2004 American Control Conference
http://www.cds.caltech.edu/~murray/papers/2004e_jin+03-acc.html

MVWT-II: The Second Generation Caltech Multi-vehicle Wireless Testbed

Zhipu Jin♮*, Stephen Waydo*, Elisabeth B. Wildanger*, Michael Lammers*,
Hans Scholze*, Peter Foley*, David Held**, Richard M. Murray***

***Abstract*— The Caltech Multi-Vehicle Wireless Testbed is an experimental platform for validating theoretical advances in multiple-vehicle coordination and cooperation, real-time networked control system, and distributed computation. This paper describes the design and development of an additional fleet of 12 second-generation vehicles. These vehicles are hovercrafts and designed to have lower mass and friction as well as smaller size than the first generation vehicles. These hovercrafts combined with the outdoor wireless testbed provide a perfect hardware platform for RoboFlag competition.**

I. INTRODUCTION

The Caltech Multi-Vehicle Wireless Testbed (MVWT) [1] is a tool for validating theoretical advances in multiple-vehicle coordination and cooperative control, networked control systems, real-time networking and high confidence distributed computation. The first-generation MVWT vehicles consist of a laptop computer mounted to a chassis that rolls on three omni-directional casters, with a pair of model aircraft ducted fans for actuation. A unique feature of this testbed is that the vehicles are underactuated and exhibit nonlinear second-order dynamics. These nontrivial dynamics force us to actively stabilize the vehicles while also trying to accomplish cooperative tasks in a manner analogous to the operation of Uninhabited Aerial Vehicles (UAV's). The MVWT vehicles run in a laboratory environment with localization achieved using an overhead camera system called the Lab Positioning System (LPS).

While the MVWT has proven useful in experimentally verifying theoretical results in nonlinear and cooperative control, several factors have limited its utility. First among them is the size of the vehicles relative to the laboratory space available. Experiments with more than 3 or 4 vehicles running on the floor at the same time have proved crowded and difficult to conduct. The use of an overhead vision system, while convenient from an implementation standpoint, has limited us to running the vehicles only within the laboratory and thus has prevented us from using larger spaces for more complex experiments, such as the RoboFlag competition [3].

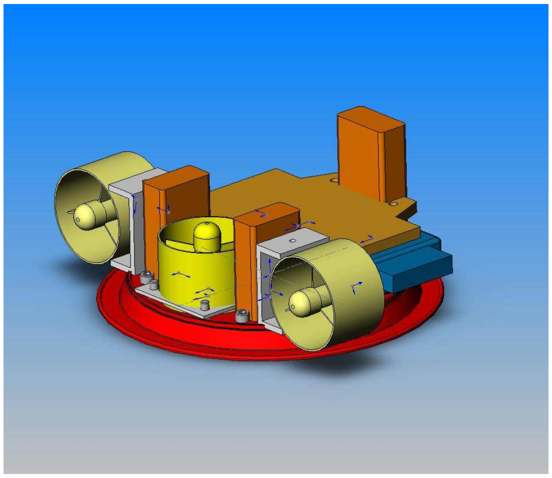

Fig. 1. Design of MVWT-II Hovercraft

During the summer of 2003, several graduate and undergraduate students worked together at Caltech to develop an additional fleet of 12 second-generation vehicles designed to address these issues which we call the “MVWT-II”. The paper describes their work and is organized as follows: In Section 2, we briefly talk about the RoboFlag game and the design consideration of the second-generation vehicles. Section 3 lists the details of the hovercraft development including the mechanical design, embedded electrical system, and simple local controller. Section 4 gives the new hovercraft's parameters and performance measurements. Summary and future work are discussed in Section 5.

