A Lightweight Formation Control Methodology for a Swarm of Non-Holonomic Vehicles

Gabriel H. Elkaim and Robert J. Kelbley
Autonomous Systems Lab
Department of Computer Engineering
University of California, Santa Cruz
1156 High Street, Santa Cruz, CA 95064, USA
831-459-3054
elkaim@soe.ucsc.edu
rkelbs@soe.ucsc.edu

Abstract— Multi-vehicle swarms offer the potential for increased performance and robustness in several key robotic and autonomous applications. Emergent swarm behavior demonstrated in biological systems show performance that far outstrips the abilities of the individual members. This paper discusses a lightweight formation control methodology using conservative potential functions to ensure group cohesion, yet requiring very modest communication and control requirements for each individual node. Previous efforts have demonstrated distributed methods to navigate a vehicle swarm through a complex obstacle environment while remaining computationally simple and having low bandwidth requirements. It is shown that arbitrary formation can be held and morphed within the lightweight framework. Simulations of the lightweight framework applied to realistic non-holonomic tricycle vehicles highlight the swarm's ability to form arbitrary formations from random initial vehicle distributions and formation morphing capabilities, as well as navigate complex obstacle fields while maintaining formation. The non-holonomic constraints are used to implement realistic controls.

TABLE OF CONTENTS

1. INTRODUCTION

Autonomous vehicle groups or swarms provide increased performance and robustness in many different applications often allowing the group to solve problems currently difficult using individual robots. Biological swarms readily provide examples where the swarm achieves meta-behaviors that are beyond the capabilities of the individuals. A variety of applications have been identified that require precise formation control with a quick response to environmental disturbances. Data gathering using distributed meshes of data acquisition nodes has many applications to different scientific areas. In essence, the placing of the nodes is an antennae design problem using a sparse set of antennae patched to maximize the signal to noise ratio for a given phenomenon. Exploration is yet another area that would benefit from group navigation using robust and autonomous formation control. Problems requiring simultaneous, coordinated motion without exceedingly complex planning algorithms has motivated group coordination and formation control algorithms as an area of autonomous systems research.

Rule-based, distributed group motion control and their emergent behavior was first identified in 1986 by Craig Reynolds, when he showed homogeneous animal motion can be created using computer graphic models based on the behavior of schooling fish and flocking bird[9]. Mathematical biologists have used rule-based techniques to model emergent swarm behavior using different rules of repulsion and attraction between neighbors [8],[11],[4]. Similarly, controls researchers have used these cohesion and repulsion rules to guide groups of homogeneous vehicles and obtain formations similar to those found in nature.

The control methodology described in this work was developed using a simplified liquid surface tension abstraction while restricting computational and communications requirements to provide a realistic implementation model. It is our intent to mimic the macroscopic characteristics of liquid surface tension behaviors observed from the flow of liquids over a non-smooth surface. In this implementation, different artificial potential functions are used to achieve group cohesion, separation, and alignment. An equilibrium position for each vehicle node can be found by allowing the vehicle nodes to follow a steepest decent towards the geometric point at which the sum of these virtual forces vanishes.

This simple navigation framework for groups of autonomous

0-7803-9546-8/06/$20.00/©2006 IEEE
IEEEAC paper #1096

vehicles facilitates formation creation, group cohesion, and formation morphing for a swarm of vehicles. It greatly simplifies the task of swarm navigation by harnessing the potential of swarm interaction; the group can easily move around its environment while avoiding collisions with obstacles and between group members. A variety of different formation control and coordination techniques have been proposed as research in swarm behavior has developed [10]. However, as control techniques become increasingly complex, their implementations become more difficult and computational and communication demands for group members may become prohibitive.

