ABSTRACT
Formations required to move to given destinations can be controlled to arrive to a predefined geometric shape or to self-organize. For a large number of autonomous mobile robots, geometric formations pose too many constraints to achieving inter-robot positions. Self-organizing formations are more suitable for a large number of robots and generate formation shapes while avoiding robots collisions. In this paper are investigated approaches for self-organizing formations of large number of robots. The approach, inspired by hydrodynamic flow, results in an artificial velocity field approach that avoids local minima due to a combination of normal and tangential velocity commands associated to obstacles. These commands, for the formation of a large number of robots that are all subject to a velocity vector command to move to the goal, avoid inter robots collisions. In this paper is investigated the efficiency of this control approach in generating a formation shape and maintaining it during motion while moving toward a goal. Experimental and simulation results confirm the applicability of this control approach.

KEY WORDS
Self-organizing formations, autonomous robots, velocity potential field, velocity command

1. Introduction

Currently, there is strong interest to implement autonomous robot systems in military, civilian, and commercial applications. Over the past decade, unmanned vehicle systems have become increasingly prevalent in the military applications primarily due to their reconnaissance, and surveillance abilities.

Enhanced intelligence through robotic systems allows military leaders to make better, more informed decisions, which in turn save lives on the battlefield. Furthermore, civilian applications are also enhanced by autonomous robotic systems. Police, emergency response, coast guard, fire response, search and rescue, border security, and arctic surveillance implement also unmanned robotic systems which allow first responders to stay a safe distance away from dangerous situations.

Autonomous robots are expected to play a large role in emerging commercial applications in natural resources sectors such as mining, oil and gas, and agriculture. Moreover, they will continue to increase efficiency in manufacturing, warehousing, and distribution.

Multi-robot systems have many benefits over their single robot counterparts, since they can perform tasks with greater efficiency, less cost, and they present a more robust solution. Autonomy and group coordination are key traits for the successful navigation of robot teams in dynamic environments. Autonomous motion in multi-robot systems is a very complex problem. Most navigation solutions for autonomous mobile robots are designed, however, for single robot systems.

Multi-robot systems are more complex than their single robot counterparts and require extra elements such as networked communication and group coordination strategies to perform their tasks effectively. Teams of mobile robots must decide as a group what is the best path of motion when difficult unexpected obstacles appear in the environment. There is still no general solution to the multi-robot coordination problem. New solutions for solving the navigation and coordination problems for multi-robot systems could change the way many robot systems are used in military, civilian, and commercial applications. A literature review will clarify the need for the current paper investigation.

In paper [1] is presented a path planning method, inspired by fluid mechanics, able to deal with unstructured terrain models. The algorithm uses the finite element method to compute a velocity potential function free from local minima. Then, several streamlines are computed as a road map and the optimal path is selected among the candidate paths. The approach was implemented on the Canadian Space Agency (CSA) Mars Robotics Testbed (MRT) rover and tested at the CSA Mars Emulation Terrain (MET). To confirm the feasibility of the method, the path planner has been tested on 284 LIDAR scans collected in a realistic outdoor challenging terrain.

In [2], a computational path generation is again inspired from fluid dynamics. It can find optimal paths in a maze.
of arbitrary complexity, and it is flexible because it readily adapts to any change in the topology of the maze. With the selection of appropriate boundary conditions, the fluid dynamics based approach does not suffer from a local minima.

Paper [3] presents a numerical potential function for point-robot path planning in the configuration space based on the theory of fluid mechanics. Ideal fluid is first simulated using Poisson’s equation and heuristic path planning algorithms are established by comparisons of the velocity potentials. Several computational techniques are experimented and compared. A bitmap collision detection technique is proposed for non-point robots. This fluid model creates an environment which is not only free of local minima but also beneficial for navigation control.

In [4], the authors present a novel approach to obstacle avoidance for a group of robots moving in a formation. This idea is originally inspired by hydrodynamics. In this approach, a virtual robot is introduced as a reference point (beacon) to determine a collision-free trajectory. Taking the virtual point as a basic point, the robot group establishes a rigid body-like formation. Since the collision-freeness of the virtual robot does not guarantee that all other robots avoid obstacles, a flexible formation control scheme in the framework of rigid body-like formation was proposed. This approach deals with obstacle avoidance in formation control for multi-robot system for the case of known environment using panel method. Simulation results are presented to verify its effectiveness.