II. MOTIVATIONS AND DESIGN CONSIDERATION

The RoboFlag competition [3] is a powerful opportunity to use the MVWT as an experimental platform for research challenges in distributed control, sensor fusion, and human-centered control in a realtime dynamic environment. The RoboFlag game has been formulated at Cornell University over the last few years. It uses more complex scoring rules and a more specialized field than RoboCup. Roughly speaking, this game is based on “capture the flag”. Two teams of 6 to 12 robots commanded by 1 or 2 human players play the game. The number of robots depends on the playing field size and the game complexity. Each team tries to attack the other team's territory, capture the other team's flag and bring it back to its own home zone. Because of the realtime dynamic environment, the complex offense/defense strategies, and the cooperation between robots, this game

---

This work was partially supported by DARPA program

♮Corresponding author, Electrical Engineering, California Institute of Technology, Pasadena, CA 91125, USA `jzp@caltech.edu`

*Graduate and undergraduate students of MVWT-II project, California Institute of Technology, Pasadena, CA 91125, USA

**Undergraduate student of MVWT-II SURF project, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

***Professor of Mechanical Engineering, California Institute of Technology, Pasadena, CA 91125, USA

helps us to understand some fundamental issues in realtime and high confidence distributed control.

Caltech and Cornell have worked on the RoboFlag competition together since 2001. In 2002, two groups of undergraduate students, team Pasadena and team Ithaca from Caltech and Cornell University respectively, joined the RoboFlag Summer Undergraduate Research Fellowship program (SURF). They spent ten weeks together developing a software system, fully completing the rules, and designing defence/offence strategies. At the end of the summer, they successfully competed with each other three times. Based on their experience and feedback, RoboFlag needs a larger, easily configurable playing field with faster, smaller vehicles to increase the feasibility and challenge. At the same time, vehicles should have second order dynamics so that the coordination control algorithms we develop will greatly rely on the advanced control techniques. With this motivation, a hovercraft design was developed in the summer of 2003 for MVWT-II vehicles shown in Fig. 1.

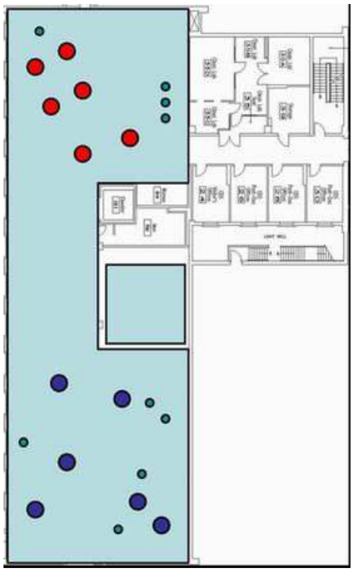

Fig. 2. Outdoor testbed on the roof. The small square in the middle is the original MVWT, the large area is the roof testbed.

There are couple of design considerations. First, the vehicles can run outdoor since developing a outdoor testbed will be a good solution to get a large playing filed. Fig. 2 is the outdoor testbed on the roof which is under development at Caltech. This will give us approximately 10 times more area than the indoor MVWT. The positioning system can be differential GPS or infrared positioning system (IR). Second, to reduce mass and friction and thus increase performance relative to the original MVWT vehicles, we need a very compact design which can enable us to reduce the mass greatly while improving the control authority (in terms of thrust/weight and torque/moment-of-inertia ratios) and reducing friction to nearly zero. Third, each vehicle should have an independent computation unit by which we can implement certain local controllers. Finally, we need to keep the cost of each hovercraft as low as possible since we will build a fleet with 12 vehicles. Also, how to make the assembly and maintenance job easy is another important issue for large number of vehicles.

III. DESIGN AND IMPLEMENTATION

A. Mechanical Design

The basic concept behind the new vehicles is the same as for the original MVWT vehicles. There is a forward-facing thrust fan on either side of the vehicle, and the vehicle is free to move in all directions. Beyond this, however, the two designs diverge dramatically. The previous MVWT vehicles were used on a smooth plastic surface, and the casters on the bottom were able to decrease the friction to a useable level. The new designs needed to be able to slide with very low friction over a surface that could be prepared on the roof of a building. Since the surface possibilities were quite limited, the only possible design was some sort of hovercraft.

