsymbol{F}\| + k_{turn}\theta_{turn} \tag{11}$$

$$\Delta s_l = k_f \|\boldsymbol{F}\| \tag{12}$$

Where, F is the sum of all the forces that came from the artificial potential function, K_{turn}, K_f are practical constants, and θ_{turn} is the difference between theta and the angle of F. This way, the robot will always try to keep up with the force field generated artificially.

V. Simulation of Multiple Robots Tracking Multiple Goals

The basics to the simulation done here are calculate the forces from the artificial potential field that actuate in the robot, where not only obstacles but concurrent robots are defined as obstacles as well, drive the robot using equations 11 and 12, cluster the goals using k-means and then send the robots to as many clusters as possible in order to scatter the robots around and more efficiently get to all the goals.

The simulations were conducted in a player stage scenario simulator for differential drive robots developed specially for this paper, where the robots receive as input only the speeds of each wheel and follow strictly the kinematic of equations 7, 8, 9 and 10. Figure 5 shows a robot turning around an obstacle in order to reach the goal.

Fig.5 Robot turning around an obstacle.

The redder the wheels look, the faster is the wheel.

Most of the issues related in [2] were seen in the simulation, such as local minima, no passage between two close obstacles, and sub-optimal paths due to oscillations like in fig. 6.

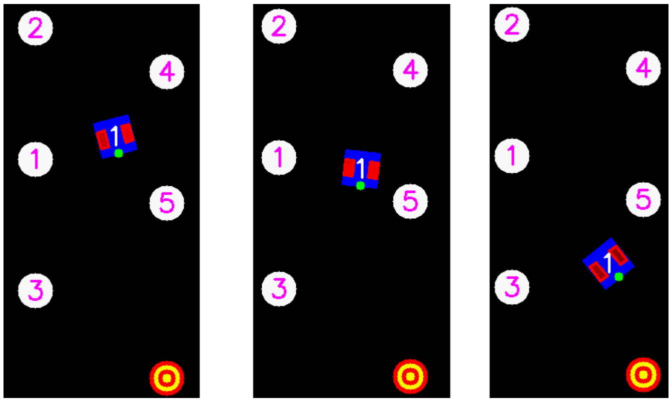

Fig 6 – A sub-optimal path to the goal generated by the APF.

Figures 7 and 8 show further experiments, the latter show the robots interacting with one another while looking for the goal, and the former exemplifies how the goal clustering with k-means help in a more efficient way to reach the goals.

