EXTENSION OF A LIGHTWEIGHT FORMATION CONTROL METHODOLOGY TO GROUPS OF AUTONOMOUS VEHICLES

Gabriel Hugh Elkaim[1] **and Robert Joseph Kelbley**[2]

[1]*Dept. of Computer Engineering, UCSC, 1156 High St., Santa Cruz, CA, 95060, USA, elkaim@soe.ucsc.edu*
[2]*Dept. of Computer Engineering, UCSC, 1156 High St., Santa Cruz, CA, 95060, USA, rkelbs@soe.ucsc.edu*

ABSTRACT

A simple yet robust formation control methodology can provide group cohesion and obstacle avoidance for teams of autonomous vehicles. We previously demonstrated a lightweight formation control methodology that uses conservative potential functions and a virtual leader for group control. This technique provides a robust and stable response with very modest communications and computational requirements for the individual vehicles. We have extended this methodology to provide better virtual leader advancement and also outline steps that should be taken to prevent inter-vehicle and obstacle collisions by modifying the effects of certain potential functions. These additions add little overhead to the methodology and have been proven effective in simulations with realistic kinematic vehicle models.

Key words: Formation Control, Virtual Leader, Artificial Potential, Obstacle Avoidance.

1. INTRODUCTION

Groups of autonomous vehicles offer the potential for increased performance and robustness in several key robotic and autonomous applications. A variety of formation control and coordination frameworks have recently been proposed [1]. As group control techniques become more complex their implementation becomes more difficult and may have unreasonable computational and communication demands. In this paper, we extend our previously discussed lightweight formation control methodology [2] that can be applied to almost any group size and formation restraint. Our technique uses artificial potential functions in conjunction with a virtual leader to provide reliable, robust control with minimal computational and communications costs. For obstacle avoidance we have simplified our environment by bounding obstacles in convex polygons. Extending this methodology to include realistic kinematic models we have identified several additional techniques that should be implemented for successful distributed control. These control techniques are both extremely robust and easily implemented, thus providing a realistic solution to group formation control and coordination, including applications in planetary exploration.

A variety of applications require precise formation control with a quick response to environmental disturbances, such as distributed scientific data gathering experiments which need a large number of different vehicles working together. Current techniques for planetary navigation are unable to provide a control response sufficient for coordination of a large number of remote controlled vehicles, thus future missions will likely focus on harnessing the power of multiple autonomous robot teams for planetary exploration, climate data collection, environmental investigation, and other objectives that require simultaneous, coordinated work. For these reasons, formation control has become a popular area for autonomous systems research.

A biologically inspired group of vehicles has the potential to provide inexpensive yet robust performance for a variety of adverse applications. Typically, the limited computational capabilities and communications bandwidth between vehicles makes large scale control optimization difficult. However, a reliable lightweight control methodology based on a simplified liquid surface tension abstraction can be implemented using current computational and communications constraints.

In 1986 Craig Reynolds showed that rule-based distributed group motion control was possible by creating computer graphic models of coordinated animal motion based on the behavior of fish schools and bird flocks [3]. Much work has been done by mathematical biologists in modeling emergent swarm behavior by reducing it to rules of repulsion and attraction between neighbors [4][5][6]. Similarly, the goal of our control methodology is to model group behavior in a simplified manner as a liquid droplet balanced between gravity and surface tension. To accomplish this we implement different artificial potential functions to achieve group cohesion, separation, and alignment. This allows an equilibrium position for each vehicle to be found forcing the vehicles to follow a steepest decent towards the geometric point at which the sum of the virtual forces becomes zero.

The virtual leader concept is used to advance the group through its environment. It is important to note the virtual leader is not a vehicle, but an imaginary point used as a guide for group movement. Virtual leader is used in a similar way as Leonard from who we have adapted the term [7]. The virtual leader makes it possible for the group to be advanced through a series of way points or areas of interest for the team. Note that only the trajectory of the virtual leader is planned. The collision free paths of all of the other elements of the group are simply generated by virtue of the changing potential functions.