a) Computation Unit: We selected Sharp Zaurus SL-5500 PDA as the computation unit. The dimensions of Zaurus is $74mm \times 138mm \times 18mm$. The weight is $212g$. Please refer to the electrical and computation issues discussed in Section III-B.

b) Skirtless Design: Considering the main design goals of low cost, simplicity, small size, stability, and ruggedness, the skirtless hovercraft design was motivated. While a skirt design could almost certainly give better performance, the labor consideration and financial constraint simply cut off this possibility. Also, after testing several prototypes with and without skirt, we found that the skirtless prototype worked very well.

c) Fans Selection and Location: While centrifugal fans are more efficient for the high-pressure, low-flow lift fan application, it was very hard to find one that was both powerful enough to lift the vehicle and compact enough to fit the space available. For this reason, an axial design was chosen, and the best fan for the application ended up being the same one used for thrust. The fans (GWS EDF-50), and the motor controllers (GWS ICS-100) were chosen with a focus on price, costing \$15 and \$20 per item, respectively. Each thrust fan gives approximately 0.7N thrust, which is enough to accelerate the vehicle quite quickly. To protect users and to keep foreign objects out of the fans, safety covers were fashioned from aluminum mesh and placed on the intakes to both thrust and lift fans. The dimensions of the Zaurus made overall layout rather difficult. We put the lift fan on one side of the plate and two thrust fans evenly to either side of the lift fan. The off-center lift fan keeps the Zaurus from hanging over the plate edges, protecting it in the event of inevitable collisions. Fig. 3 is the fan force map measured in MWVT lab.

d) Batteries Selection: The batteries used were rechargeable Lithium types, with one $1800mAh$ battery for the lift fan and two $950mAh$ batteries for thrust fans. The Lithium batteries are more expensive than NiMH or NiCd batteries, but they can hold much more power

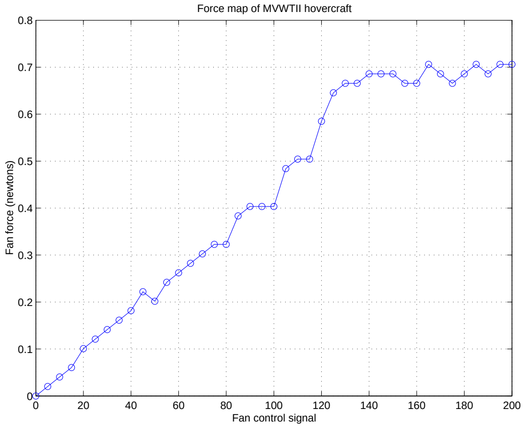

Fig. 3. Force map of the hovercraft fans

considering their size and weight. Since the batteries make up a substantial portion of the total vehicle weight, it was decided that the small size was worth the extra cost.

e) Balance: A major design issue for the hovercraft was balance, especially with the skirtless design and the off-center location of the fans. If the vehicle's base tilted, air would escape unevenly around the edges, and the craft would be propelled in the direction it tilted. Also it was originally thought that the off-center fan would cause uneven hovering height between front and back. These problems were largely avoided by strategically positioning batteries and by adding small brass weights to various bolts around the craft.