This work provides further extensions for a previously discussed lightweight formation control methodology [3],[2] designed for applications of medium sized vehicular groups and can be applied for groups of all size with varying formation constraints. This lightweight methodology uses conservative energy potentials in artificial relationships between group vehicles, a *virtual leader*, and their environment. This framework has been shown to provide reliable, robust control with minimal computational and communications costs. Obstacle avoidance is obtained through the simplification of obstacle environments by bounding obstacles in convex polygons. Current testing has been advanced using realistic kinematic models of non-holonomic vehicles. Simulations thus far have shown these techniques to be extremely robust and easily implemented, providing a realistic solution to group formation control, formation morphing, and coordination.

The virtual leader abstraction is used to advance the group through its environment. Note that the virtual leader is not a physical vehicle, but rather an imaginary point used as a guide for group movement. Virtual leader is constructed similarly to Leonard[7] from whom we have adapted the term. The great benefit of the virtual leader abstraction is that trajectories for the individual nodes are not required, rather only the path of the virtual leader (VL) need be planned through areas of interest for the entire team, thus greatly simplifying the process.

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 [5],[6]. 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.

Communication requirements for the group can be varied depending on the precision needed for vehicle control. Each node updates its trajectory based on the locations of other group vehicles, obstacles, and the virtual leader. Obstacle information needs only be broadcasted to group members once, and requires modest data storage requirements because of the convex polygon encapsulation used. The position of the virtual leader will at most have to be transmitted to each group member once per update, assuming one node is responsible for recalculating the virtual leader's position. Bandwidth requirements for all group members to share their position amongst other group members is very modest for smaller groups (less than 20) because only the longitude and latitude values are needed. For larger groups, position information can be limited to vehicles within a neighboring radius surrounding each node. Simulation results show that neighboring vehicle information is adequate for less precise formation requirements. The update rate for vehicle reference positions can also be varied depending on the bandwidth requirements of specific applications.

Due to vehicle motion limitations, however, potential functions can sometimes work against each other in ways that force a given vehicle onto an undesired course. For example, a group must spread out lengthwise when navigating through narrow openings between obstacles, but as the formation narrows the neighboring vehicle potential function forces 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; unfortunately the vehicles may not be able to adjust their heading and velocity quickly enough, resulting in under-damped 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.

Additional difficulties occur as vehicles traverse through narrow openings between objects; the velocity of the group's virtual leader must be adjusted to ensure its force on lagging vehicles does not exceed a reasonable value. Thus the virtual leader's velocity should be reduced 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 excessive total forces in the virtual leader's direction of motion. This method ensures that virtual leader control mimics the dynamics of the swarm vehicles being controlled.

2. FORMATION FORCES

Virtual Leader Forces

Dynamically generating control forces for smooth group navigation is accomplished using artificial potential functions. This is modeled on a liquid water abstractions, such that the swarm can flow around the obstacles easily while maintaining cohesion. The virtual leader potential functions is simply a function of each node's position and the virtual leader's current placement. An initial position for the virtual leader can be calculated by finding the center of mass for the entire group. The initial desired group formation locations must be the equilibrium positions when calculating group forces, so placing the virtual leader at the groups center is a good practice. The center of mass location for N vehicles is defined as:

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

The initial distance d_0^{VL} between each node and the virtual leader then defines the equilibrium distance for each vehicle with respect to 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 will act to direct each group member back into their relative positions. These virtual leader forces, F_{VL}, can be defined 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 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.

Inter-Vehicle Forces

The artificial potential functions that produce vehicle-to-vehicle forces are based on the nominal distance from each vehicle to the rest of the swarm. These artificial vehicle to vehicle interactions are conservative forces that attract vehicles together as their distance increases and repels vehicles as their distance decreases. A new spring constant value K_{ij} is used, and now inter-vehicle forces, F_{ij}, between two vehicles can be defined 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.

