Proceedings of the 2005 IEEE
International Conference on Robotics and Automation
Barcelona, Spain, April 2005

Air Vehicle Simulator: an Application for a Cable Array Robot

Kane Usher, Graeme Winstanley,
Peter Corke and Dirk Stauffacher
Robotics Team
CSIRO ICT Centre
P.O. Box 883, Kenmore, Australia 4069
firstname.lastname@csiro.au

Ryan Carnie
School of Electrical Engineering
Queensland University of Technology
Brisbane, Australia 4000
firstname.lastname@csiro.au

***Abstract*— The development of autonomous air vehicles can be an expensive research pursuit. To alleviate some of the financial burden of this process, we have constructed a system consisting of four winches each attached to a central pod (the simulated air vehicle) via cables — a cable-array robot. The system is capable of precisely controlling the three dimensional position of the pod allowing effective testing of sensing and control strategies before experimentation on a free-flying vehicle. In this paper, we present a brief overview of the system and provide a practical control strategy for such a system.**

***Index Terms*— Cable-array robot, control, autonomous air vehicles.**

I. Introduction

Over the past several years, the CSIRO Robotics Team has expended considerable energy into the development of a low-cost autonomous air vehicle in the form of a 60 size RC helicopter, see e.g. [1]. As with much of the research in field robotics, a large proportion of this work has related to solving engineering problems which, to some extent, has been at the expense of conducting research. Although most of the engineering problems have been overcome, including for example vibration isolation for sensing pods (see e.g. [2]) and the design of a low-cost inertial sensing unit, experimenting with air vehicles still remains problematic. Hurdles include the requirement for a skilled pilot to catch the aircraft in the event of a failure and the inevitable repair expenses in the case of such failures.

As a means of reducing development time and increasing research productivity, this paper describes a cable-array robot for air vehicle simulation. Cable-array robots, also known as cable-driven or cable-suspended robots, are defined as those robots which control an end-effector using multiple actuated cables. An example of such a system is shown in Fig. 1.

Cable-array robots possess several advantages over traditional serial or parallel robot mechanisms. Firstly, cable-array robots can operate over much larger workspaces and provide higher performance in terms of relative stiffness and speed. They have fewer moving parts and can handle large loads relative to total robot weight. Disadvantages of cable-array robots include possible cable interference, possible inaccuracies at the end-effector due to cable stretch, limited force application in the downward direction, and the requirement for an overhead space.

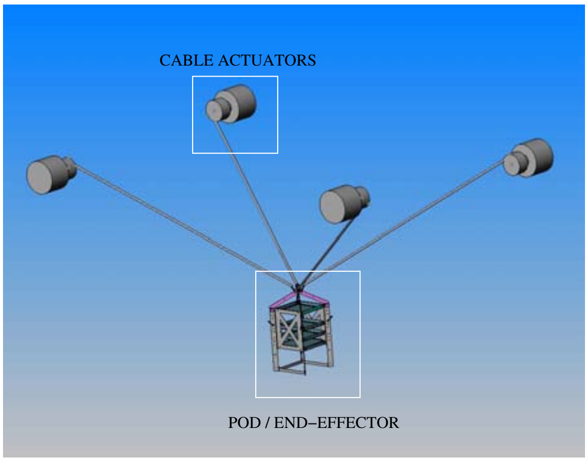


Fig. 1. An example of a cable-array robot. By changing the length of the cables, the central pod's position can be controlled. Diagram adapted from [3].

The application scope of cable-array robots includes: building and maintenance of large constructions; cranes; materials processing and handling; clean-up of disaster sites; monitoring and inspection of built environments; and humanitarian de-mining. However, the purpose of the robot described in this paper is to investigate vision-based control algorithms for aerial vehicles including:

- static/moving target tracking
- autonomous landing
- pose stabilization
- insect-based navigation strategies
- terrain following and
- collision avoidance

A. Literature Review

To constrain an end effector with n degrees of freedom, cable-array robots require $n+1$ cables. In most cases, gravity can be considered as acting as a 'virtual cable', providing control over additional degrees of freedom. There are then two classes of cable-array robots, constrained and under-constrained systems, depending on the number of degrees of freedom controlled. As fully constrained systems require more cables, the available workspace is reduced due to limited force application at specific positions in the workspace and the increased