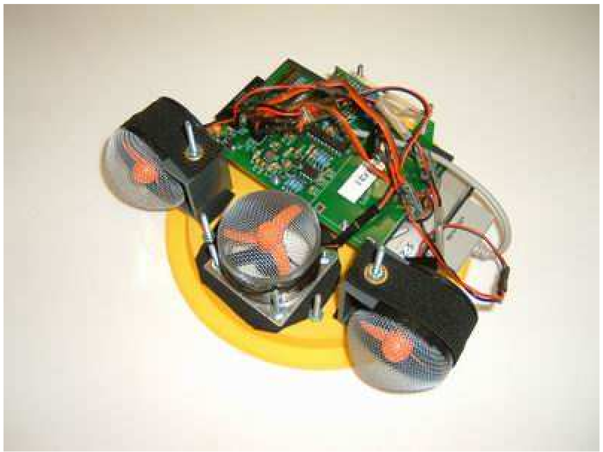

Fig. 4. MVWT-II hovercarft

The hovercraft was designed and modelled by using SolidWorks, which greatly eased vehicle layout. Also, it left us with very precise measurements of where to cut or drill. In order to drill holes quite accurately on the base plate and to decrease labor per vehicle, the base was drilled on a CNC machine. A jig was milled for drilling holes in the fan-mounting brackets. Since most of the time-consuming labor was automatic, the actual assembly was quite quick, consisting almost entirely of bolting parts together and connecting cables. Two people could build a vehicle from prefabricated parts in about 30 minutes. Altogether, the mechanical parts (including fans, speed controllers, and batteries) cost $300 per hovercraft.

B. *Embedded System*

The embedded computing system in the hovercraft is split into two sections. The Sharp Zaurus PDA provides wireless Ethernet connectivity and processing power to run simple local controllers onboard. The bridge between the Zaurus and the hovercraft fans is a custom interface board built around an Atmel micro-controller.

The Zaurus runs a custom version of Linux 2.4 on a 206MHz Strong ARM processor. It has 16MB FLASH and 64MB SDRAM. There are two main reasons to choose Zaurus. First, the Sharp Linux-based OS is easy to develop custom software with standard tools (there is a cross compiler of GCC available for the Strong ARM processor). Second, it is much cheaper than comparable specialized single-board computers, such as PC104. The Zaurus connects to the network in the vehicles lab through a wireless Ethernet card in the Compact Flash slot. The network connection allows the hovercraft to receive commands from more powerful controllers running on off board computers and communicate with other vehicles.

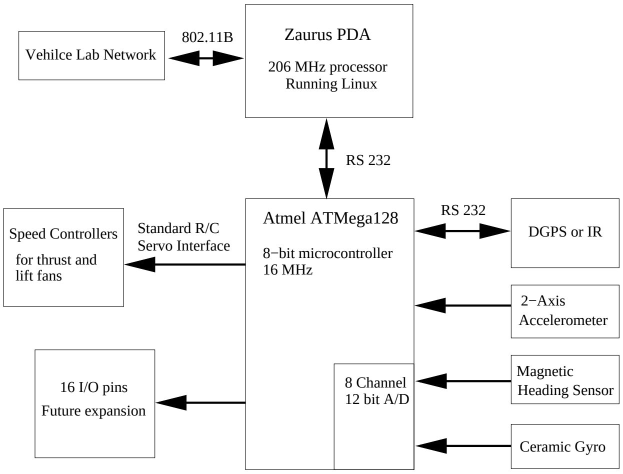

Fig. 5. Diagram of the embedded system

There is a limited amount of I/O available on the Zaurus, so all the low-level control of the hovercraft hardware is handled by an Atmel ATMega128 micro-controller. The Zaurus communicates with the micro-controller through one of ATMega's RS-232 serial ports.

The ATMega128 is an 8-bit RISC micro-controller running at 16MHz. It includes an 8-channel 12-bit A/D converter, 8 external interrupts, two RS-232 ports, and 4 timers/counters. The ATMega128 is incorporated into a custom PCB with a power supply, a ceramic gyro, a two-axis accelerometer, a magnetic heading sensor, a external module reserved for differential GPS or IR system, three outputs to the lift and thrust fan speed controllers, and 16 general purpose I/O lines for future expansion. The ATMega128 reads the sensors continuously and sends packets of sensor data back to the Zaurus at approximately 50 Hz. It also receives fan-speed control commands from the Zaurus and sets the speed controller outputs appropriately.

For developing the software of the ATMega128 we used the free WinAVR set of tools and Atmel's AVRISP programmer. WinAVR includes a port of gcc and a cross-compiler for the AVR architecture which runs under Microsoft Windows OS. The AVRISP is a simple programmer that allows in-circuit programming of the Atmel through the 6-pin ISP connector.