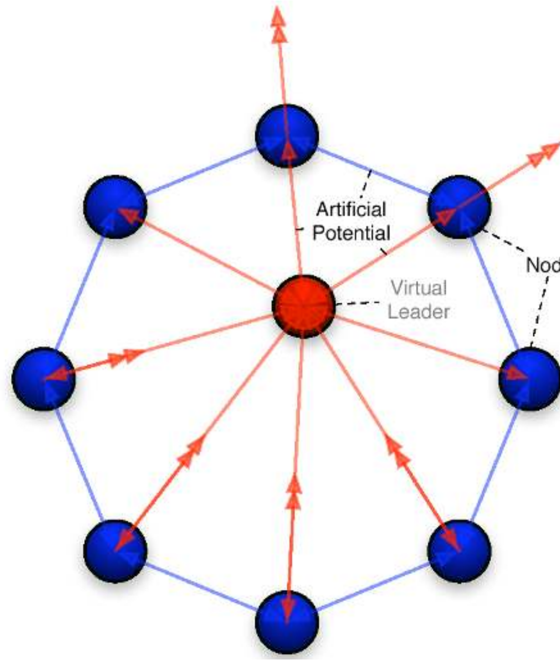

Figure 1. As the virtual leader is advanced, artificial potentials for the group react to force the formation back to equilibrium.

Obstacle Forces

Obstacles should produce a repulsive force to keep group vehicles from colliding with it, and allow the group to traverse smoothly between multiple obstacles. To simplify obstacle force calculations, obstacles are enclosed in bounding convex polygons. The obstacles then impose a repulsive force relationship between the vehicles and polygon edges that is inversely proportional to the distance between them. In this work, no attempt at obstacle detection is made, instead, it is assumed that knowledge of obstacle position and shape is available.

Enclosing obstacles in convex polygons has several key 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 [3], a simple coordinate translation and rotation 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 side of an obstacle, the distance is simply the Euclidean norm to 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 increases as d_k^{ob} decreases. To provide a more aggressive 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.

3. FORMATION CONTROL

Initial Formations

Assuming vehicles are distributed at random within their environment, formation positions must be assigned in a way that reduces the possibility of inter-vehicle collisions and allows nodes to transition to their new positions efficiently. In order to accomplish this, the equilibrium condition for the desired initial formation is calculated (the geometry of the initial formation, as desired). These points need to be matched to each node in its current position such that each vehicle is assigned a location within the desired formation. The goal is to match these initial and final positions in such a way as to make the initial paths into the formation achievable.

For each node initially placed at random, the closest position from the set of available desired formation positions is found. Of all of the nodes, the node which is farthest from its desired position is paired with that position, and both are deleted from the set. This iteration is repeated until all nodes have been assigned a position within the desired formation. This method allows the vehicles to spread out uniformly to their new positions, and results in collision-free trajectories into the initial formation.

Note that the natural formation configuration given the liquid water formulation is a circle of vehicles whose radius grows with the number of nodes, and simply leaving the virtual leader stationary while iterating from a random initial node placement will bring the swarm into this initial formation.

Formation Morphing

Arbitrary formations can be formed other than the ring formation. In order to hold these arbitrary formations, a set of formation forces is imposed upon the swarm in order to shift the equilibrium points of the vehicles to their new position. That is, the nodes are placed in the desired formation, and the inter-vehicle forces and virtual leader forces are computed. The sum of these forces is recorded for each node in the formation. The formation forces are simply equal and opposite to those calculated, and result in each node being held at an equilibrium point.

Once a stable formation has been obtained, the swarm can easily be morphed into new formations by using an additional force F_{morph}, which is nothing more than the formation force discussed above. These forces can be computed for any number of arbitrary formations, and used to change from one formation to another very simply and easily. These forces can then be added to the previous force summation equation (11) which results in:

$$F_{tot} = \sum F^{ij} + F_{VL} + F_{ob} + F_{morph} \qquad (9)$$

In order to change between two formations, we simply allow the F_{morph} to be a linear combination of the current and the next formation, and allow the nodes to find their own equilibrium. That is:

$$F_{morph}^{tot} = x * F_{morph}^{cur} + (1 - x) * F_{morph}^{next} \qquad (10)$$

where x varies from 0 to 1. As long as x is increased slowly from 0 to 1, then the formation will morph from the current to the next. Note that this can be done at any time, with the formation moving or stationary, and will result in collision free paths for each node to get into the next formation.