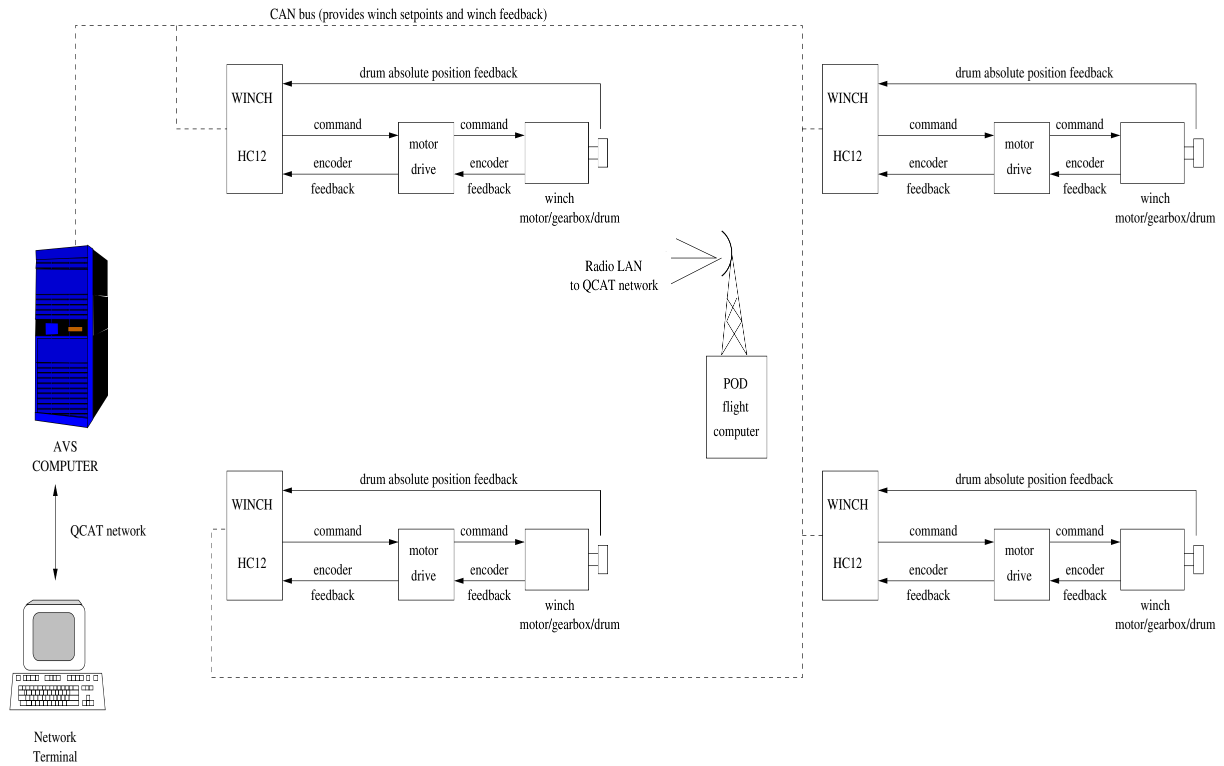


Fig. 2. The architecture of the AVS system.

risk of cable interference. Increased computation and mechanical complexity can also limit the scope of such systems. An example of a fully constrained system is the NIST RoboCrane [4]. In this system, an object (end-effector) is suspended by six cables which, with the addition of the gravity vector, constrains the six degrees of freedom of the object. Other systems include the FALCON system of [5] and the WARP system [6] both of which use seven or more cables to completely constrain the degrees of freedom for the end-effector.

Under-constrained systems are more popular in the literature due to their relative simplicity and larger workspace availability — this is of course at the expense of degrees of freedom. A striking example of an under-constrained cable-array robot is the SkyCam system [7] used in many sporting arenas around the world. SkyCam consists of a central pod housing servo electronics and a camera. The pod is driven around the stadium with a set of four computer controlled winches. Control of further degrees of freedom in the system has been added by providing on-pod processing and servoing for controlling the camera's roll, pitch and yaw, in addition to the position control provided by the cable-array system.

