Towards an Indoor Testbed for Mobile Networked Control Systems

Martin Larsson Jonas Lindberg Jens Lycke Karl Hansson
Aziz Khakulov Emil Ringh Fredrik Svensson Isak Tjernberg
Assad Alam Jose Araujo Farhad Farokhi Euhanna Gadhimi
Andre Teixeira Dimos V. Dimarogonas Karl H. Johansson

KTH Royal Institute of Technology, 100-44 Stockholm, Sweden.

Abstract

In this paper, we consider the design of an indoor testbed composed of multiple aerial and ground unmanned vehicles for experimentation in Mobile Networked Control Systems. Taking several motivational aspects from both research and education into account, we propose an architecture to cope with the scale and mobility aspects of the overall system. Currently, the testbed is composed of several low-cost ARdrones quadrotors, small-scale heavy duty vehicles, wireless sensor nodes and a vision-based localization system. As an example, the automatic control of an ARdrone is shown.

keywords: Quadrotors, ARdrones, Heavy-duty vehicles (HVDs), Vision-based localization, Wireless Sensor Networks.

1 Introduction

Increasingly, control systems are operated over large-scale networked infrastructures. The control of multiple Unmanned Aerial Vehicles (UAVs) and Intelligent Ground Vehicular Systems (IGVs) are an example of that. Applications for these systems range from environmental monitoring [1, 2], surveillance and reconnaissance for security [3], to roadway traffic monitoring [4, 5] in the case of UAV systems. Several From an IGV perspective, a higher interest has been paid to the coordinated control of vehicles in roads in order to improving safety of drivers, increasing road capacities and ultimately save vehicle fuel and reduce gas emissions. These features for the case of heavy-duty vehicles has been particularly targeted by many researchers [6, 7], where they propose vehicle platooning strategies to solve these issues. Several examples of IGV systems were evaluated in the Grand Cooperative Driving Challenge (GCDC) challenge in 2011 [8].

In all these applications, vehicles must rely on local sensing, computation and control and take benefit of available communication links with other vehicles. Moreover, when taking into account the possible distances between vehicles as well as their number, it may be infeasible to rely on solely centralized computations to achieve mission goals and so, decentralized methods should be applied. This becomes a problem of networked estimation and control, where the vehicle dynamics as well as network dynamics should be taken into account at all times.

These research applications serve as motivation for our experimental testbed. Testbeds have been developed within robotics, with ground vehicles [9], UAVs [10, 11] as well as the combination of ground robots and wireless sensor networks [12]. The testbed proposed in this paper has the objective to join both ground and aerial vehicles as well as wireless sensor networks. We use Commercial-Of-The-Shelf (COTS) small-scale heavy-duty vehicles (HDVs) by Tamyia [13] and the ARdrones quadrotors by Parrot [14] and several Telosb wireless nodes [15]. The localization of each vehicle is performed by a camera system composed of Logitech Pro 900 Webcameras [16].

We will begin by discussing the design requirements for the system based on the driving applications and research performed in our lab as well as practical considerations. Ultimately we would like to have a robust, flexible and scalable testbed with a reach environment of dynamical systems, sensors, actuators as well as networking components. From this point of view, design considerations must be taken for the choice of the hardware and software components. We start by presenting the overall system architecture and proceed to detailed explanation of the main components of the testbed: the vehicles, software and the proposed infrastructure. Finally we present a single experiment with the ARdrones.

2 Motivating Design Considerations

The design of the experimental testbed proposed in this paper was motivated by the interest on the control of multi-agent systems with applications to area surveillance, roadway traffic monitoring and heavy-duty vehicle platooning. These applications lead to fundamental research on networked control and estimation over large-scale dynamical systems with resource limitations, and the impact of mobility in communication networks.