The goal of using potential functions is to repel vehicles away from each other and obstacles while also providing cohesive group motion as their virtual leader is progressed through the environment [8][9]. The potential functions used in our methodology are identified as inter-vehicle forces, virtual leader forces, and obstacle forces. As the virtual leader position is moved, the artificial potential relationships will move the group along the path defined by the virtual leader motion. This global dependence on virtual leader motion reduces the task of planning multiple collision free paths for many vehicles to planning just one collision free path.

However, due to vehicle motion limitations, potential functions can sometimes work against each other in ways that propel a given vehicle onto an undesired course. For example, a group must spread out when navigating through narrow openings between obstacles, however, as the formation narrows the neighboring vehicle potential function forces will increase. If these inter-vehicle forces grow too quickly the result will be vehicles accelerating in the direction of nearby obstacles. In time, a repulsive force from the obstacle will counteract the inter-vehicle forces, however, the vehicles may not be able to adjust their heading and velocity quickly enough, resulting in an underdamped motion. A simple and effective solution is to limit the magnitude of the inter-vehicle and virtual leader forces to a value derived as a function of trajectory update delay, maximum vehicle velocity, and the obstacle repulsion constants used. (This will be discussed in Section 5.)

Another problem that occurs as vehicles traverse through narrow openings between objects is that the velocity of the group's virtual leader must be adjusted to ensure its forces on lagging vehicles does not exceed a reasonable force value. Therefore, the virtual leader's velocity should reduce as the forces it applies to the vehicles increases. One solution is to limit the motion of the virtual leader to a distance that will prevent any of the vehicles from obtaining total force values in the virtual leader's direction of motion that exceeds the motion capabilities for the vehicles. This method ensures that virtual leader control mimics the dynamics of the vehicles it is controlling.

Advancements to our lightweight control methodology discussed in this paper were initially modeled using tricycle steering vehicle models in MATLAB. The Player/Stage platform was used next to show that the control techniques could be successfully implemented using a realistic robotic simulator and control software modeling groups of differential drive robots in complicated object environments.

This paper is organized into the following sections: Section 2 describes the effects of virtual leader forces, Section 3 describes how inter-vehicle forces are applied, Section 4 describes how obstacle forces are implemented, Section 5 covers how kinematic limits are used in collision prevention, Section 6 outlines some optimal virtual leader control rules, Section 7 discusses the different simulation studies used, and Section 8 concludes with a summary of the results and indications for future work.

2. VIRTUAL LEADER FORCES

As mentioned in Section 1, the basic formulation of the group formation control problem is modeled upon a simplified liquid surface tension. For the purposes of this paper, we limit the problem to a two-dimensional plane, although this analysis could be extended to three dimensions. Various formulations were previously attempted in order to generate a set of mathematical relations that would produce a macroscopic behavior to that of a liquid droplet of water flowing between and around various obstacles.

Before artificial potential functions can be applied, an initial formation for the group must be determined. These initial group formation locations will be used as equilibrium positions when calculating group forces. Next, a virtual leader (VL) position must be determined. Typically, the initial virtual leader position should be the center of mass for the desired initial group formation, with the center of mass defined for N vehicles as:

$$x_{cm} = \frac{1}{N}\sum_{i=1}^{N} x_i \quad \text{and} \quad y_{cm} = \frac{1}{N}\sum_{i=1}^{N} y_i \qquad (1)$$

It is then trivial to find the initial distance d_0^{VL} between each node and the virtual leader. When the group is in its proper formation d and d_0 should be equal. As the virtual leader is advanced and the group falls out of equilibrium, the virtual leader forces F_{VL} are found as:

$$\begin{bmatrix} F_x^{VL} \\ F_y^{VL} \end{bmatrix} = K_{VL} \begin{bmatrix} d_x^{VL} - d_{x_0}^{VL} \\ d_y^{VL} - d_{y_0}^{VL} \end{bmatrix} \tag{2}$$

$$d_x^{VL} = x_{VL} - x_i \tag{3}$$
$$d_y^{VL} = y_{VL} - y_i \tag{4}$$

where K_{VL} is the spring constant used to provide the desired cohesive effects for group attraction and advancement. Fig. 1 shows the inter-vehicle and virtual leader artificial potential relationships for a circular formation at equilibrium.

