LINEAR, NONLINEAR AND CLASSICAL CONTROL OF A 1/5TH SCALE AUTOMATED EXCAVATOR

E. Sidiropoulou[*], E. M. Shaban[*], C. J. Taylor[*], W. Tych[†], A. Chotai[†]

[*] *Engineering Department, Lancaster University, Lancaster, UK, c.taylor@lancaster.ac.uk*
[†] *Environmental Science Department, Lancaster University, Lancaster, UK*

Keywords: Identification; model-based control; proportional-integral-plus control; state dependent parameter model.

Abstract

This paper investigates various control systems for a laboratory robot arm, representing a scale model of an autonomous excavator. The robot arm has been developed at Lancaster University for research and teaching in mechatronics. The paper considers the application of both classical and modern approaches, including: Proportional-Integral (PI) control tuned by conventional Ziegler-Nichols rules; linear Proportional-Integral-Plus (PIP) control, which can be interpreted as one logical extension of the conventional PI approach; and a novel nonlinear PIP design based on a quasi-linear model structure, in which the parameters vary as a function of the state variables. The paper considers the pragmatic balance required in this context, between design and implementational complexity and the potential for improved closed-loop performance.

1 Introduction

Construction is of prime economic significance to many industrial sectors. Intense competition, shortfall of skilled labour and technological advances are the forces behind rapid change in the construction industry and one motivation for automation [1]. Examples of excavation based operations include general earthmoving, digging and sheet-piling. On a smaller scale, trenching and footing formation require precisely controlled excavation. Full or partial automation can provide benefits such as reduced dependence on operator skill and a lower operator work load, both of which are likely to contribute to improvements in consistency and quality.

However, a persistent stumbling block for developers is the achievement of adequate fast movement under automatic control. Here, a key research problem is to obtain a computer controlled response time that improves on that of a skilled human operator. This presents the designer with a difficult challenge, which researchers are addressing using a wide range of approaches; see e.g. [2, 3, 4].

This paper considers a laboratory robot arm, a 1/5th scale representation of the more widely known Lancaster University Computerised Intelligent Excavator (LUCIE), which has been developed to dig trenches on a construction site [4, 5]. Despite its smaller size and light weight, the 1/5th model has similar kinematic and dynamic properties to LUCIE and so provides a valuable test bed for the development of new control strategies. In this regard, the paper considers both classical and modern approaches, including: Proportional-Integral (PI) control tuned by conventional Ziegler-Nichols rules; linear Proportional-Integral-Plus (PIP) control, which can be interpreted as one logical extension of the conventional PI approach [6, 7]; and a novel nonlinear PIP design based on a quasi-linear model structure in which the parameters vary as a function of the state variables [8].

Further to this research, the paper briefly considers utilisation of the robot arm as a tool for learning and teaching in mechatronics at Lancaster University. In fact, the laboratory demonstrator provides numerous learning opportunities and individual research projects, for both undergraduate and postgraduate students. Development of the complete trench digging system requires a thorough knowledge of a wide range of technologies, including sensors, actuators, computing hardware, electronics, hydraulics, mechanics and intelligent control.

Here, control system design requires a hierarchical approach with high-level rules for determining the appropriate end-effector trajectory, so as to dig a trench of specified dimensions. In practice, the controller should also include modules for safety and for handling obstructions in the soil [4]. Finally, the high-level algorithm is coupled with appropriate low-level control of each joint, which is the focus of the present paper.

2 Hardware

The robot arm has a similar arrangement to LUCIE [4, 5], except that this laboratory 1/5th scale model is attached to a workbench and the bucket digs in a sandpit. As illustrated in *Fig. 1*, the arm consists of four joints, including the boom, dipper, slew and bucket angles. Three of these are actuated by hydraulic cylinders, with just the slew joint based on a hydraulic rotary actuator with a reduction gearbox: see [9] for details.