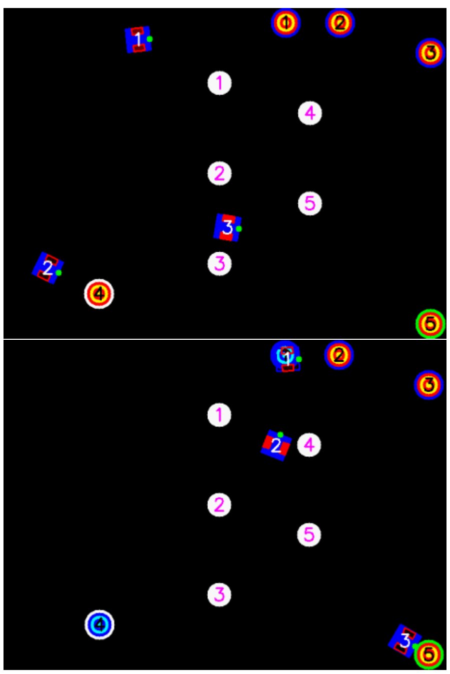

Fig. 7 Robots pursuing goals that have already been clustered using k-means

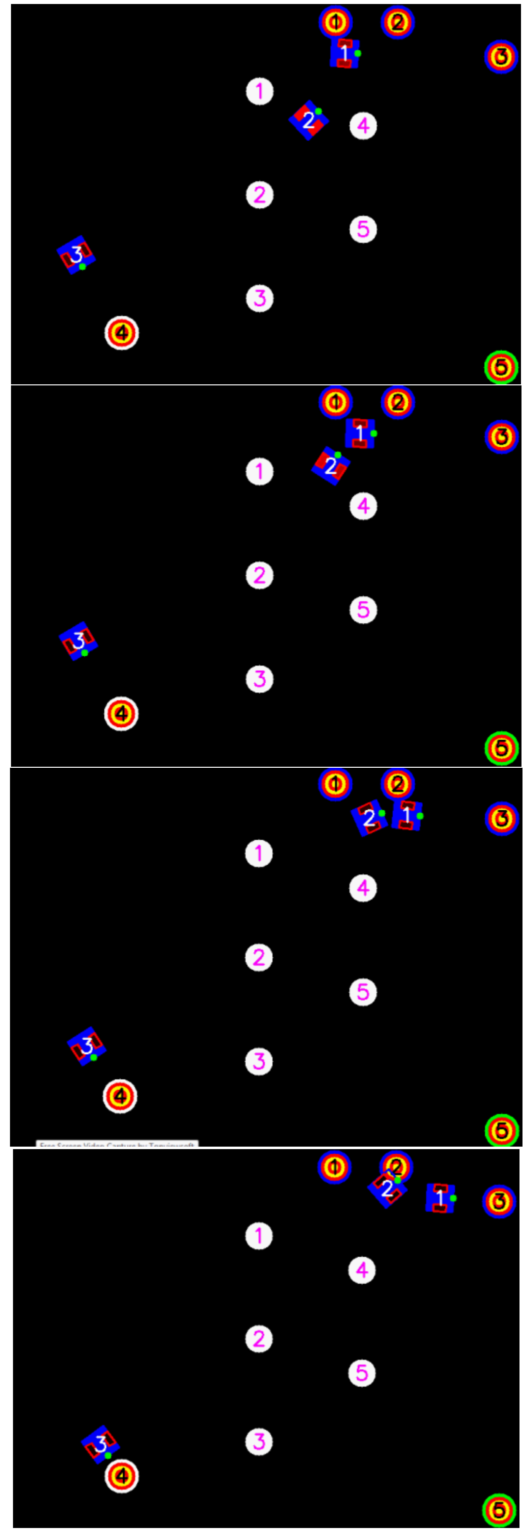

Fig. 8 Robots interacting prior to reaching their respective goals

VI. CONCLUSIONS

The APF can be used to attack the problem of path planning for multiple robots, but several measures should be taken for it to happen efficiently due to the inherent limitations of it. In this paper the modifications used in the simulation slightly improved the sub-optimal paths to the goal, but don't solve it well for moving obstacles, for that matter one obvious extension to the work would be to implement the evolutionary artificial potential field explained in [3], or even the high-level low-level planning hierarchy suggested in [5]. The work here showed that clustering algorithms should work well with an APF field approach in order to maximize the efficiency of the robots reaching the goals in the map. However, further research has to be done in the way of how the clustering is done, in order to improve it, or at least remove the need to know how many clusters there will be in the scenario.

References

[1] R. Siegwart, Ilah R. Nourbakhsh , D. Scaramuzza, "Introduction to Autonomous Mobile Robots", 2nd edition.

[2] Y. Koren, J. Borenstein , "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation".

[3] C. Qixin, H. Yanwen, Z. Jingliang, "An evolutionary Artificial Potential Field Algorithm for Dynamic Path Planning of Mobile Robot".

[4] P. Vadakkepat, K. C. Tan , W. Ming-Liang, "Evolutionary Artificial Potential Fields and Their Application in Real Time Robot Path Planning.

[5] E. Fathy Mohamed, K. El-Metwally, A.R. Hanafy, "An Improved Tangent Bug Method Integrated with Artificial Potential Field for Multi-robot Path Planning"

[6] E.Fathy Mohamed, K. El-Metwally, A.R. Hanafy, "Multi-Robot tracking of Multiple Moving Targets Using Potential field Approach"

[7] S.S. Ge, Y.J. Cui, "New potential Functions for Mobile Robot Path Planning".

[8] J. MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations".

[9] G. Bradski, A. Kaehler, "Learning OpenCv. Computer vision with the OpenCv Library" 1st edition.

Waldez Azevedo Gomes Junior, Exchange undergrad student from University of Brasilia which is located in the city of Brasilia, Brazil. Currently in the 4th year of a 5-year major in control and automation engineering.

He has already been in SAE Aerodesign competitions in Brazil with the team Draco Volans in the performance area, and already participated in robot soccer competitions with the UnBall robot soccer team developing computer vision algorithms.

Mr. Gomes has been awarded a scholarship from CAPES, a Brazilian government institution, to study at Auburn University for the semesters of the fall and spring in the years of 2012 and 2013.