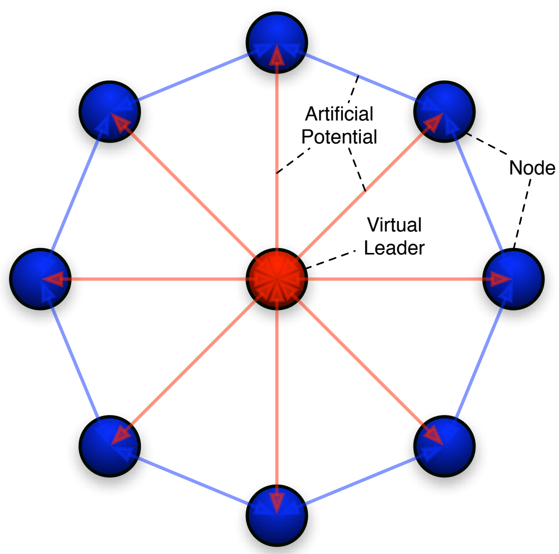

Figure 1. Vehicles at equilibrium in circular formation. Red and blue arrows represent virtual leader and inter-node artificial potentials, respectively.

Fig. 2 demonstrates how the virtual leader forces increase as the leader's position changes. In response the group will move in the same direction as the virtual leader until an equilibrium position is again reached.

3. INTER-VEHICLE FORCES

The separation forces for group vehicles are based on the nominal distance from each vehicle's nearest neighbors. These vehicle to vehicle interactions are conservative forces that attracts vehicles together as their distance increases and repels vehicles as their

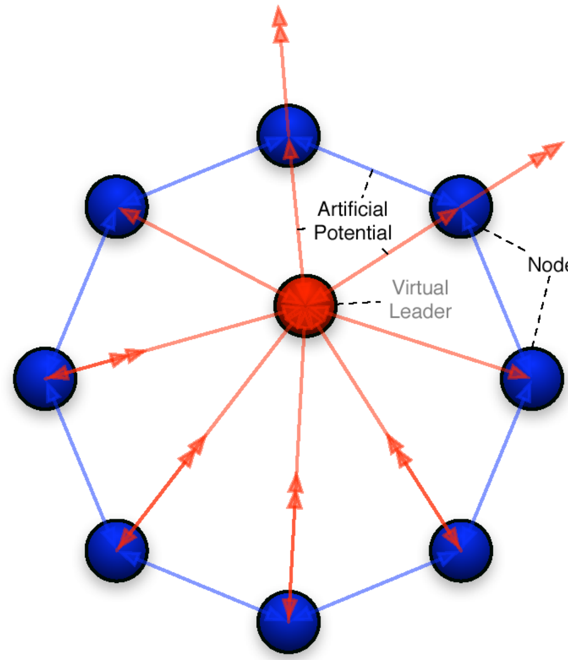

Figure 2. As the virtual leader is advanced, artificial potentials for the group react to force the formation back to equilibrium.

distance decreases. Using K_{ij} as the new spring constant and the initial formation locations discussed in Section 2, we simply define these inter-vehicle forces F_{ij} between two neighbors as:

$$\begin{bmatrix} F_x^{ij} \\ F_y^{ij} \end{bmatrix} = K_{ij} \begin{bmatrix} d_x^{ij} - d_{x_0}^{ij} \\ d_y^{ij} - d_{y_0}^{ij} \end{bmatrix} \tag{5}$$

$$d_x^{ij} = x_j - x_i \tag{6}$$
$$d_y^{ij} = y_j - y_i \tag{7}$$

Similar to the virtual leader distance relation, the two vehicles are at their initial distance d_0^{ij} when the force between them is zero. Formations can be varied in real time as the group progresses by varying these d_0^{ij} values for each vehicle, allowing the force computations to smoothly move the group from one formation to another.