Examples of some relevant research topics on these fields are:

- Distributed and decentralized cooperative control and estimation
- Control and estimation with network constraints (data loss, delays and bandwidth limitation)
- Vehicular routing algorithms and Roadway traffic modelling for network connectivity analysis
- Wireless Sensor Networks

We refer the reader to [17] for current research challenges in networked control systems, with special emphasis on multi-agent systems. These are some of the topics that we expect to evaluate.

From the educational point of view, the Automatic Control Lab at KTH currently teaches undergraduate courses on Modelling of Dynamical Systems, Hybrid and Embedded Control Systems, Nonlinear

Control and a Project Course on Automatic Control. In all of them, there exist a laboratorial component in which requires the students to perform experimentation on real dynamical systems. At the moment, all the dynamical systems are static. It is our goal to integrate the proposed testbed to allow the students to understand the issues of mobility when performing control and estimation of dynamical systems.

3 System Requirements

Following the motivational aspects presented we can define the following requirements for the testbed:

- Robust and Reliable
- Easy to use and mantain
- Modular so that each component can be used separately from the rest if required
- Scalable with ease integration of heterogeneous types of sensors, actuators and vehicles
- Data logging and real-time measurements capabilities
- Inexpensive

In order for the testbed to be used by researchers, undergraduate and master students, it must provide a good degree of robustness, reliability as well as ease of usability. One important aspect is that all the selected components are also used by other people so current development efforts could be supported by manuals, online forums, etc. Moreover, all the software and hardware should be made in a way that it is easy for any person to operate it in a manner of days. From a user point of view, we must be able to provide real-time measurements as well as data logging capabilities so the developed algorithms could be evaluated. In our case, various control and estimation algorithms are to be developed and so, all the sensing and actuation values should be available at all times. Since all the communication between components is done over a network, we must be able to have an idea of the delay introduced, and network connectivity at all times. From a point of view of development, we should be able to implement algorithms both in each vehicle as well as in a centralized *mission controller*, so decentralized control could be implemented in a realistic way as well as in an emulated way. It is our goal to utilize as much COTS equipment as possible and do not rely in highly advanced and costly systems. In this way we are able to keep the development costs low as well as increase the number of systems components and scale up in a faster/easier way.

4 Components for multirobot experiments

Our objective is to select suitable vehicles to target the motivational aspects in Sec. 2. For that, we must choose ground and aerial vehicles with rich sensing, computational, actuation and communication

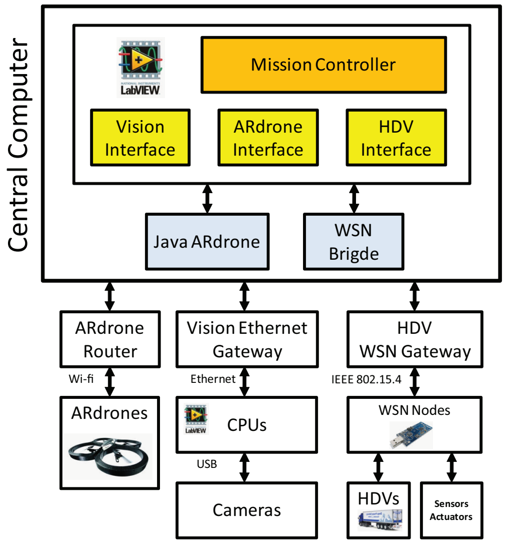


Figure 1: System architecture for the testbed.

capabilities. Cost and ease of support and maintenance from the robotics communities should be also considered. With the introduction of low-cost wireless sensors such as the IEEE 802.15.4 radio-enabled Telosb node [15] or the Arduino platform [18], it is now easy to integrate sensing, actuation, computation and communication under a single platform. In the case of the Telosb node, its computational and memory capabilities are not high, but the flexibility to customize the communication layers in the device are very appealing. In this way we are able to implement our own networking protocols or use the ones available from the wireless sensor network community using for e.g. TinyOS [19]. Moreover, there has been an increase interest from companies to provide sensor and actuation boards with a plug and play feature that allows for the fast development of many applications [20].