Morphing Simulations

Tricycle steering kinematic models were used to simulate vehicle movement for formation morphing testing. After vehicles were placed randomly within the environment, positions were assigned as previously described in the initial formation discussion. For this simulation, a circular formation was initially chosen. Once the vehicles assembled in a circle, the morphing procedure was used to change to a single file line, then a pyramid, and finally the box formations shown in Figure 2.

4. COLLISION PREVENTION

These virtual leader, inter-vehicle, and obstacle forces can then be used as the basis for group formation control. The resulting forces can be summed together to find the total force acting on a vehicle as:

$$F_{tot} = \sum F_{ij} + F_{VL} + F_{ob} \qquad (11)$$

The magnitude and direction of F_{tot} can then be used as the new desired control reference for each vehicle, knowing that reaching this objective should restore equilibrium to the group. It is important to note the position at which the forces go to zero becomes the new reference target for individual vehicles and that each vehicle is then given that new open loop feed forward trajectory, implemented in a receding horizon

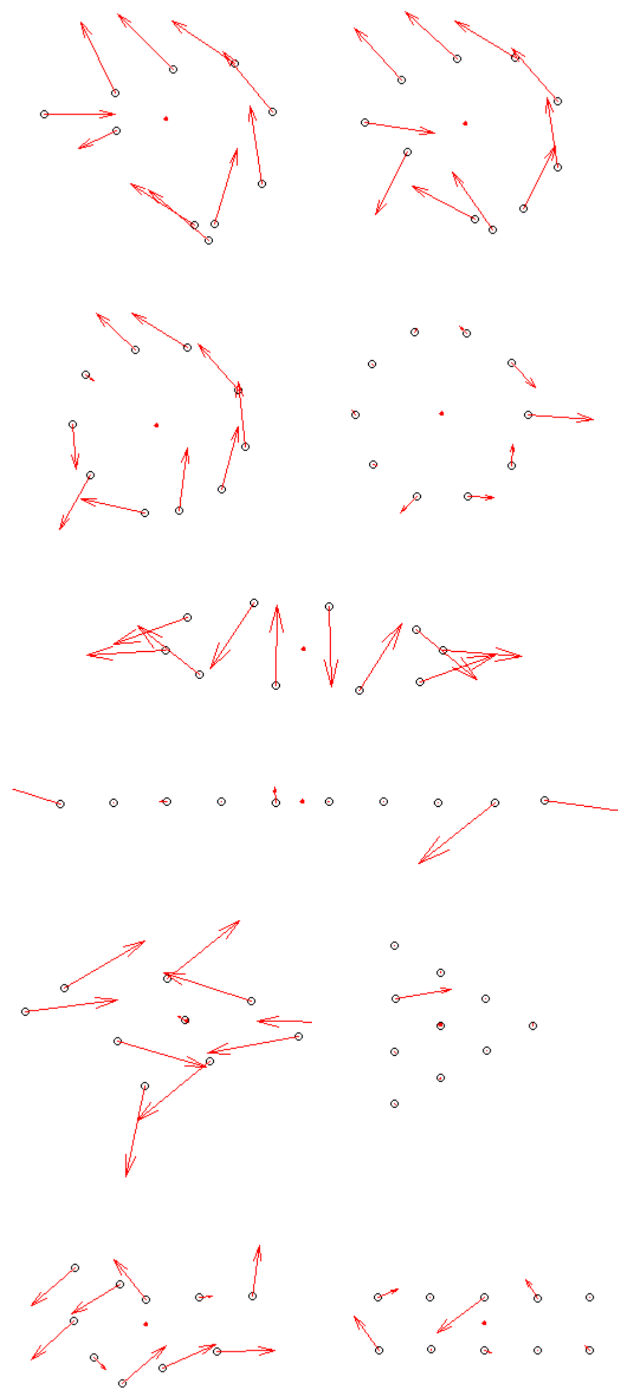

Figure 2. A group of vehicles moving to a circular formation after an initial random distribution, then changing formations to a line, pyramid, and box.

manner. The resulting force F_{tot} is not simply an acceleration or velocity reference signal, those vehicle specific signals are derived from a separate closed loop controller that navigates the vehicle to its new reference position in the vehicle's coordinate system.

