The Caltech Multi-Vehicle Wireless Testbed[1]

Lars Cremean♯, William B. Dunbar, Dave van Gogh, Jason Hickey,
Eric Klavins, Jason Meltzer and Richard M. Murray

Engineering and Applied Science   California Institute of Technology
♯Corresponding Author: lars@cds.caltech.edu

Abstract: *In this paper we introduce the Caltech Multi-Vehicle Wireless Testbed (MVWT), a platform for testing decentralized control methodologies for multiple vehicle coordination and formation stabilization. The testbed consists of eight mobile vehicles, an overhead vision system that provides GPS-like state information and wireless Ethernet for communications. Each vehicle rests on omni-directional casters and is powered by two high-performance ducted fans. Thus, a unique feature of our testbed is that the vehicles have second order dynamics, requiring real-time feedback algorithms to stabilize the system while performing cooperative tasks. The testbed will be used by various research groups at Caltech and elsewhere to validate theoretical advances in multi-vehicle coordination and control, networked control systems, real-time networking and high confidence distributed computation.*

1 Introduction

Controlling large scale, decentralized and networked systems is one of the most important challenges of control theory and practice today. Unmanned aerial vehicles, automated highway systems and automated manufacturing processes all involve multiple, interacting, highly dynamic entities. Elements of these systems are usually distributed in space and must coordinate with each other using sensing and communications networks. Inaccuracies in sensing and delays, interruptions and faults in communications as well as possible node failures are not amenable to standard control approaches, which often consider the problem of control as separate from communications and other issues resulting from decentralization. The large scale nature of these problems, furthermore, makes a global approach impractical, while a general methodology for ensuring global behavior from a collection of locally controlled elements remains elusive.

The Multi-Vehicle Wireless Testbed (MVWT) is a tool for validating theoretical advances in multiple-vehicle coordination and control, networked control systems, real-time networking and high confidence distributed computation (Figure 1). A unique feature of our testbed is that the vehicles (Figure 2) have second order dynamics. Other multi-vehicle testbeds such as those used in robotic soccer [5] feature "kinematic" vehicles that are able to stop almost instantly in the face of possible collisions or other emergencies. In contrast, in controlling the MVWT vehicles one must account for inertia. Because of this difference, the coordination algorithms we develop will rely on advanced control techniques instead of AI style planning. Recent progress in such techniques include the application of model predictive control (MPC) to groups of vehicles [2] and the Caltech ducted fan experiment [1], as well as graph theoretic connections made between stability of certain multi-vehicle dynamical systems and the structure of information flow between the components of such systems [3].

The inspiration behind the MVWT vehicle design is the goal of controlling unmanned aerial vehicles (UAVs)

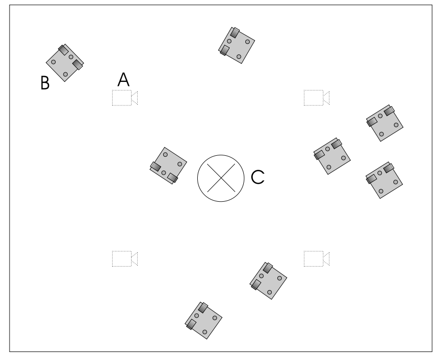

Figure 1: A diagram of the Caltech MVWT. Overhead cameras (A) provide state information to the vehicles (B). An obstacle (C), due to an unrelated experiment in the same space must be avoided.

Figure 2: An MVWT vehicle rests on three ball casters and supports a stripped down laptop, batteries and two ducted fans for propulsion.

[1]Partial support for this work was provided by AFOSR under grant F49620-01-1-0227.

singly and especially in groups [7, 8]. We believe we have captured the essence of what makes these systems difficult to control while keeping the complexity of the testbed low. A similar testbed, under construction at UIUC [10], uses hovercraft.

2 The Vehicle

Hardware. Each MVWT vehicle consists of a stripped-down laptop (with the screen removed) mounted on a Plexiglas frame with three low-friction, omni-directional casters. Over this are mounted two high-performance ducted fans each capable of producing up to 4.5 N of continuous thrust. Figure 2 depicts the vehicle complete with two NiMH battery packs and a fan interface board. Each vehicle is uniquely identified by markings on a “hat” placed on top of the vehicle. These markings are used by the “Lab Positioning System” (LPS) to determine the vehicle’s state and identity. Communication between the LPS, the command and control computers and the vehicles is handled by the laptops via 802.11b wireless cards. Control programs running on the laptop interface with the fans via an interface board containing a PIC microprocessor. The board connects the USB port of the laptop to the fan motor controllers, allowing the speed of the motors to be controlled from software. The latest design vehicle measures 25.4 cm deep, 35.6 cm wide and 18.1 cm high. With the casters, laptop, fans, batteries and interface board, the vehicle’s mass is approximately 5.15 kg.