4. OBSTACLE FORCES

To deal with obstacles, we enclose them in bounding convex polygons and impose a repulsive force relationship between the vehicles and polygon edges that is inversely proportional to the distance between them. In this work, we do not deal with obstacle detection and assume that we have knowledge of obstacle position and shape.

Enclosing obstacles in convex polygons has several advantages. Firstly, convex polygons simplify the task of obstacle avoidance by the group. This is accomplished by determining which face of the polygon the individual vehicle lies in front of, and how far away from that surface the vehicle is. Secondly, simplifying obstacles to polygons reduces both the calculation and communications bandwidth required to relay obstacle information between vehicles.

The total obstacle force on a vehicle by n obstacles is given by:

$$F_{ob} = K_{ob} \sum_{k=1}^{n} \frac{1}{d_k^{ob}} \tag{8}$$

where d_k^{ob} is either the perpendicular distance to the face or the straight distance from an object vertex, as appropriate. A vehicle must be determined to lie either in a rectangle in front of a face, or in the wedge between two faces. As derived previously in [2], a simple coordinate translation and rotation technique can be used to determine if a vehicle is located within the projection of the face of an obstacle and if so, it's distance d_k^{ob}. If the vehicle is not located within the face of any sides of an obstacle it is trivial to find the distance from the nearest vertex.

The obstacle force spring constant K_{ob} can be imagined as the threshold distance where F_{ob} will swing from a value $F_{ob} \leq 1$ to a value larger than 1 that will increase linearly as d_k^{ob} decreases. To provide a nonlinear response as d_k^{ob} decreases Eq. 8 can be altered. This will allow the repulsive forces emitted by obstacles to rapidly increase as vehicles approach. Fig. 3 shows an example of nonlinear force gradients for a five sided obstacle.

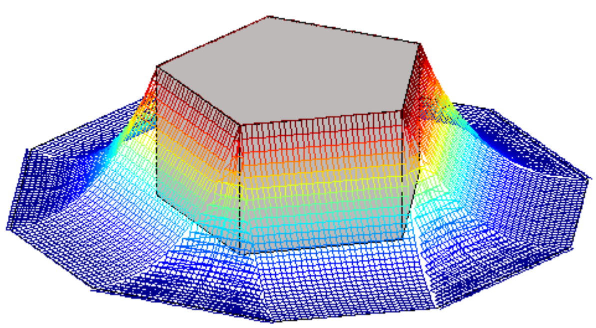

Figure 3. Nonlinear obstacle artificial potential field model. Force values increase more rapidly as the distance from the obstacle decreases.

5. COLLISION PREVENTION

One advantage of using a combination of different artificial potential functions for formation control is that the resulting forces can be summed together to find the total force acting on a vehicle as:

$$F_{tot} = F_{i-1}^{ij} + F_{i+1}^{ij} + F_{VL} + F_{ob} \tag{9}$$

The magnitude and direction of F_{tot} can then be used as the new desired control reference for each vehicle, with the idea that reaching this objective will restore equilibrium to the group.

As the group flows to reach equilibrium, some vehicles may find their new trajectory calculations could cause a collision between other vehicles or obstacles. However, as new trajectories are calculated in real time the vehicles should flow to an equilibrium point at which the sum of all virtual forces is zero, preventing any collisions.

This assumption will not hold, however, if the vehicle dynamics are such that a group member does not update its trajectory information fast enough to recover from its acceleration in undesired directions. A simple way to accommodate vehicle dynamics is to prevent vehicles from being forced into positions they may be unable to recover from depending on their maximum velocity, trajectory refresh rate, and force constants.