Besides the SkyCam system, most under-constrained cable-array robots have been restricted to simulations and small laboratory experiments. Ebert-Uphoff et al. [3], [8–10] have recently focused on stability measures and the force feasibility analysis of workspaces accessible by cable-array robots. However their work is highly theoretical with no evidence of testing on a real system. Likewise, Havlíik [11] presented an under-constrained cable-array system for a construction application but to date this work has been limited to theory.

Gorman et al. have provided theoretical work in the analysis of the dynamics of cable-array robots [12] and strategies for optimally distributing the forces amongst the cables [13]. The force distribution algorithm is formulated as an optimisation problem — computational limitations may preclude real-time operation and results for this work have been limited to simulations. Later work by the same authors includes the application of a sliding mode controller to an under-constrained cable-array system [14]. Results on a relatively small scale, four wire cable-array system indicate good path tracking with a relatively straight forward sliding mode controller.

Yanai et al. have presented anti-sway control strategies for the simplest form of the cable-array robot, the overhead crane [15], [16]. Comparisons of manual control versus their dynamic compensation method showed a marked improvement in the end-effector trajectory but the complexity of scaling the algorithm to deal with more than one cable could prove to be a hurdle for real-time operation.

This paper describes the development of the CSIRO Air Vehicle Simulator (AVS) system and describes our control strategy for the system. Section II describes the system architecture and briefly outlines the winch design. Section III briefly presents the kinematics for the case of a four-cable robot, while Section IV descibes the system control. Section V briefly describes visual servoing experiment illustrating the potential usefulness of the system, and finally, Section VI presents some concluding remarks and outlines directions for future research.

II. System Design and Architecture

The AVS is a four wire cable-array system covering a workspace of approximately $12\,\mathrm{m}$ long by $8\,\mathrm{m}$ wide by $6.3\,\mathrm{m}$ high. The pod position can be controlled by three winches, the fourth winch is used to increase the available area of operation. This means that at any one time there is a redundant cable and a strategy is required to deal with cable slackness. This section describes the system architecture and design.

A. Architecture

The basic system architecture is shown in Fig. 2. Each of the four winches consists of a Baldor motor, gearbox and drum in combination with a Baldor motor-drive. The motor-drive is commanded and controlled by a HC12 micro-controller which in turn receives commands from a central controlling computer via CANbus. The controlling computer gathers all the feedback from the winches, and, using the system kinematics, estimates the pod's position. The controlling computer also issues commands to each of the winches, based upon the feedback and on the demands requested from either the pod or another user. The Dynamic Data Exchange (DDX) system [17] lies at the heart of the software architecture allowing the exchange of data between separately running processes.

Feedback of the system state includes the motor encoder, an absolute position obtained from a potentiometer coupled with the winch drum, and cable tension sensing. The HC12 and winch sets are configured to accept and execute demands on cable tension, position and velocity — this allows for testing of a variety of control strategies.

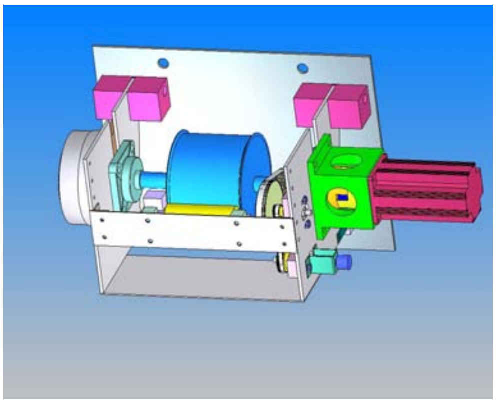
B. Winch design

The AVS system has been designed to accommodate a pod load of approximately $10\,\mathrm{kg}$ and to quickly accelerate to a nominated top speed of approximately $3\,\mathrm{ms}^{-1}$ in any coordinate direction. Brushless motors were found to provide the right combination of torque and speed for the application and are coupled with 10:1 planetary gear-sets to minimise system backlash. When coupled to the winch drum (which has a diameter of $0.15\,\mathrm{m}$), the system is capable of delivering line speeds of approximately $3.3\,\mathrm{ms}^{-1}$ and a peak line tension of approximately $1750\,\mathrm{N}$. Fig. 3(a) and Fig. 3(b) show conceptual and actual views of a winch unit. The motors are driven by sinusoidal drives (BALDOR MicroDrive) and are connected to standard $240\,\mathrm{V}$ power.