The gyro we are using is the Tokin CG-16D ceramic gyro, which is mounted vertically on the interface board, a product of the Coriolis effect on an internal vibrating ceramic column printed with electrodes. The output of the gyro is 1.1 mV/deg/sec $\pm 20\%$ referenced around 2.4V. The offset at zero angular speed can vary as much as ± 300mV. A simple differential amplifier is built to compensate this and modify the gyro output such that it can be easily processed by A/D converter. The gyro is linear in the range ± 840 degrees/s according to our tests and the Atmel can sample the gyro outputs with a frequency up to 100 Hz.

The accelerometer is an Analog Devices ADXL202 two-axis MEMS accelerometer which measures acceleration in the forward/backward and side-to-side directions. Its output is a pulse-width modulated (PWM) signal which is measured with the ATMega128 timer/counters. Every ± 1g acceleration corresponds to approximately $\pm 12.5\%$ duty cycle. The ADXL202 can measure ± 2g acceleration with a sample frequency of the accelerometer up to 100 Hz.

Absolute heading is measured by a heading sensor which is composed by a Honeywell HMC1052 two-axis magneto-resistive sensor, two A/D converter, and a compassing circuit. This sensor provides two analog outputs and can be used to calculate the heading with about 1.5 degree orientation accuracy. Currently, the interface board only returns the raw A/D data from each axis. The functions to calibrate the compass and calculate the orientation run on the Zaurus.

Absolute position is read from the module reserved for the differential GPS or IR system. The module interfaces through a serial port, which is connected to the second UART port on the ATMega128.

The lift and thrust fan speed controllers on the hovercraft are controlled through the standard R/C servo interface. The ATMega128 generates the PWM signal at 50Hz to set the speed of each fan. The interface board's electrical power comes from the battery of the lift fan.

C. Local Controller Design

The MVWT-II hovercraft has the same propulsion principle as the first-generation MVWT vehicle. According to [1], the equations of the hovercraft motion can be written as

$$\begin{aligned} m\ddot{x} &= -\mu\dot{x} + (F_R + F_L)\cos\theta \\ m\ddot{y} &= -\mu\dot{y} + (F_R + F_L)\sin\theta \\ J\ddot{\theta} &= -\psi\dot{\theta} + (F_R - F_L)r_f \end{aligned}$$

These equations are derived by observation from the simple schematic of the vehicle shown in Fig. 6 and are the same as motion equations of first-generation vehicle.

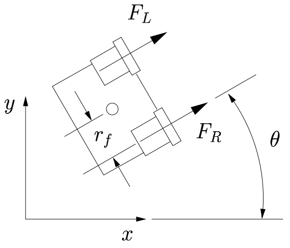

Fig. 6. Schematic of hovercraft. The coordinate frame is inertial and the forces F_L and F_R are applied at the fan axes.

These equations include four physical parameters: the mass m, the moment of inertia J, and the linear and rotational viscous friction coefficients μ and ψ. Linear and rotational friction coefficients μ and ψ depend on the the lift fan thrust and the surface of the field. This makes the the lift fan thrust another possible control force.

For the RoboFlag game, the local controller takes the state $X = (x, \dot{x}, y, \dot{y}, \theta, \dot{\theta})$ and a reference velocity $V = (v_x, v_y)$, calculates the desired thrust $F = F_R + F_L$ and torque $T = F_R - F_L$. (We assume the lift fan thrust is constant.) The equations to transfer everything into error coordinates which is represented by $(\theta_e, \dot{\theta}, \xi_1, \xi_2)$ are

$$\begin{aligned} \theta_e &= \theta - \tan^{-1}(v_y/v_x) \\ \xi_1 &= \frac{\dot{x}v_x + \dot{y}v_y}{\|V\|} - \|V\| \\ \xi_2 &= \frac{-\dot{x}v_y + \dot{y}v_x}{\|V\|} \end{aligned}$$

and the control law is

$$\begin{bmatrix} F \\ T \end{bmatrix} = -K \begin{bmatrix} \theta_e \\ \dot{\theta} \\ \xi_1 \\ \xi_2 \end{bmatrix}.$$

The 2×4 gain matrix K is calculated off-line for different controllers and the fan thrusts are

$$\begin{aligned} F_R &= 1/2(F + T) + F_0 \\ F_L &= 1/2(F - T) + F_0 \end{aligned}$$

where F_0 is a "feed-forward" force that should be re-configurable by on-line software or parameter files. In reality, however, the fans are saturation units which have the minimum thrust and maximum thrust. The minimum thrust is 0N, i.e. the fan cannot thrust thrust backwards. The maximum thrust is about 0.7N.