Nowadays, we are able to buy several low-cost robotics kits for both ground and aerial experimentation, plug and play sensors and actuators, as well as wireless sensors and integrate everything together.

In order to provide measurements of a global position of each vehicle we must have a robust localization system. Current commercially available systems to provide position measurements of robots are generally highly accurate but very expensive. Due to this fact, we develop a custom localization and tracking solution.

With respect to the software utilized in the testbed, we made the decision on relying on software that would be commonly known by the students and researchers involved in the university and not integrate any robotics-specific OS. Moreover, the software utilized should also be able to provide a simulation environment for the experimental setups proposed.

5 Experimental Testbed Components

The testbed consists of many components that are interfaced together as is depicted in Fig. 1. In what follows, we present the vehicles, software and infrastructure of the testbed.

5.1 Mobile Unmanned Vehicles

In the current tested we have deployed two types of vehicles for ground and aerial experimentation. For ground experiments we chose a small scale version of a heavy-duty vehicle to perform realistic platooning missions and for air we selected a commercial quadrotor which is a very flexibility platform with various onboard sensors.

5.1.1 Small Scale Heavy-Duty Vehicles

Since we would like to stay as close as possible to the realistic HDVs used for our research, we selected the Tamyia Scania HDV model as the ground vehicle [13]. Fig. 2(a) shows the Scania HDV. The HDV with tractor and trailer has a cost of around 500 Euro and can be easily adapted to allow for being controlled using Telosb nodes instead of the common RC controller. Moreover, it has the flexibility to carry different sensors as e.g. Infrared sensors to detect the distance to the HDV in front of it performing a platooning mission. Fig. 2(b) depicts the back of the HDV together with Telosb nodes. These wireless nodes communicate with the central computer through a WSN Gateway and perform the HDV actuation for turning and speed. The central computer is then responsible for computing the actuation inputs for each HDV in the network. The actuation signals are sent to the HDV with a period defined by the user. By using wireless sensor nodes we are able to have a good control of the delays and packet losses in the network by designing a suitable Medium Access Control (MAC) mechanism. For our current setup we are using the IEEE 802.15.4 standard MAC protocol [21].

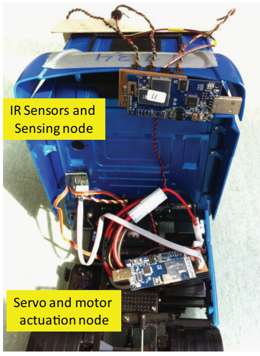
Payload weight may be increased or decreased in order to have an impact on the vehicle dynamics in a fairly realistic way. It also provides a geared motor with 3 different gear ratios, suspension with metal springs and friction dampers.

The HDV is separated in the tractor and trailer. The tractor has a total length of 520 mm and the trailer 930 mm. When connected the total length of the HDV is approximately 1150 mm.

In order to avoid errors when performing detection of the distance with the infrared sensors, we designed the system with four infrared sensors. The system is composed by two short range (4 cm to 30 cm) and two long range (20 cm to 150 cm) sensors in each corner from Phidgets (3520 and 3522 [20]). In this way, even if the HDV in front is not going completely straight when performing longitudinal platooning, one of the sensors should give a good distance measurement. The sensors are then connected to the wireless node which transmits the distance measurement in an event-based or periodic fashion, as defined by the user.

The testbed has five HDV at the moment and we are able to implement control algorithms for trajectory and speed tracking for single and multiple HDVs. With the acquired measurements for actuation and sensing we generate dynamical models of the HDV and implement model-based estimation and control algorithms.

(a) ARdrone Quadrotor and Heavy-Duty Vehicle



(b) Heavy-Duty Vehicle with Telosb nodes

Figure 2: Image shows, 2(a) the ARdrones [14], the HDV including trailer [13], and 2(b) with the changes made in the HDV to be controlled using Telosb nodes [15] and interface the Infrared sensors to measure distances to truck ahead.