A computationally simple way to prevent these collisions is to define a threshold distance around all obstacles that ensures every vehicle has sufficient time to adjust from undesired trajectories. Any distance d_{max} that is greater than the product of a vehicle's maximum speed and its refresh rate should provide enough clearance for collision recovery. Therefore, when summing artificial potential functions a force cap should be placed on F_{ij} and F_{ia} values such that:

$$\left| F_{i-1}^{ij} + F_{i+1}^{ij} + F_{ia} \right| \leq \frac{K_{ob}}{d_{max}} \tag{10}$$

Another collision concern arises when the group is required to compact into a narrow formation to traverse between several close by obstacles. This may result in vehicles from opposite sides of a formation to temporarily obtain colliding trajectory calculations because only neighboring formation vehicles are being considered. Currently only neighboring formation vehicles are considered to reduce the algorithm's complexity and increase performance, however, an additional step can be added when computing F_{ij} values to ensure that no non-neighboring vehicles are entering a predetermined radius d_{min} for each vehicle. Typically this radius should be a function of the desired formation spacing and can use the same spring constant as given in Eq. 5, which could be modified such that:

$$F_{ij} = F_{i-1}^{ij} + F_{i+1}^{ij} + K_{ij}\left(d_{min} - d\right) \tag{11}$$

where d is the actual distance between vehicles.

6. OPTIMAL VIRTUAL LEADER CONTROL

To ensure stable group navigation the virtual leader location must be advanced in a way that does not cause excessive effects on any of the vehicles. This situation arises as the group traverses through narrow openings between obstacles. The vehicles at the head of the formation will clear the obstacles first and be able to resume their desired velocity. However, vehicles at the tail end of the formation will lag behind. If the velocity of the virtual leader is not properly adjusted these trailing F_{VL} forces will grow rapidly and may force vehicles into collision trajectories they will be unable to recover from, similarly as described in Section 5.

A simple relation of vehicle limitations to virtual leader advancement should be applied when monitoring each vehicle's force magnitudes. However, unlike the force limitations discussed in Section 5, now we are only concerned with the forces in the same direction as the virtual leader's motion. This will ensure that group cohesion is maintained and prevent the vehicles in the lead of the formation from advancing faster than vehicles in the middle and tail of the formation.

The virtual leader should therefore only be advanced as far to ensure the following relation holds:

$$F_{tot} \cdot \hat{d_{VL}} \leq \alpha d_{step} \tag{12}$$

where d_{step} is the maximum distance a vehicle can travel in the given formation refresh rate, α is used as the group's advancement coefficient, and $\hat{d_{VL}}$ is the unit vector of the virtual leader's direction of motion.

If a more precise formation control is required, the acceleration of the virtual leader can also monitored. A sudden deceleration of the virtual leader could result in overshoot while vehicles are accelerating to their new control references. If the vehicles cannot decelerate fast enough, they may try to turn around and enter an undesired trajectory to reach the overshot destination. Forces in the virtual leader's direction should therefore be monitored as previously discussed to also ensure acceleration and deceleration constraints are kept.

7. SIMULATION RESULTS

To test our extensions to the previous lightweight flocking methodology, simulation studies were performed to determine the effectiveness of these methods for navigating a flock through an obstacle field. Attempts were made to simulate vehicle kinematics, however, no simulations of obstacle detection capabilities were included.

Various combinations of the spring constants K_{ij}, K_{VL}, and K_{ob} and proportionality gains for the feedback control were used in the simulations.

7.1. MATLAB

Initially MATLAB was used to model obstacle environments for the vehicle groups. Fig. 4 shows the tricycle steering vehicle used to create realistic acceleration and velocity limitations for the kinematic models used in testing. Fig. 5 (right) shows how sudden artificial force increases can cause under-damped vehicle motion, resulting in undesired vehicle to vehicle and obstacle to obstacle collisions. Fig. 5 (left) highlights how the group is kept stable by limiting individual vehicle movement to a response limited by a maximum derived from its dynamic capabilities, as described in Section 5.