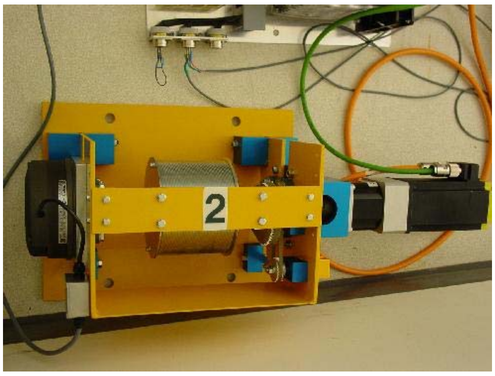
The drum is designed to accommodate the full length of required cable in one layer so as to minimise errors in cable length estimation introduced by the multi-layer case. In addition, a spring-loaded roller pushes against the drum to prevent the cable from jumping off the drum grooves in the event of loss of cable tension.

The winch assembly and drive units have been designed such that they mount directly onto I-beams and connect to the central pod via an overhead pulley for each winch. Fig. 3(c) shows a complete single winch assembly with its associated HC12 and drive unit as installed in the testing arena at the CSIRO QCAT site.

1) Cable tension management: During the design phase, the management of slack cables was highlighted as a critical issue for which a two-pronged strategy was devised. The first strategy was to manage cable tension via feedback control. The second was to ensure that



(a) Concept



(b) Actual

(c) As installed

Fig. 3. The winch assembly.

the cable would not drop of the winch drums by the introduction of a spring roller system pushing the cable against the drum.

Tension feedback control involved designing mechanisms for the measurement of cable tension. The motor/gearbox/drum sets have been mounted such that under cable tension, the mounting bracket rotates about a fixed point, if not for two tension bars preventing rotation. These tension bars are strain-gauged and provide an estimate of cable tension. Of course, motor acceleration influences these measurements but we have found that a simple median filter eliminates most of these effects.

In practice, managing cable tensions via feedback was found to be effective with winches in isolation. However, when all cables were connected to the pod, the winches tended to ‘fight’ each other. The second stage of the cable management strategy, that is the spring roller system, has in fact been found to be sufficient for preventing the cables from jumping off the drums and tension control has been discarded.

C. The Pod

The AVS pod is essentially a cage which houses the flight-computer, sensors, and batteries to power all on-board systems. The design is both light and strong and provides flexibility for carrying or mounting different sensors and components. The pod carries in its base configuration:

- miniITX computer
- Firewire cameras
- EiMU — a small CSIRO designed Inertial Measuring Unit [18]

with the facility to fit a variety of other sensors and components as desired. In terms of on-board power, batteries have been selected to allow testing for 1 to 2 hrs between battery charging or substitution. Refer to Fig. 1 for a schematic view of the pod.

The pod has multiple cable attachment points allowing for testing of different control strategies. For example, *is it better to treat the pod as a point mass or should further degrees of freedom be controlled?* Additional degrees of freedom could also be controlled by adding a pan/tilt or similar mechanism as with the SkyCam system mentioned in Section I-A. To date, treating the pod as a point mass moving in three dimensions has proved effective. Under this strategy, although the pod orientation remains relavely constant during motion, significant rolling and pitching can occur.

III. System Kinematics

The position of each winch pulley is known *a priori* and with the cable length estimates provided by appropriately scaling the motor encoder count, the problem of estimating the pod position in the work space is one of trilateration. The dimensions of the workspace are approximately (referring to Fig. 4):

$$\begin{aligned} a &= 12\,\mathrm{m} \\ b &= 8\,\mathrm{m} \\ h &= 6.3\,\mathrm{m} \end{aligned}$$