5.1.2 Quadrotors

The quadrotors selected for the testbed are the ARdrones from Parrot [14] which are commercially available for around 300 Euro. These quadrotors are running an embedded Linux OS on a 468mhz ARM processor and are able to communicate with a base station computer or smartphone over Wifi. As sensors, they have two cameras, one pointing the front and another to the bottom. Additionally, the ARdrone has a 6DOF, MEMS-based inertial measurement unit IMU, which provides pitch, roll and yaw measurements as well as an ultrasound for reporting the ARdrone height. These measurements are used for the built-in stabilizing controller of pitch, yaw and roll. Supervisory control can be performed using a base station or smartphone. The control commands are reference values for the control of roll, pitch and yaw angles as well as lift throttle. The API was developed in Java which provides an interface to the onboard sensors and actuate on the ARdrone. This API is connected to LabVIEW [22] where the mission controller is implemented. Our future goal is to control the ARdrone using Telosb nodes instead of the wi-fi channel.

Several control objectives can be realized with this setup. At the moment we have four ARdrones in the testbed and we are currently acquiring Arducopters [23] as other air vehicles. The benefit of the Arducopter is that the onboard sensing, actuation and stabilizing controller can be designed by the user which is not allowed in the ARdrone.

5.2 Software

The main software used in the testbed is LabVIEW since it is a very intuitive and powerful programming tool for performing real experiments, and it is currently used in many of the courses offered at KTH. More specifically, we use the IMAQ Dx toolkit for image capture and processing and Control Design and PID and Fuzzy Logic toolkit for control and estimation. In LabVIEW we are also able to simulate our closed-loop control of the HDV and ARDrone based on their dynamical systems. An example of

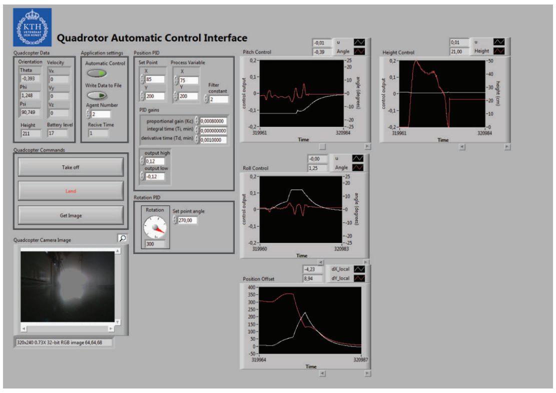


Figure 3: ARdrone control interface in LabVIEW. In the interface you are able to manually control the ARdrone, take off and land it, as well as analyzing all the sensing measurements transmitted.

the LabVIEW interface for the ARdrone control can be seen in Fig. 3. Moreover, we develop a Java application to perform the interface between the ARdrones and LabVIEW. The wireless nodes are all running the TinyOS operating system, with the IEEE 802.15.4 standard MAC.

5.3 Localization infrastructure and design

The testbed is currently implemented in a 50 m long and 2.5 m wide corridor in the 8th floor at the EE School building at KTH. A total of five HDV and four ARdrones as well as around 200 wireless nodes Telosb are available for deployments. For vehicle localization we are using seven Logitech 900 Pro webcameras [?] that cover the whole corridor.

At the current deployment stage, the seven cameras are connected to three separate CPUs running LabVIEW and communicating with the central computer over Ethernet, as depicted in Fig. 1. Each camera overlaps the range of the following camera so each vehicle moving between cameras can be correctly localized by at least one of the cameras.

For localization purposes, each vehicle has a given pattern and color. In this way we are able to quickly localize 4 (due to the color limitations) vehicles for each given pattern. The pattern has a diamond shape which allows us to clearly identify the vehicles orientation. Currently we are developing a localization algorithm with LED-based markers using geometric combinations of n regions in a m by m matrix marker. In this marker, $m^2 - 4$ regions are specially defined for showing the robot orientation. In this way we are able to localize up to $(m^2 - 4)^2$ vehicles. We have to remark that localizing more vehicles will increase computational time.