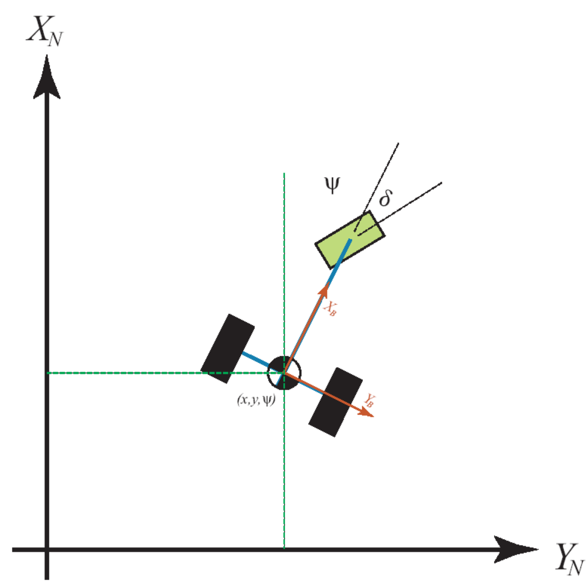

Figure 4. Tricycle steering model used in simulations.

As discussed in Section 6, changing the advancement coefficient of the virtual leader in Eq. 12 alters the formation cohesion of the group when responding to formation disturbances. Lowering the coefficient slows the advancement of the group but provides better stability and a more consistent formation. Fig. 6 compares the effects of using different values for the advancement coefficient α when navigating through obstacles.

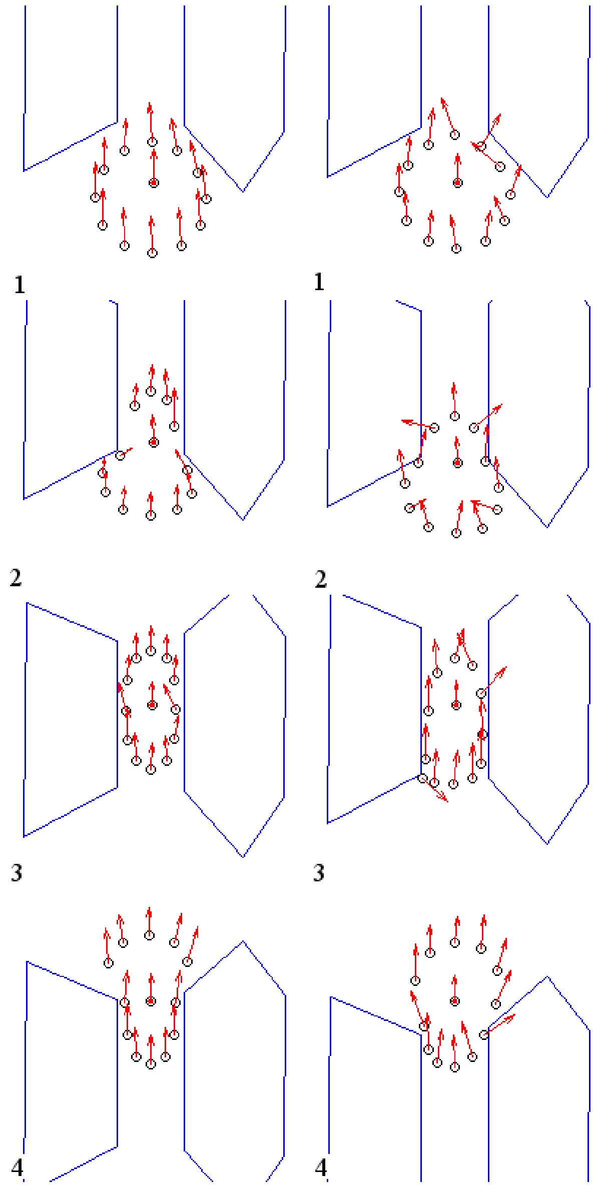

Figure 5. A group of vehicles moving between 2 obstacles. Vehicle velocity is shown as arrows and the center, solid node represents the virtual leader. Force limits and VL control (left) provide a stable response. Unregulated potential functions can cause loss of stability in narrow passages (right), resulting in collisions between vehicles and obstacles.