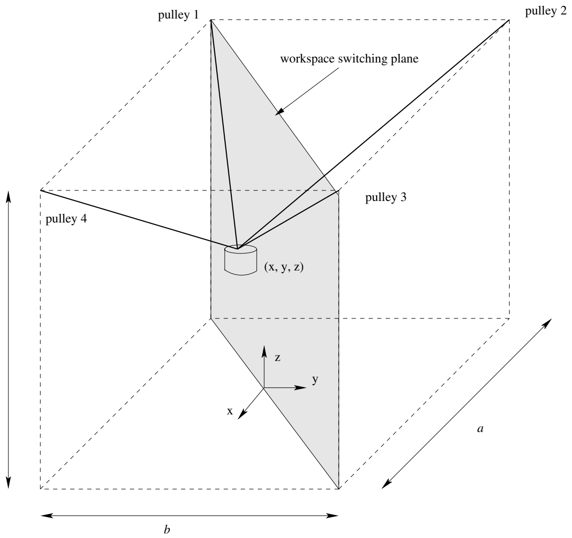


Fig. 4. The geometry of the AVS system. Winches have been omitted for clarity. Cable lengths are measured from the top of the pulleys. The switching plane used for control is also shown in the figure.

For the case of treating the pod as a point mass, the length of each cable can be described in terms of the position of the pod and the position of the associated pulley point [14]:

$$l_i^2 = (x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 \quad (1)$$

where (x, y, z) is the pod position and (x_i, y_i, z_i) is the position of the top of the ith pulley. Here, the origin of the coordinate system is defined with respect to the centre of the ground-plane of the workspace.

The position of each of the pulleys is given by:

$$\begin{aligned} (x_1, y_1, z_1) &= (\frac{-a}{2}, \frac{-b}{2}, h) \\ (x_2, y_2, z_3) &= (\frac{-a}{2}, \frac{b}{2}, h) \\ (x_3, y_3, z_3) &= (\frac{a}{2}, \frac{b}{2}, h) \\ (x_4, y_4, z_4) &= (\frac{a}{2}, \frac{-b}{2}, h) \end{aligned}$$

In terms of cable lengths, given a specified pod position (x, y, z), the required lengths are:

$$\begin{aligned} l_1 &= \sqrt{(x + \tfrac{a}{2})^2 + (y + \tfrac{b}{2})^2 + (z - h)^2} \\ l_2 &= \sqrt{(x + \tfrac{a}{2})^2 + (y - \tfrac{b}{2})^2 + (z - h)^2} \\ l_3 &= \sqrt{(x - \tfrac{a}{2})^2 + (y - \tfrac{b}{2})^2 + (z - h)^2} \\ l_4 &= \sqrt{(x - \tfrac{a}{2})^2 + (y + \tfrac{b}{2})^2 + (z - h)^2} \end{aligned} \quad (2)$$

These equations can be used to pre-calculate the required cable lengths for a given pod position. In effect, the cable lengths will form the *state vector* for the control system. Solving for the inverse kinematics requires a choice of which cables are active, as the system is over-constrained. Here we choose cables 1, 2, and 3 for which (x, y, z) are given by:

$$\begin{aligned} x &= \tfrac{1}{2a}\left(l_2^2 - l_3^2\right) \\ y &= \tfrac{1}{2b}\left(l_1^2 - l_2^2\right) \\ z &= h - \sqrt{l_1^2 - \left(x + \tfrac{a}{2}\right)^2 - \left(y + \tfrac{b}{2}\right)^2} \end{aligned} \quad (3)$$

Note that the equation for z is left in terms of x and y and the solution taken for z ensures that $z \leq h$. Differentiating these relations leads to the following expressions for cable velocity:

$$\begin{aligned} \dot{l_1} &= \frac{1}{l_1}\left[\dot{x}\left(x+\frac{a}{2}\right)+\dot{y}\left(y+\frac{b}{2}\right)+\dot{z}\left(z-h\right)\right] \\ \dot{l_2} &= \frac{1}{l_2}\left[\dot{x}\left(x+\frac{a}{2}\right)+\dot{y}\left(y-\frac{b}{2}\right)+\dot{z}\left(z-h\right)\right] \\ \dot{l_3} &= \frac{1}{l_3}\left[\dot{x}\left(x-\frac{a}{2}\right)+\dot{y}\left(y-\frac{b}{2}\right)+\dot{z}\left(z-h\right)\right] \\ \dot{l_4} &= \frac{1}{l_4}\left[\dot{x}\left(x-\frac{a}{2}\right)+\dot{y}\left(y+\frac{b}{2}\right)+\dot{z}\left(z-h\right)\right] \end{aligned} \quad (4)$$