Using F_{tot} to direct the vehicles to equilibrium positions within their environment effectively acts as a damping unit for the system of conservative potential functions. This prevents the system from experiencing sustained oscillations that exists in undamped spring/mass systems.

As the group flows to reach equilibrium, some vehicles may find their new trajectory calculations could cause a collision between other vehicles or obstacles. This is implemented in a receding horizon manner, such that new trajectories are calculated continuously on-line before the vehicles have reached their new target points; the vehicles eventually flow to an equilibrium point at which the sum of all virtual forces is zero, preventing any collisions.

In reality, physical vehicles have motion limitations and navigation system update limits. These limits could cause obstacle collisions if a group member is unable to update its trajectory information quickly enough to recover when forced in undesired directions. A simple way to accommodate vehicle dynamics is to prevent vehicles from being forced into these undesired positions by saturating the forces using a function of vehicle maximum velocity, trajectory refresh rate, and virtual spring constants.

A computationally simple way to limit control forces and prevent these collisions is to define a threshold distance around all obstacles that ensures every vehicle has sufficient time to adjust and recover 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:

$$\left|\sum F_{ij} + F_{ia}\right| \leq \frac{K_{ob}}{d_{max}} \tag{12}$$

When traversing narrow gaps between obstacles, the group may have to compact into a skinny formation. This could result in vehicles from opposite sides of a formation to both generate colliding trajectory calculations due to the forces from the opposing obstacles. Again, the solution is to place a saturation limit on the sum total of the vehicle forces, that is:

$$|F_i| = \min\left(|F_i|, |F_{max}|\right) \tag{13}$$

where F_{max} is the saturation limit on the total virtual force. These two heuristic limits have been shown in simulation to prevent any inter-vehicle collisions.

5. VIRTUAL LEADER ADVANCEMENT

The virtual leader forces provide the basis for group navigation between desired way points. To ensure stable group movement, the virtual leader should be advanced dynamically to prevent excessive effects on any of the group's vehicles. When traversing through obstacles the group must reduce its speed to allow all members to squeeze through smaller openings. Initially, all group members slow as the vehicles at the head of the formation compact to traverse the opening. These leading vehicles will then, however, accelerate ahead of the remaining group, now free of the previous obstacle forces. If the virtual leader's velocity is not adjusted, vehicles in the rear of the formation will lag behind and the artificial forces acting on them will increase rapidly. The virtual leader should therefore be advanced in a way to keep it consistent with dynamic group behavior.

To achieve good virtual leader advancement, 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 4, now only forces in the same direction as the virtual leader's motion are of concern. 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 rear of the formation. Thus the virtual leader should only be advanced such as to ensure that the following relation holds:

$$F_{tot} \cdot \hat{d_{VL}} \leq \alpha d_{step} \tag{14}$$

where d_{step} is the maximum distance a vehicle can travel during the given formation refresh time, 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.

The general methodology for group advancement is as follows. First, all of the forces (obstacle, inter-vehicle, virtual leader) upon the vehicles in their current position are calculated. Secondly, the vehicle positions at which the sum of these forces goes to zero is found. This point is then used as the reference position for each individual vehicle, which is then advanced towards this target using a simple kinematic controller that controls only that vehicle. This entire process is repeated before the vehicle reaches its reference position in a receding horizon manner. A simulation of this method is shown in Figure 3 which shows the virtual leader maintaining its proximity to a group in pyramid formation as it traverses between obstacles.

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 reference points. 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.