When performing localization of the ARdrones in 3D with the camera system, we take advantage of the correct measurement of the height of the ARdrone given by the onboard ultrasound sensors. For both HDV and ARdrone localization the error is always below 5 cm.

The resolution of each acquired image is 640x400 and we can perform the search of all the robots in each of the cameras with period of 50 ms. The image processing algorithm running in LabVIEW takes the following steps:

1. Take picture with all cameras
2. Do Background Subtraction (BGSub) and update all backgrounds for all cameras
3. For the cameras that had pictures where BGSub got detection go to step 5, if not go to 3
4. Look for one agent at the time in all the pictures
5. For every detection we register the values
6. Do a filter to sort out what detection to use (get rid of false ones)
7. Repeat 4-6 for every agent type and color

We remark that no optimizations are being done at the moment and the vision system is still open for modifications.

6 Experiments

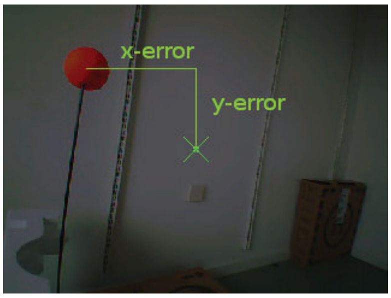
Due to space limitations we will only show a simple test with the ARdrone. This experimental test was to make the ARdrone following an object. This object could for example be another UAV, but for simplicity a red ball was used. The ARdrone faced the ball at all times while trying to keep a fixed distance of one meter from the ball to itself. From the image stream of the camera, LabVIEW [22] was used to calculate the centroid of the red ball. The control algorithm then tried to keep the center of the ball in the center of the camera image. This was done by yawing as a function of the error in x and increasing lift speed as a function of the y-error (see figure 4(a)). A Proportional Derivative (PD) controller was used since the integrating part was removed since no steady state errors were encountered.

LabVIEW also calculated the area of the ball in the image and then used that to estimate the distance of the ball to the camera. The viewed area A of an object at distance d decreases as the inverse square of d. That is $A \propto d^{-2}$, which gives $d = 1/\sqrt{kA}$, where k is a constant. Due to the oscillatory behavior of the ARdrone, a low pass filter and an outlier filter had to be implemented.

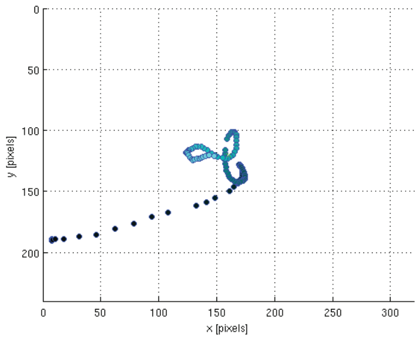
As seen in figure 4(b) the UAV was able to keep the ball in the center of its image. Also, it was able to stay approximately one meter away from the ball, however, due to the relative high noise the distance controller did not work as well as the one keeping the ball in the center of the image.

7 Conclusion

In this paper we propose an indoor testbed composed of unmanned aerial and ground vehicles. The testbed is to be used for research in networked control systems as well as for educational purposes. Even though the architecture and main components are defined, the testbed is still being developed and several algorithms are being evaluated on it. These algorithms range from platooning of HVDs, to coordinated control of the ARdrones to track HVDs. In the near future we plan to optimize the vision



(a) Camera view and error when tracking a red ball



(b) Control of ARDrone to track a red ball

Figure 4: Showing how the controller tries to keep the ball centered in the camera view in Fig. 4(a). In Fig. 4(b) the blue dots show the calculated centroid pixel of the red ball. Light blue dots is later in time then darker blue ones. As it can be seen, the UAV is able to keep the ball in the center of the picture.

system and finish the implementation of the new LED-based localization system. Moreover, we plan to integrate Arducopters and other small low-cost holonomic and non-holonomic ground robots.