IV. Control

As mentioned previously, assuming the pod acts as a point mass, at any one time there is a redundant cable. The strategy used here is to divide the workspace into two triangular prisms where the split is defined by the diagonal plane connecting pulleys 1 and 3, as illustrated in Fig. 4. Of course, the workspace could be divided into four triangular prisms, however, this introduces further switching points into the system which could in turn introduce chattering type problems.

In terms of pod motion control, there are two modes, *position* and *velocity* control. Each case results in a series of cable velocity demands which are then sent over CANbus to the individual winches. Position control is simply a further loop around the velocity control and thus velocity control is discussed first.

A. Velocity control

In velocity control, a user specifies Cartesian velocities which are then converted to cable velocity demands. Initially, the Jacobian type approach, as given in Equation 4, was used. However, it was found that this strategy lead to significant drift type problems in which the redundant cable would become too slack after some period of operation.

An alternative strategy has since been implemented in which cable velocities are calculated in terms of position/cable length errors. That is, given a specified Cartesian velocity, $(\dot{x^*}, \dot{y^*}, \dot{z^*})$, and knowledge of the pod's present position, an estimate of where the pod should be at the next time step is calculated. Cable lengths are calculated at the estimated future position, from which cable velocities can be calculated (given knowledge of the control loop cycle time and the present cable length). Mathematically, the pod's present cable lengths, l_i^t, are measured, from which the pod's Cartesian position, (x_t, y_t, z_t) can be calculated through Equation 3. Thus, given a set of Cartesian velocity demands,the pod's position can be estimated at time $t + \Delta t$:

$$x_{t+\Delta t} = x_t + \dot{x}^* \Delta t \quad (5)$$

$$y_{t+\Delta t} = y_t + \dot{y}^* \Delta t \quad (6)$$

$$z_{t+\Delta t} = z_t + \dot{z}^* \Delta t \quad (7)$$

Cable lengths are then calculated at the estimated position via Equation 2. Cable velocities can then be found:

$$\dot{l_i} = \frac{{l_i}^{t+\Delta t} - l_i^t}{\Delta t} \quad (8)$$

Pod motion resulting from these equations is extremely smooth and free of the drift problems associated with the Jacobian-based approach.

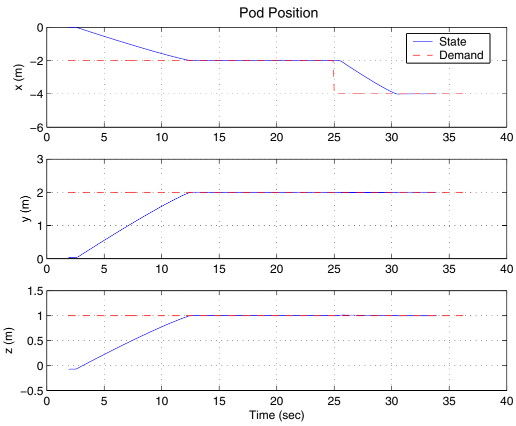


Fig. 5. An example of attaining a specified workspace position.

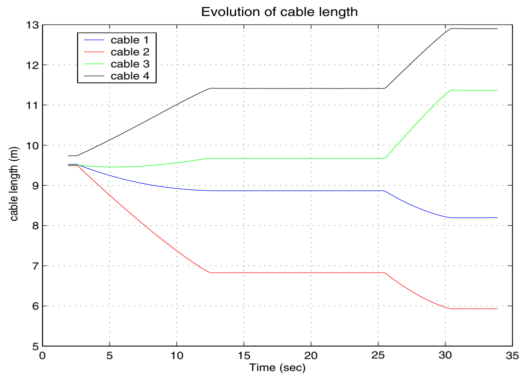


Fig. 6. The evolution of cable lengths payed out attaining the pod position of Fig. 5.