TABLE I

PARAMETERS OF FIRST-GENERATION VEHICLE AND MVWT-II HOVERCRAFT

Vehicle Parameters	**First-generation vehicle**	**MVWT-II hovercraft**
Plane Dimensions	rectangle with 37.00 cm ×27.00 cm	round disk with diameter 20.00 cm
Height	18.00cm	7.50 cm
Mass	5.05 ± 0.05 kg	0.75kg
Moment of Inertia	0.05 kg m^2	0.00316 kg m^2
Distance between Thrust Fans	24.6 cm	17.8 cm
Maximum Fan Thrust	5.1 N	0.7N
Linear Friction on Indoor Testbed	4.5 kg/s	0.15 kg/s with maximum lift fan thrust
Rotational Friction on Indoor Testbed	0.064 kg m^2/s	0.005 kg m^2/s with maximum lift fan thrust
Maximum Speed	1.2 m/s	$>$ 2.5 m/s
Unit Cost	More than $2000	Less than $860
Lift Fan Battery Lifetime	$20 - 25$ minutes	$35 - 40$ minutes
Hover Height	N/A	$< 2mm$
Computation Unit	Dell Latitude L400	Sharp Zaurus SL-5500
Processor	Intel 700 MHz	Strong ARM 206 MHz
Wireless Network	Yes	Yes
Onboard Sensor	Gyro	Gyro, accelerometer, and heading sensor
Positioning System	Overhead cameras	Overhead cameras, DGPS or IR

IV. PARAMETERS AND EXPERIMENTS

The final hovercraft design resulted in an effective yet low cost device. It was impossible to achieve a high hover height with the low cost limit in ten weeks, so the hovercraft were unable to operate on rough surfaces such as lawn or pavement. However, they are still effective for coordinated control tests on a relative smooth field such as MVWT, gymnasium, and building roof covered by BerberMax carpet padding.

Parameters of MVWT-II hovercraft and first-generation vehicle are listed in Table I. We conducted some performance tests on Caltech MVWT to obtain these data. The overhead vision system provided the necessary position for these experiments.

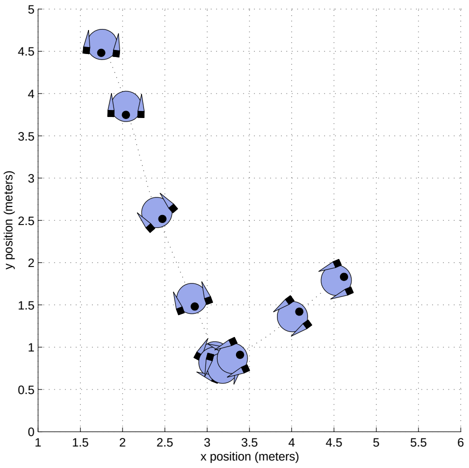

Fig. 7. A simple experiment of the hovercraft

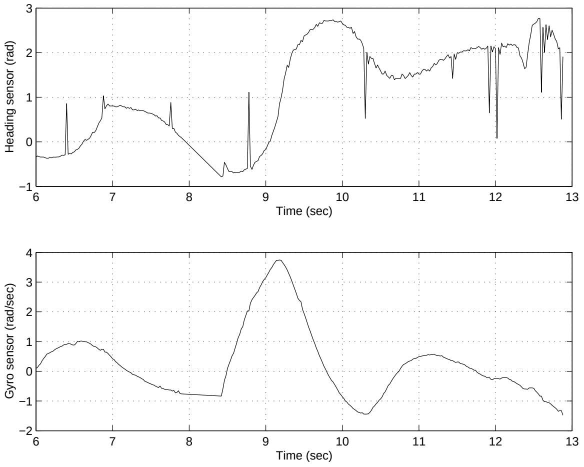

Fig. 8. Heading sensor data and gyro data

Fig. 7 shows a top view of a simple experiment. The hovercraft goes straight and then makes a left turn. Fig. 8 is the heading sensor data and gyro data. In this experiment, we use a simple PID controller for the heading and also the lift fan's output is controlled as a "brake" to slow down the hovercraft before it turns.