Software. Each MVWT vehicle runs the QNX real-time operating system, which is easily tailored to new situations, making it ideal for embedded applications such as ours. Control code for the vehicles is written using the RHexLib C++ robot programming suite, originally developed to control the RHex hexapod robot [9, 4]. RHexLib facilitates a modular programming style wherein each basic function of the vehicle is encapsulated as a `Module`, an abstract base class defining a standard interface to the RHexLib *ModuleManager*, which handles initializing, activating and updating modules. Communication among vehicles and between vehicles and the command and control computer is accomplished using a customizable communication system (called libcomm) which resides over TCP/IP and 802.11b. For trajectory generation we use the Nonlinear Trajectory Generator (NTG) software [6] running in parallel (as a separate thread) with RHexLib.

3 The Arena

The vehicles maneuver in a 6.71 m × 7.32 m rectangular arena with a post in the middle of the floor (see Figure 1). The post is part of an unrelated experimental apparatus, although it does serve as a “natural” obstacle, making control and coordination more challenging. When this modestly sized space is occupied by eight fast moving MVWT vehicles, the need for coordinated real- time control is readily apparent. A command PC is located next to the arena and can be used for transmission of global information and commands to the vehicles in the arena.

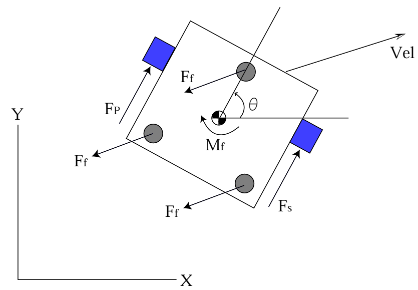

Figure 3: A schematic of an MVWT vehicle.

Mounted on the ceiling over the arena are four cameras which, together with a computer dedicated to vision processing, form the LPS, which determines the positions, orientations and velocities of the vehicles and broadcasts them over the local wireless network. The core of the LPS is a vision system which consists of four monochrome CCD cameras (Pulnix TM-6710) and four vision processing boards (Matrox Genesis and Genesis Plus). Each camera covers an area of approximately 4.88 x 4.26 m on the floor and produces 640x480-pixel images at 60Hz. The room was designed to facilitate processing by making the floor entirely white and the vehicles identified by black patches (blobs). On each vehicle hat, blobs and the holes in them are used to determine position, orientation and vehicle identity. Vehicles can be localized to within 1cm and processing a typical frame takes under 20ms.

4 Modeling and Control

A schematic of a vehicle with coordinate axes showing position $(x, y) \in X \times Y$ and orientation θ is shown in Figure 3. Modeling the frictional forces F_f and moment M_f as viscous, the equations of motion are:

$$\begin{aligned} m\ddot{x} &= -\eta\dot{x} + (F_s + F_p)\cos\theta \\ m\ddot{y} &= -\eta\dot{y} + (F_s + F_p)\sin\theta \\ J\ddot{\theta} &= -\psi\dot{\theta} + (F_s - F_p)r. \end{aligned} \tag{1}$$

The mass m of the vehicle is 5.15 kg and the measured rotational inertia J is 0.050 kg-m^2. The starboard and port fan forces are denoted F_s and F_p, respectively, and r (0.124 m) denotes the (common) moment arm of the forces where the center of geometry and mass of the vehicle are assumed to coincide. The coefficient of viscous friction, η, is 5.5 kg/s and the coefficient of rotational friction, ψ, is 0.084 kg m^2/s.

In the rest of this section we describe our first attempts at controlling the vehicles, thereby illuminating their unique characteristics.

4.1 LQR Tracking Control

An equilibrium point for the dynamics in equation (1) is any constant position and orientation with zero velocity. However, the linearized dynamics of equation (1) are not controllable around any equilibria. To achieve controllability, we can consider the error dynamics around a constant velocity $\dot{x}_{nom}$ and heading

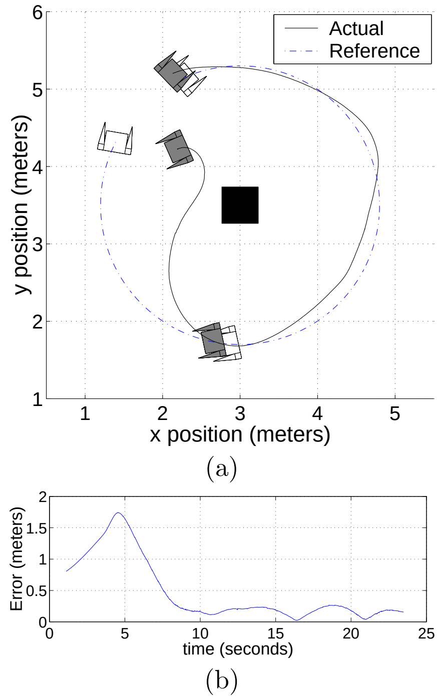

Figure 4: (a) Data from a vehicle tracking a circular reference trajectory from rest. (b) The error $||(x, y) - (x_r, y_r)||$ as a function of time for the data in (a).