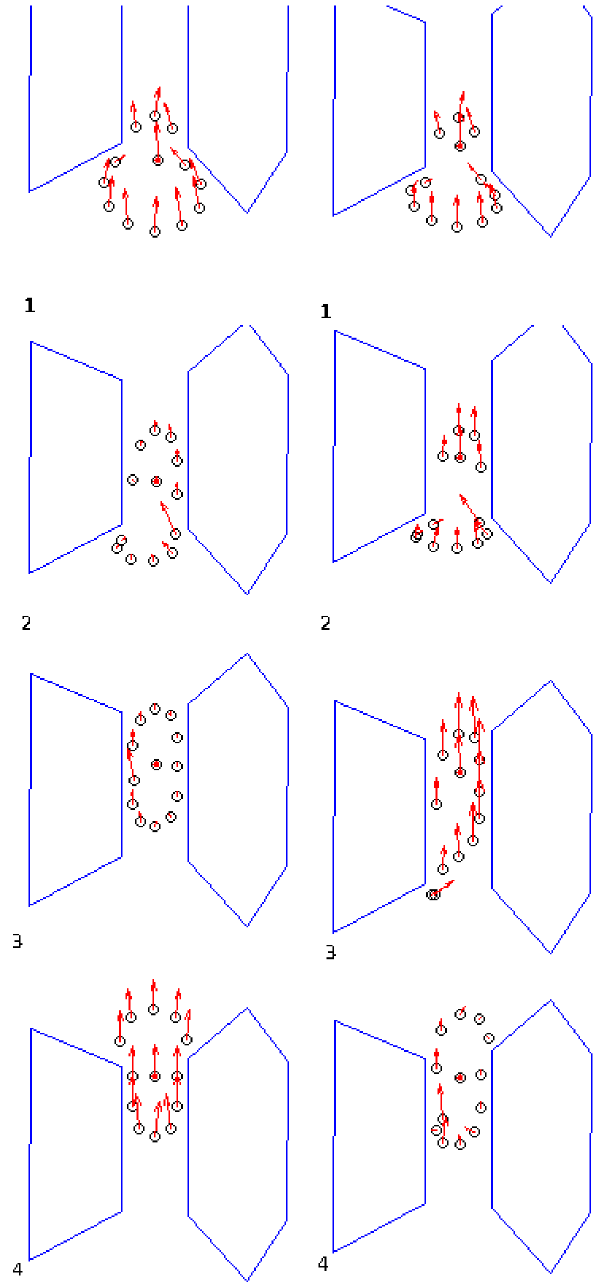

Figure 6. A group of vehicles moving between 2 obstacles. Vehicle velocity is shown as arrows and the center, solid node represents the virtual leader. An advancement coefficient of $\alpha = 0.3$ (left) provides better formation cohesion through obstacles compared to a coefficient of $\alpha = 1.5$ (right).

7.2. Player/Stage

The Player/Stage robotics platform is a software system developed to provide both a fully functional robotics server (Player) and a simulation engine (Stage), both of which communicate through a common interface but are separate software modules. The Player/Stage platform allows robotics control routines to be written using Player's client libraries and simulated in the Stage environment, allowing a variety of techniques to be tested before they are implemented on physical robotic systems. Because the Player server is independent of the simulation environment, control software should be capable of being transferred directly from simulation to any supported robotics platform.

The Player/Stage simulations were conducted to show that our control methodology could be implemented using a realistic robotic simulator modeling differential drive robots in complicated object environments. Further work will include the experimental demonstration of this framework on a number of small autonomous vehicles. Therefore, it is beneficial for testing purposes to simulate the actual control software that can be used in future experiments. Fig. 7 outlines the group movement as modeled in the Stage environment through a series of obstacles.