The heading sensor is based on a two-axis magnetic filed sensor and it easily suffers from the nearby magnetized devices such as CRT monitors, power supplies, etc. So it's not a good idea to use it in a small and crowded lab, but it can provide good orientation data in a open space such as a gymnasium or a outdoor field.

V. SUMMARY AND FUTURE WORKS

The MVWT-II vehicle faced and overcame many design challenges. These included the requirements of low weight, relative low cost and simplicity. The small size of the Zaurus greatly contributed to the compact size of the vehicle. A larger computational platform would have required greater power and greater lift strength. The lightweight solution facilitated the advantageous downsizing of the hovercraft towards lower power consumption and a smaller lift fan.

There are some design challenges that the MVWT-II vehicle has not yet overcome, but its flexibility will allow for continual development. For greatest flexibility, the MVWT-II vehicle provides the DGPS or IR interface. This will allow use of the MVWT-II vehicle outdoors or in other testbeds. The rooftop testbed has not reached its full potential for at this time, for there are not yet hovercraft that can make full use of it. The flexibility of the MVWT-II vehicle will make it a functional testbed on which to test and develop new algorithms for distributed control. The flexibility in adding new sensors such as GPS module, Sonic ranger, etc, onto the Atmel interface board will support new developments on the MVWT-II vehicle for varied purposes. Also, the flexibility of the MVWT-II vehicle makes each vehicle a small platform on which to effectively test and develop new control algorithms for non-linear systems.

Currently, the local controller and low level software is under testing. Once the current fleet of vehicles has been proven experimentally, we will construct an additional group of 12 vehicles to enable cooperative control experiments, such as RoboFlag, using up to 24 total vehicles. Other future work includes implementing the Computation and Control Language (CCL) [6] on the Zaurus PDAs to complement work on using CCL for cooperative control currently in progress. We will also develop a system by which trajectories can be computed on a server using the Nonlinear Trajectory Generation (NTG) software developed at Caltech [7]. This server will communicate to the vehicles using the wireless network.

REFERENCES

[1] T. Chung, L. Cremean, W. B. Dunbar, Z. Jin, E. Klavins, D. Moore, A. Tiwari, D. van Gogh, and S. Waydo. A platform for cooperative and coordinated control of multiple vehicles: The Caltech Multi-Vehicle Wireless Testbed. In *Proc. of the 3rd Conference on Cooperative Control and Optimization*, Dec. 2002.

[2] L. Cremean, W. Dunbar, D. van Gogh, J. Hickey, E. Klavins, J. Meltzer, and R. M. Murray. The caltech multi-vehicle wireless testbed. In *Proc. of the 41st IEEE Conference on Decision and Control*, volume 1, pages 86–88, 2002.

[3] Raffaello D'Andrea and Richard M. Murray. The RoboFlag competitions. In *Proc. of the American Control Conference*, 2003.

[4] R. D'Andrea and R. M. Murray. The roboflag testbed. In *Proc. of the American Control Conference*, pages 656 – 660, 2003.

[5] J. P. Desai, J. P. Ostrowski, and V. Kumar. Modeling and control of formations of nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation*, 17(6):905–908, 2001.

[6] Eric Klavins. A language for modeling and programming cooperative control systems. In *Proceedings of the International Conference on Robotics and Automation*, 2004. To Appear.

[7] M.B. Milam. *Real-Time Optimal Trajectory Generation for Constrained Dynamical Systems*. Ph.d thesis, California Institute of Technology, 2003.