θ_{nom} giving the reference state

$$(x_r(t_0) + t\dot{x}_{nom},\ y_r(t_0) + t\dot{y}_{nom},\ \theta_{nom},\ \dot{x}_{nom},\ \dot{y}_{nom},\ 0)^T,$$

where $\dot{y}_{nom} = \dot{x}_{nom}\tan(\theta_{nom})$. The nominal inputs are $F_s = F_p = (\eta\dot{x}_{nom})/(2\cos\theta_{nom})$. The error dynamics, denoting the states with a subscript e, become

$$\begin{aligned} m\ddot{x}_e &= -\eta(\dot{x}_e + \dot{x}_{nom}) + (F_s + F_p)\cos(\theta_e + \theta_{nom}) \\ m\ddot{y}_e &= -\eta(\dot{y}_e + \dot{y}_{nom}) + (F_s + F_p)\sin(\theta_e + \theta_{nom}) \\ J\ddot{\theta}_e &= -\psi\dot{\theta}_e + (F_s - F_p)r. \end{aligned} \quad (2)$$

The controllable equilibria of equation (2) are thus any constant $(x_e, y_e, \theta_e, \dot{x}_e, \dot{y}_e, \dot{\theta}_e) = (c_1, c_2, 0, 0, 0, 0)$ for $c_1, c_2 \in \mathbb{R}$. LQR controllers based on the linearization of equation (2) have been validated on the testbed. Also, the simulations in [2] stabilize a formation of vehicles with these error dynamics. We also consider the error dynamics around a circular reference with constant radius and angular velocity, a trajectory more suitable to the arena dimensions. Figure 4 shows a vehicle tracking a circular reference trajectory using an LQR controller.

4.2 Trajectory Generation

Optimization based approaches, such as model predictive control, are also being investigated for coordinated control of the testbed vehicles [2]. The Nonlinear Trajectory Generation (NTG) software package developed at Caltech [6] is currently being used to solve (on-line) optimal control problems for path planning of individual vehicles.

5 Conclusions and Future Work

Motivated by the need for an experimental platform to validate theoretical advances in decentralized control, we have developed a new testbed for studying multi-vehicle, networked control. The main feature of the testbed is the use of ducted fans to control the vehicles, resulting in input-constrained and underactuated second order dynamics. We described the Lab Positioning System and the wireless network, and preliminary controller design efforts. Many improvements are planned. We are also building a complete simulation environment which will allow us to use the same RHexLib code that we run on the actual vehicles on simulated vehicles. Finally several options are being explored, such as a Bluetooth network, to expand upon and provide alternatives to the wireless Ethernet communications we are presently using, thereby providing a rich environment for networking control research.

Acknowledgments: Many others have contributed to the MVWT project, including Jason Cardema, Zhipu Jin, Melvin Flores, Kelly Klima, Ben Lee, Jonathan Chauvin, Steven Low, Dave Moore, Henrik Sandberg, Laure Sinegre, Cristian Tapus, Abhishek Tiwari, Stephen Waydo and Sam Yeager.

References

[1] William B. Dunbar, Mark B. Milam, Ryan Franz, and Richard M. Murray. Model predictive control of a thrust-vectored flight control experiment. In *2002 IFAC World Congress*, Barcelona, Spain, 2002.

[2] William B. Dunbar and Richard M. Murray. Model predictive control of coordinated multi-vehicle formations. In *2002 Conference on Decision and Control*, Las Vegas, NV, 2002.

[3] J. Alexander Fax and Richard M. Murray. Graph laplacians and stabilization of vehicle formations. In *2002 IFAC World Congress*, Barcelona, Spain, 2002.

[4] E. Klavins and U. Saranli. Object orient state machines. *Embedded Systems Programming Magazine*, 2002. In Press.

[5] G. Kraetzschmar, P. Stone, and T. Balch, editors. *Robot Soccer WorldCup IV, Lecture Notes in Artificial Intelligence*. Springer, 2001.

[6] M. B. Milam, K. Mushambi, and R. M. Murray. A computational approach to real-time trajectory generation for constrained mechanical systems. In *Proceedings of the 2000 Conference on Decision and Control*, Sydney, Australia, 2000.

[7] Mark B. Milam, Ryan Franz, and Richard M. Murray. Real-time constrained trajectory generation applied to a flight control experiment. In *2002 IFAC World Congress*, Barcelona, Spain, 2002.

[8] A. Richards, J. Bellingham, M. Tillerson, and J. P. How. Co-ordination and control of multiple UAVs. In *AIAA Guidance, Navigation, and Control Conference*, Monterey, CA, 2002.

[9] U. Saranli, M. Buehler, and D. E. Koditschek. RHex: A simple and highly mobile hexapod robot. *The International Journal of Robotics Research*, 20(7):616–631, July 2001.

[10] A. Stubbs and G. Dullerud. Networked control of distributed systems: A testbed. In *Proceedings of IMECE 200, 2001 ASME International Mechanical Engineering Congress and Exposition*, New York, Nov 2001.