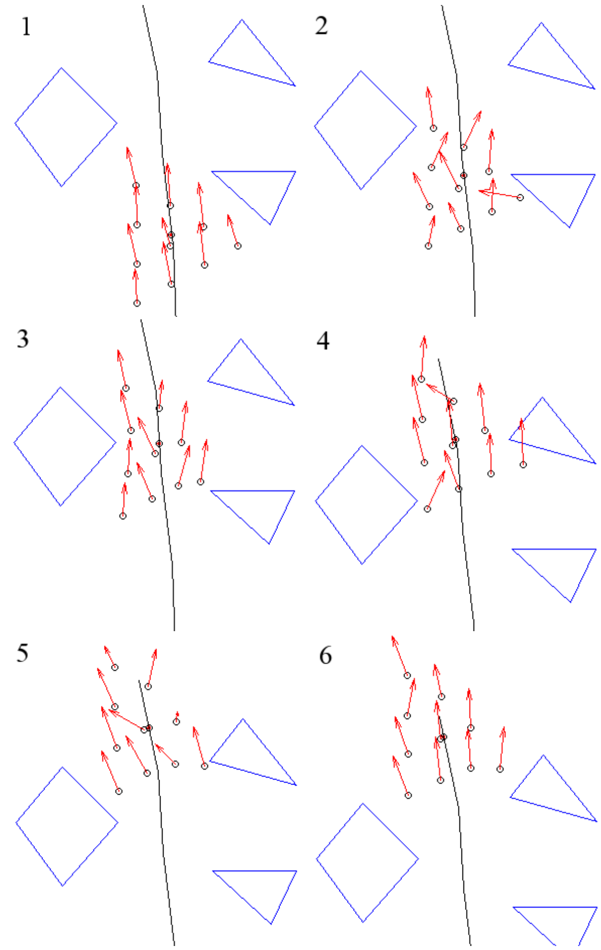

Figure 3. A group of vehicles guided by the virtual leader (red dot), moving between 3 obstacles. Vehicle velocities shown as red arrows.

6. Non-Holonomic Constraint Adjustments

All physical systems have motion constraints that must be addressed to achieve precision control requirements. A tricycle model has been used as a realistic kinematic model in previous simulations, with a simple PID controller used to track the reference trajectory. It is well known that a non-holonomic tricycle model is in fact a Brocket integrator, and cannot be simply controlled to an area within the minimum turn radius on either side of the vehicle. Previous work has been done in adapting to these limitations, typically to allow parallel parking capabilities for autonomous vehicles as discussed in [1]. There are many, many ways in which to solve this problem, in this work we attack this limitation with a simple heuristic method that works quite well in simulation.

This is important to the formation morphing problem, because often when the formation is changing, the equilibrium position for a given node is within the unreachable area. A common approach is to use two inverse sinusoid trajectories. We use a similar technique in our implementation to allow our vehicles to reach points located to either side of the vehicles; the vehicle first drives forward (assuming no obstacle) and then backwards into the desired position. A cosine function is used to angle the vehicle as it drives forward towards the target position. When it has driven forward far enough, the inverse of the same cosine trajectory is used to back the vehicle into it's desired position as shown in Figure 4. In the case of an obstacle to the front, the order is reversed and the vehicle first drives rearwards, and then forwards along the two cosine curves. The PID controller is used as a feedback around the feedforward trajectory generated by the cosines.

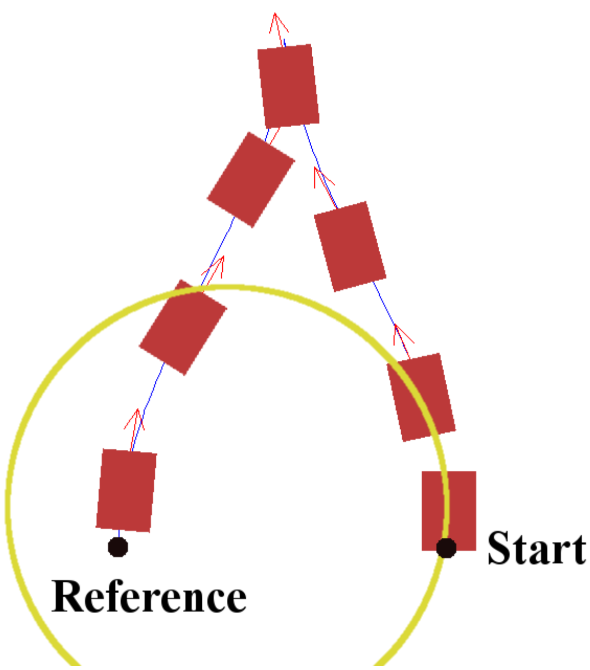

Figure 4. Blue line shows trajectory used by tricycle steering model (red squares) to reach reference positions inside its turning radius (yellow). The vehicle's heading is shown by the red arrows.