In [5] are used stream functions which satisfy Laplace's equations as local-minima free methods for producing potential-field based navigation functions in two dimensions. These functions generate smooth paths (i.e. suited to aircraft-like vehicles) and a method is developed for building analytic stream functions. The effects of introducing multiple obstacles are also discussed.

Paper [6] presents a general framework for coordinated motion control of autonomous swarms in the presence of obstacles. The proposed framework combines concepts and techniques from potential flows, artificial potentials and dynamic connectivity to realize complex swarm behaviors. Existing concepts from potential flows in fluid mechanics are used to solve the single-agent navigation problem. As an extension, an analytical solution to the stagnation point problem is provided. The potential flow based framework is then modified to facilitate the coordinated control of swarms navigating through multiple obstacles. Artificial potentials are employed for swarming as well as enhanced obstacle avoidance. A novel concept of dynamic connectivity is utilized to improve the performance of obstacle avoidance (Line of Sight Connectivity) and to organize diverse swarm behaviors (Random Connectivity). Simulation results with a set of developed algorithms are included to illustrate the viability the proposed framework.

Paper [7] presents a generic investigation of the decentralized self-organizing potential field-based control approach for mobile agents in a cluttered environment. The paper presents a vector-harmonic potential field approach, also adopted in the present paper in a modified firm for the case of a large number of autonomous robots. Paper [8] presents preliminary results regarding the control of self-organizing formations of a small number of autonomous agents using velocity potential fields for the particular application of material transfer. In this paper velocity potential field approach is applied, differently from other papers, to self-organizing formations of robots in the case of an unknown environment using local sensing and reactive control.

2. Velocity Potential Field Approach

2.1 Superposition of Elementary Plane Flows

Using the elementary plane flow model, complex flows can be approximated by combining various elementary flows [2-7]. Source and uniform flow are described by the following stream function $\psi$ and velocity potential $\phi$, respectively

$$\psi = -\frac{q}{2\pi}(\theta_2 - \theta_1) + Ur \sin \theta$$

(1)

$$\phi = \frac{q}{2\pi} \ln \frac{r^2}{r_1} - Ur \cos \theta$$

(2)

while clockwise vortex and uniform flow are described by

$$\psi = \frac{K}{2\pi} \ln r + Ur \sin \theta$$

(3)

$$\phi = \frac{K}{2\pi} \theta - Ur \cos \theta$$

(4)

where $r$ and $\theta$ are polar coordinates. The doublet of source and vortex is described by

$$\psi = -\frac{q}{2\pi} \theta - \frac{K}{2\pi} \ln r$$

(5)

$$\phi = \frac{q}{2\pi} \ln r - \frac{K}{2\pi} \theta$$

(6)

Fig. 1 illustrates the flow in the case of combined source and vortex.

2.2 Mobile Robot Controller Inspired by the Velocity Potential Flow Theory

For a reactive navigation controller, velocity potential flow functions were implemented for two desired movements, the uniform flow to describe the flow of the
vehicle towards the goal position and the attractive flow for travel at the maximum linear velocity set by the user. Figure 1 shows a differential drive mobile robot moving with the attractive flow towards the goal position.

\[ \phi = -\frac{q}{2\pi} \ln r - \frac{K}{2\pi} \theta \]  

(7)

By applying the cylindrical velocity equations from, the normal and tangential velocities, \( U_n \) and \( U_t \) are calculated to be

\[ U_n = -\frac{q}{2\pi r} = -\frac{A}{r} \]  

(8)

\[ U_t = \frac{1}{r}\left( -\frac{K}{2\pi} \right) = -\frac{B}{r} \]  

(9)

where A and B are constant parameters.

Next, the spiral vortex (source and vortex) potential flow function is used to push the robot away and around obstacles. The path of the robot can be chosen along the flow lines of a spiral vortex, as it approaches the obstacle, and changes course to avoid collision [2-7].

Fig. 1 The flow in the case of combined source and vortex

The normal and tangential velocities of the robot, as it travels along the spiral vortex, can be calculated from the cylindrical velocity equations. Figure 2 shows the differential drive mobile robot moving with the attractive flow towards a goal position.