REFERENCES

[1] G. M. Goncalves, E. Pereira, J. B. de Sousa, J. Morgado, R. Bencatel, J. Correia, and L. Felix, "Unmanned air vehicles for coastal and environmental research," in *10th International Coastal Symposium - ICS 2009, Lisboa, Portugal*, 2009.

[2] P. B. Sujit, J. B. de Sousa, and F. L. Pereira, "Coordination strategies between uav and auvs for ocean exploration," in *European Control Conference*, 2009.

[3] V. Kumar, D. Rus, and S. Singh, "Robot and sensor networks for first responders," *IEEE Pervasive Computing*, vol. 3, pp. 24–33, October 2004.

[4] M. Boban, T. T. V. Vinhoza, J. Barros, M. Ferreira, and O. K. Tonguz, "Impact of Vehicles as Obstacles in Vehicular Ad Hoc Networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 1, pp. 15–28, January 2011.

[5] K. Ro and J.-S. Oh, "Lessons learned: Application of small uav for urban highway traffic monitoring," in *45th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, USA, 2007.

[6] C. Bonnet and H. Fritz, "Fuel consumption reduction in a platoon: Experimental results with two electronically coupled trucks at close spacing," *SAE paper 2000 - 01 - 3056*, 2000.

[7] A. Alam, A. Gattami, and K. Johansson, "An experimental study on the fuel reduction potential of heavy duty vehicle platooning," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, sept. 2010, pp. 306 –311.

[8] *Grand Cooperative Driving Challenge (GCDC)*, 2011. [Online]. Available: http://www.gcdc.net/

[9] N. Michael, J. Fink, and V. Kumar, "Experimental testbed for large multirobot teams," *Robotics Automation Magazine, IEEE*, vol. 15, no. 1, pp. 53 –61, march 2008.

[10] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The grasp multiple micro-uav testbed," *Robotics Automation Magazine, IEEE*, vol. 17, no. 3, pp. 56 –65, sept. 2010.

[11] S. Bouabdallah, P. Murrieri, and R. Siegwart, "Design and control of an indoor micro quadrotor," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 5, Apr. 2004, pp. 4393 – 4398 Vol.5.

[12] J. R. Martnez-de Dios, A. Jimnez, and A. Ollero, "Towards an integrating validation and simulation tool for cooperating objects," in *Proceedings - 2nd International Workshop on Networks of Cooperating Objects CONET2010. Stockholm, Sweden*, 2010.

[13] *Tamyia Scania R620 6X4 Highline Full Operation Kit.* [Online]. Available: http://www.tamiya.com/english/products/56322scania/index.htm

[14] Ardrone quadrotor. [Online]. Available: http://ardrone.parrot.com/

[15] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," *Information Processing in Sensor Networks 2005.*, Apr. 2005.

[16] Logitech 9000 pro webcam. [Online]. Available: http://www.logitech.com/sv-se/webcam-communications/webcams/devices/5867

[17] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*, ser. Applied Mathematics Series. Princeton University Press, 2009.

[18] *Arduino*, The Arduino Platform. [Online]. Available: http://www.arduino.cc/

[19] D. Gay, P. Levis, and D. Culler, "Software design patterns for tinyos," *SIGPLAN Not.*, vol. 40, no. 7, pp. 40–49, 2005.

[20] *Products for USB Sensing and Control*, Phidgets. [Online]. Available: http://www.phidgets.com/

[21] *IEEE 802.15.4 standard: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)*, IEEE, 2006. [Online]. Available: http://www.ieee802.org/15/pub/TG4.html

[22] *LabVIEW*, National Instruments. [Online]. Available: http://www.ni.com/labview/

[23] Arducopter. [Online]. Available: http://code.google.com/p/arducopter/