B. Position control

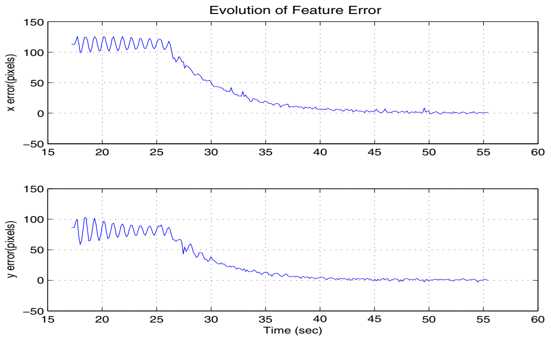
Position control allows for the attainment of a user specified desired pod position together with a maximum velocity. This is implemented as a loop around the velocity controller described above, in which a trapezoidal velocity profile is found from the initial and demanded positions.

Experiments prove the effectiveness of this rather simplistic control approach. Fig. 5 shows the trajectory of the pod given a demanded position, while the evolution of cable lengths is shown in Fig. 6. The first motion was specified to occur with a maximum velocity of $0.3\,\mathrm{m/s}$, while the second motion has a maximum velocity of $0.4\,\mathrm{m/s}$. The motion is smooth and relatively accurate although it must be remembered that position is calculated from cable length and hence will inherit any inaccuracies in cable length measurement.

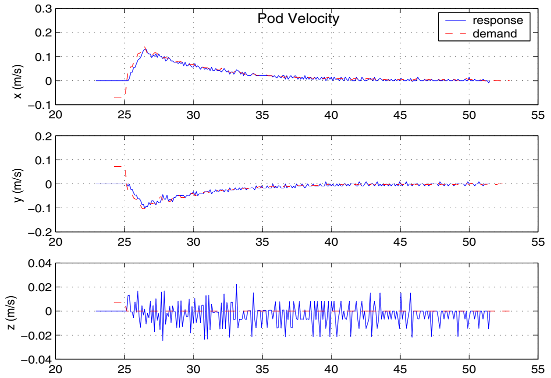
V. Example Experiment

To provide an example of the possible uses of the AVS system, a simple colour-based visual servoing experiment was performed. In this experiment, the imaging system on the AVS pod segmented a red witches hat using a pre-trained two dimensional lookup table of the object's colour (in YCrCb space). The segmented blob was then located with respect to the centre of the image, and the system servoed on this error using velocity demands.

Fig. 7(a) shows the evolution of feature error during the experiment while Fig. 7(a) shows the velocity de-



(a) Evolution of feature error.



(b) Velocity demands and responses.

Fig. 7. Results from the colour-based visual servoing experiment.

mands and response. This simple experiment illustrates the type of research we are aiming to conduct using the system, namely vision-based navigation and control. Although the system is restricted to three degrees of motion control, it is still very useful for developing and testing algorithms *before* or *in parallel* to the development occuring on an actual vehicle, whether it be a land, air or sea vehicle.

VI. Conclusion

The development of autonomous air vehicles can be an expensive and frustrating task due to reliability issues and the catastrophic cost of any failure. Cable-array robots provide an ideal testing platform for autonomous air vehicles as they allow an excellent range of motion and provide good position control. Testing can proceed without the risk of damage to expensive aircraft components or the need for a suitably qualified testing pilot.

This paper has presented the design and implementation of a four cable system which currently allows for control over three degrees of freedom. Further research will include the investigation of constraining further degrees of freedom in the system. However, the primary aim of the AVS is to ease the development of vision and control algorithms for implementation on field-based vehicles and this will be the focus of future research.

Acknowledgment

The authors would like to thank the rest of the CSIRO ICT Centre Robotics Team: Leslie Overs, Matthew Dunbabin, Stephen Brosnan, John Whitham, Pavan Sikka, Jonathan Roberts and Craig Worthington.

References

[1] G. Buskey, P. Corke, J. Roberts, P. Ridley, and G. Wyeth, “The CSIRO autonomous helicopter project,” in *Experiemntal Robotics VIII*, ser. STAR, B. Siciliano and P. Dario, Eds. Springer-Verlag, 2003, vol. 5.