2.2 System Parameters

When designing the algorithm for the reactive navigation controller with the velocity potential method, it was decided that the vehicle would flow towards the goal using a uniform flow field. The vehicle would travel on a straight line path from the initial position to the goal position. If the vehicle was not initially pointing at the goal, it would continue to travel at the same linear velocity, but would also rotate slowly until it points in the direction of the goal. When the vehicle enters the region of interest around the goal (dashed circle in Figure 4), it was designed to automatically slow down and then stop at the goal position. Figure 4 shows the mobile robot velocity and angle parameters, the distance to the goal position, and the region of interest around the goal position. The robot parameters from Figure 4 are the following [8]:

current position of the vehicle

\( (x, y, \theta) \)
position of the goal

\((x_G, y_G, \theta_G)\)

Maximum linear and angular velocity (chosen by the user)

\(v_{\text{max}}, \omega_{\text{max}}\)

Distance from current position to the goal position

\(\rho_G = \sqrt{(x_G - x)^2 + (y_G - y)^2}\)  \(\text{(10)}\)

Goal radius (critical radius around the goal position)

\(\rho_{GR} = \text{Distance defined by user}\)

The equation used to slow down the vehicle and to cause it to stop at the goal position when it enters the goal radius, suggested in [7], is

\(f(\rho_G, \rho_{GR}) = e^{-(\rho_G/\rho_{GR})}\)  \(\text{(11)}\)

Desired vehicle angle to goal is

\(\theta_a = \tan^{-1}\left(\frac{y_G - y}{x_G - x}\right)\)  \(\text{(12)}\)

\(\Delta \theta_a\) is defined as the difference between the desired angle and the current vehicle angle

\(\Delta \theta_a = \theta_a - \theta\)  \(\text{(13)}\)

2.3 Goal Attractive Velocity Command

The magnitude of the attractive velocity, shown in Fig.2, is equal to the maximum speed \(v_{\text{max}}\) of the robot

\(|u_a| = v_{\text{max}}\)  \(\text{(14)}\)

Attractive component of velocity command is given by

\(u_a = |u_a| (\cos \theta_a + j \sin \theta_a)\)  \(\text{(15)}\)

where \(\theta_a\) is shown in Fig. 4. Velocity vector command to move to the goal is

\(v_a = k_a u_a (1 - f(\rho_G, \rho_{GR}))\)  \(\text{(16)}\)

where \(k_a\) is a constant weight. Angular velocity is the following

\(\omega_a = k_a (1 - f(\rho_G, \rho_{GR})) \frac{\Delta \theta_a}{\Delta t}\)  \(\text{(17)}\)

2.4 Collision Avoidance Velocity Command

The magnitudes of the normal and tangent vectors, shown in Figure 5, are function of the shortest distance to the goal \(\rho_O\) and the obstacle radius [7, 8]

\(|u_n| = |u_t| = \frac{1}{\rho_O}\)  \(\text{(18)}\)

The equation for slowing down the vehicle close to the obstacle is given by

\(f(\rho_O, \rho_{OR}) = e^{-(\rho_O/\rho_{OR})}\)  \(\text{(19)}\)

Obstacle angles, associated with the shortest sensor distance \(\rho_o\), are:

the of the angle normal vector \(U_n\)

\(\theta_n = \theta_O - \pi\)  \(\text{(20)}\)

the of the angle normal vector \(U_t\)

\(\theta_t = \theta_n \pm \frac{\pi}{2}\)  \(\text{(21)}\)

Normal velocity vector command is given by

\(u_n = |u_n| (\cos \theta_n + j \sin \theta_n)\)  \(\text{(22)}\)

while the tangent velocity vector command is

\(u_t = |u_t| (\cos \theta_t + j \sin \theta_t)\)  \(\text{(23)}\)

The amplitudes \(U_n\) and \(U_t\) are defined in equations 8 and 9. The resultant velocity vector command is obtained by
weighted summation of the normal vector with the tangent vector. The resultant vector has an x-component $R_x$, and a y-component $R_y$. The angle $\theta_{new}$ of the resultant vector is the new desired angle of the robot.

$$\theta_{new} = \tan^{-1}\left(\frac{R_y}{R_x}\right)$$  \hspace{1cm} (24)

$\Delta \theta$ for the obstacle is

$$\Delta \theta_O = \theta_{new} - \theta$$  \hspace{1cm} (25)

Overall velocity vector commands are [7, 8]

$$\mathbf{v}_a = k_a \mathbf{u}_a f(\rho_O, \rho_{OR}) + k_t \mathbf{u}_t f(\rho_O, \rho_{OR})$$  \hspace{1cm} (26)

$$\mathbf{v}_r = k_a \mathbf{u}_a (1 - f(\rho_O, \rho_{GR})) + k_n \mathbf{u}_n f(\rho_O, \rho_{OR}) + k_t \mathbf{u}_t f(\rho_O, \rho_{OR})$$  \hspace{1cm} (27)

$$\omega_r = k_o (\rho_O, \rho_{OR}) \frac{\Delta \theta_O}{\Delta t}$$  \hspace{1cm} (28)

3. Simulation Algorithm

Player Proxies™ was used to connect the robots to the PC over a wireless network. The simulation program calculates and sets the velocities of all of the team members in a loop until one of the robots reaches the goal position. Figure 6 shows how the controller sets the proxies for any number of robots and the instruction to obtain $x$, $y$ and $\theta$.