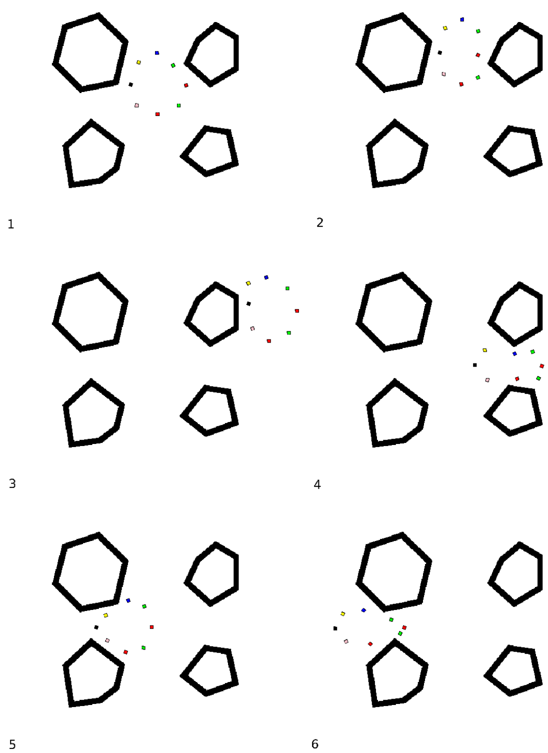

Figure 7. A group of vehicles moving between 4 obstacles as modeled within the Player/Stage simulator.

8. CONCLUSIONS

This paper presents a computationally lightweight method of planning the path of multiple autonomous vehicles moving in a flock formation, using artificial potentials, virtual leaders, geometrical object modeling, and proportional feedback in position computation. This technique has been proven in simulation studies to provide robust formation control when applying basic limitations to the artificial potentials used.

As our lightweight planning method was adapted to realistic vehicle models several interesting results were noticed in our simulations. Group performance can be increased by advancing the virtual leader in a manner that flows in relation to the dynamic capabilities of the group as shown in Section 6. Also, because of the sensor refresh rate limitations of the system additional steps must be taken to ensure that as different group forces are combined they can not put a vehicle on a collision trajectory with other group members or any environmental obstacles.

Control is still distributed such that each vehicle determines its behavior based on low bandwidth information from the other vehicles. These control techniques are both extremely robust and easily implemented, and should provide a realistic solution to group formation control and coordination, including applications in planetary exploration. Current limitations in the framework include certain obstacle types that split the flock into sub-flocks. Group cohesion is still an issue of further exploration. Future work will include experimental demonstration of this framework on a number of small autonomous vehicles. Successful experimental results will require that objects can be detected and modelled properly and that this information, including the location of the virtual leader and each group member, can be communicated in real time. The lightweight nature of our control methodology should allow it to be implemented with existing localization and mapping algorithms, allowing it to be developed in parallel to other navigational advancements.

REFERENCES

[1] Spears, W., Spears, D., Hamann, J., and Heil, R. Distributed, physics-based control of swarms of vehicles. *Autonomous Robots*, 17:137–162, 2004.

[2] Elkaim, G. H. and Siegel, M. A lightweight control methodology for formation control of vehicle swarms. In *IFAC*, Prague, Czech Republic, July 2005.

[3] Reynolds, C. Flocks, birds, and schools: A distributed behavioral model. *Comput. Graph.*, 21:25–34, 1987.

[4] Okubo, A. aspects of animal grouping: Swarms, schools, flocks, and herds. *Adv. Biophys*, 22:1–94, 1986.

[5] Warburton, K. and Lazarus, J. Tendency-distance models of social cohesion in animal groups. *J. Theoret. Biol*, 150:473–488, 1991.

[6] Flierl, G., Grunbaum, D., Levin, S., and Olson, D. From individuals to aggregations: The interplay between behavior and physics. *J. Theoret. Biol*, 196:397–454, 1999.

[7] Leonard, N. and Friorelli, E. Virtual leaders, artificial potentials and coordinated control of groups. In *IEEE Conf. Decision Control*, Orlando, FL, 2001.

[8] Khatib, O. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5:90–98, 1986.

[9] Latombe, J. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, USA, 1991.