[2] M. Dunbabin, S. Brosnan, J. Roberts, and P. Corke, “Vibration isolation for autonomous helicopter flight,” in *Proceedings of the International Conference on Robotics and Automation*, New Orleans, April 2004, pp. 3609–3615.

[3] P. Bosscher and I. Ebert-Uphoff, “A stability measure for underconstrained cable-driven robots,” in *Proceedings of the International Conference on Robotics and Automation*. New Orleans, LA, USA: IEEE, April 2004, pp. 4943–4949.

[4] J. Albus, R. Bostelman, and N. Dagalakis, “The NIST ROBOCRANE,” *Journal of Robotic Systems*, vol. 10, no. 5, pp. 709–724, 1993.

[5] S. Kawamura, W. Choe, S. Tanaka, and S. Pandian, “Development of an ultrahigh speed robot falcon using wire driven system,” in *Proceedings of the International Conference on Robotics and Automation*. Nagoya, Aichi, Japan: IEEE, 1995, pp. 215–220.

[6] K. Maeda, S. Tadokoro, T. Takamori, M. Hattori, M. Hiller, and R. Verhoeven, “On design of a redundant wire-driven parallel robot warp manipulator,” in *Proceedings of the International Conference on Robotics and Automation*. Detroit, Michigan, USA: IEEE, May 1999, pp. 895–900.

[7] CFInFlight, “What is skycam?” September 2004, http://www.cfinflight.com/pages/skycam.asp.

[8] P. Bosscher and I. Ebert-Uphoff, “Wrench-based analysis of cable-driven robots,” in *Proceedings of the International Conference on Robotics and Automation*. New Orleans, LA, USA: IEEE, April 2004, pp. 4950–4955.

[9] A. Reichel and I. Ebert-Uphoff, “Force-feasible workspace analysis for underconstrained point-mass cable robots,” in *Proceedings of the International Conference on Robotics and Automation*. New Orleans, LA, USA: IEEE, April 2004, pp. 4956–4962.

[10] A. Riechel, P. Bosscher, H. Lipkin, and I. Ebert-Uphoff, “Concept paper: Cable-driven robots for use in hazardous environments,” in *Proceedings of the 10th International Topical Meeting on Robotics and Remote Systems for Hazardous Environments*, Gainesville, FL, March 2004.

[11] Štefan Havlík, “Cable suspended manipulation robots,” in *Proceedings of the 16th International Symposium of Automation and Robotics in Construction*. IFAC/IEEE, 1999, pp. 269–274.

[12] W.-J. Shiang, D. Cannon, and J. Gorman, “Dynamic analysis of the cable array robotic crane,” in *Proceedings of the International Conference on Robotics and Automation*. Detroit, Michigan, USA: IEEE, May 1999, pp. 2495–2500.

[13] ——, “Optimal force distribution applied to a robotic crane with flexible cables,” in *Proceedings of the International Conference on Robotics and Automation*. San Francisco, California, USA: IEEE, April 2000, pp. 1948–1954.

[14] J. J. Gorman, K. W. Jablokow, and D. J. Cannon, “The cable array robot: Theory and experiment,” in *Proceedings of the International Conference on Robotics and Automation*. Seoul, Korea: IEEE, May 2001, pp. 2804–2810.

[15] N. Yanai, M. Yamamoto, and A. Mohri, “Inverse dynamics analysis and trajectory generation of incompletely restrained wire-suspended mechanisms,” in *Proceedings of the International Conference on Robotics and Automation*. Seoul, Korea: IEEE, May 2001, pp. 3489–3494.

[16] ——, “Anti-sway control for wire-suspended mechanism based on dynamics compensation,” in *Proceedings of the International Conference on Robotics and Automation*. Washington DC, USA: IEEE, May 2002, pp. 4287–4292.

[17] P. Corke, P. Sikka, J. Roberts, and E. Duff, “DDX: A distributed architecture for robot control,” in *Australian Conference on Robotics and Automation*, 2004.

[18] P. C. Jonathan Roberts and G. Buskey, “Low-cost flight control system for a small autonomous helicopter,” in *Proceedings of Australian Conference on Robotics and Automation (ACRA)*, Auckland, New Zealand, November 2002, pp. 71–76. [Online]. Available: ACRA02-heli.pdf