The velocity calculations, shown in Figure 7, are looped until each robot calculation is carried out. When obstacles are sensed, $\rho_o$ and $\theta_o$ are measured and $\mathbf{u}_a$ and $\mathbf{U}_t$ are calculated with equations 18, 22 and 23. Velocity commands for $v_a$, $v$, and $\omega$ from equations 26-28 are set, and when a robot has reached the goal position, all of the vehicles are stopped and the simulation ends.

Simulations and experiments using the velocity potential controller were carried out in MATLAB™ and Player/Stage™.

![Fig. 5. Velocity vector commands for attraction to the goal and collision avoidance [8]](image)

![Fig. 6. Controller setting the proxies for any number of robots.](image)
4. Simulation Results

Simulations were carried out to test first the velocity potential controller on a group of three robots. The reason for this choice is to obtain simulation results for a comparison with the experimental results from section 5. Figure 8 shows snapshots of the simulation of the self-organizing formation with three robots. All robots were able to successfully navigate towards the goal without any collisions.

By incorporating the velocity potential method, each robot created a safety zone around its body that kept the other robots safely away. Next simulation results refer to a large team consisting of 20 mobile robots. The formation becomes elongated as the robots navigated towards the goal position. Since the twenty robots simulation had many team members, the controller generated a formation of oval shape towards the rear of the formation and pointed shape toward the goal, since the back team members needed to slow down in order to avoid colliding with the robots at the head of the formation. Figure 9 shows snapshots of the self-organizing formation for a team of twenty mobile robots. In this case of a formation consisting of many robots, velocity potential method achieved for each robot a safety zone around its body that kept the other robots safely away.

The final oval shape is achieved as a result of self-organization of the formation under the velocity command given for each robot by equations 27 and 28. This formation was not pre-defined as a target formation. The relative positions of the robots in the formation are governed by the collision avoidance velocity commands, $\mathbf{U}_a$ and $\mathbf{U}_t$, and by the goal attracting velocity command $\mathbf{U}_g$. The shape resembles the natural shape of the plume of gas driven by a steady wind. The results in Fig. 9 are dependent on initial conditions shown in the first image where the robots are placed in an arbitrary formation.
Further simulations will eventually use different control structures and include, besides the goal attraction command $U_a$ and inter-robot collision avoidance commands, $U_h$ and $U_i$, also inter-robot attraction commands for better formation hold, independent of initial conditions.

5. Preliminary Experimental Results

In the experiments with the X80™, the goal was to test the velocity potential for self-organizing formation using the available three robots. In the experiment, the three robots were placed in a free space. Using the self-organizing formation approach, the robots navigated towards the goal position in front of the camera, while maintaining a safe distance from each other. Figure 10 shows snapshots of the experimental results for three robots for a self-organizing formation.

Fig. 9. Snapshots of the simulation of a self-organizing formation of twenty robots
controller created a self-organizing formation, where robots on the team would orient themselves in order to reach the goal, but also not to collide with each other along the path.

The proposed navigation controller, the velocity potential flow controller, for a large multi-robot systems, was able to move through the environment like a swarm formation using this new mechanism of self-organization to a group of large number of decentralized autonomous robots.

The oval shape of the formation is maintained consistently in the last part of the motion without the need to redefine it.

References


6. Conclusion

The purpose of this paper was to present a navigation and formation control strategy for a team of large number of robots through a two-dimensional terrain to a goal position without colliding. In order to solve the motion planning problem, a velocity potential navigation controller was developed. The navigation controller was inspired by the velocity potential function in fluid mechanics, and modeled the flow to the goal as a uniform flow field, as well as the flow around the obstacle as a spiral vortex. Simulations and experiments using the velocity potential controller were carried out in MATLAB and Player/Stage. Simulations and experiments for a formation of three robots verified the algorithm.

After verifying the navigation strategy for the team of three robots, formation controllers were simulated to coordinate the members of a twenty robots self-organizing formation. The results illustrate the velocity potential method, and allowed the robots movement to be defined in a distributed decentralized manner by the velocity potential controller. The multi-robot velocity potential...