7. CONCLUSIONS

A computationally lightweight methodology that facilitates group navigation for multiple autonomous vehicles in a flock formation has interesting applications for scientific data gathering and exploration. The lightweight methodology consists of artificial potentials for node movement, a virtual leader for group path planning, and geometric object modeling for obstacle avoidance and has been shown to provide robust formation control with the additional of previously discussed techniques .

By addressing realistic node kinematic, computational, and communications restrictions, this lightweight framework has shown in simulations to provide good formation control results for medium sized groups of non-holonomic ground vehicles. It has been shown that these vehicles can be organized into different formations from a random initial placement. Formations can then be switched (or morphed) dynamically by adding using additional forces to move individual vehicles into their new equilibrium position.

As our lightweight planning method was adapted to realistic vehicle models several interesting results were noticed during simulations. Group performance can be improved by advancing the virtual leader in a manner that flows in relation to the dynamic capabilities of the group as shown in Section 5. Some care must be taken with the formation refresh rate in order to ensure that as the forces are combined they do not put a vehicle on a collision trajectory with other group members or any environmental obstacles.

The general methodology for group advancement is as follows. First, all of the forces (obstacle, inter-vehicle, virtual leader) upon the vehicles in their current position are calculated. Secondly, the vehicle positions at which the sum of these forces goes to zero is found. This point is then used as the reference position for each individual vehicle, which is then advanced towards this target using a simple kinematic controller that controls only that vehicle. This entire process is repeated before the vehicle reaches its reference position in a receding horizon manner.

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 UAV swarms. While the analysis had been performed in 2D, the work generalizes out to 3D quite readily. 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 ground vehicles. Successful experimental results will require that objects can be detected and modeled properly and that this information, including the location of the virtual leader and each group member, can be communicated in real time. The simplified nature of our control methodology should allow it to be implemented with existing localization and mapping algorithms, allowing parallel development with other navigational advancements.

REFERENCES

[1] R. W. Brockett. Asymptotic stability and feedback stabilization. *Differential Geometric Control Theory*, pages 181–191, 1983.

[2] G. H. Elkaim and R. J. Kelbley. Extension of a lightweight formation control methodology to groups of autonomous vehicles. In *iSAIRAS*, Muchen, Germany, September 2005.

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

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

1999.

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

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

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

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

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

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

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

Gabriel Elkaim *received the B.S. degree in Mechanical/Aerospace Engineering from Princeton University, Princeton, NJ, in 1990, the M.S. and Ph.D. Degrees from Stanford University, Stanford, CA, in Aeronautics and Astronautics, in 1995 and 2002 respectively. In 2003, he joined the faculty of the Computer Engineering department, in the Jack Baskin School of Engineering, at the University of California, Santa Cruz, Santa Cruz, CA, as an Assistant Professor. His research interests include control systems, sensor fusion, GPS, system identification, and autonomous vehicle systems. His research focuses on intelligent autonomous vehicles, with an emphasis on robust guidance, navigation, and control strategies. Specifically, he has founded the Autonomous Systems Lab at UC Santa Cruz, and is currently developing an autonomous wing-sailed marine surface vehicle and off-road autonomous ground vehicles.*

Robert Kelbley *is a graduate student at University of California, Santa Cruz. His research interests include group formation control and autonomous surface vehicles. Rob obtained a B.S. degree in Computer Engineering from Ohio Northern University, and is currently pursuing his Master's degree.*