The velocity of the joints is controlled by means of the applied voltage signal. Therefore, the whole rig has been supported by multiple I/O asynchronous real-time control systems, which allow for multitasking processes via modularisation of code written in Turbo C++ ®. The computer hardware is an AMD-K6/PR2-166 MHz personal computer with 96 MB RAM.

The joint angles are measured directly by mounting rotary potentiometers concentric with each joint pivot. The output signal from each potentiometer is transmitted with an earth line to minimise signal distortion due to ambient electrical noise.

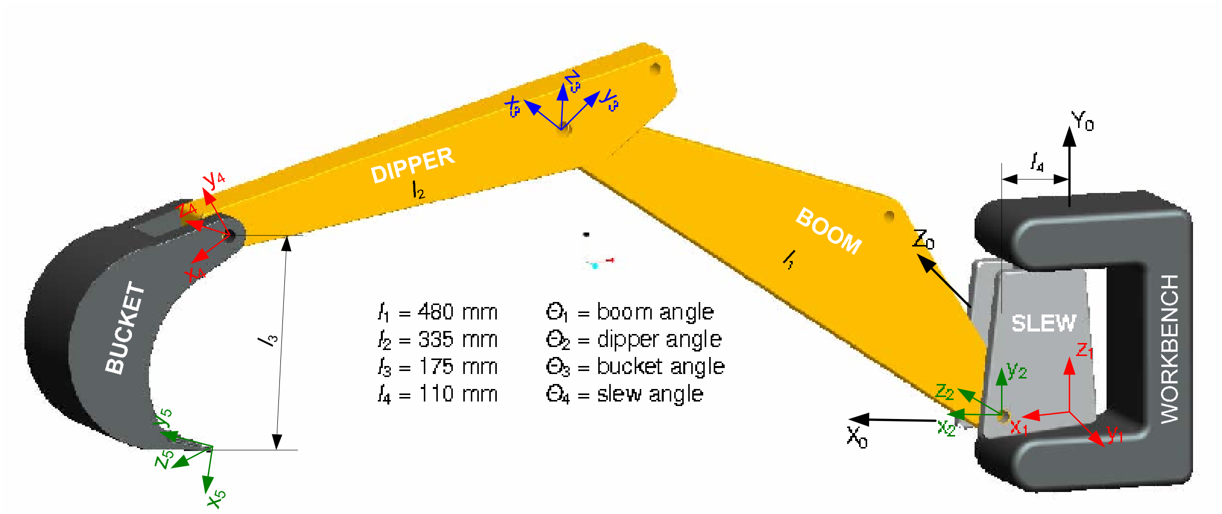

Figure 1: Schematic diagram of the laboratory excavator showing the four controlled joints.

These signals are routed to high linearity instrumentation amplifiers within the card rack for conditioning before forwarding to the A/D converter. Here, the range of the input signal just after conditioning does not exceed ± 5 volts. This A/D converter is a high performance 16 channel multiplexed successive approximation converter capable of 12 bit conversion in less than 25 micro seconds. At present only eight available channels are being used. In the future, therefore, there would be no problem for incorporating additional sensors into the system; e.g. a camera for detecting obstacles, to be used as part of the higher level control system, or force sensors.

Valve calibration is essential to provide the arm joints with meaningful input values. This calibration is based on normalizing the input voltage of each joint into input demands, which range from -1000 for the highest possible downward velocity to +1000 for the highest possible upward velocity of each joint. Here, an input demand of zero corresponds to no movement. Note that, without such valve calibration, the arm will gradually slack down because of the payload carried by each joint.

In open-loop mode, the arm is manually driven to dig the trench, with the operator using two analogue joysticks, each with two-degrees of freedom. The first joystick is used to drive the boom and slew joints while the other is used to move the dipper and bucket joints. In this manner, a skillful operator moves the four joints simultaneously to perform the task. By contrast, the objective here is to design a computer controlled system to automatically dig without human intervention.

3 Kinematics

The objective of the kinematic equations is to allow for control of both the position and orientation of the bucket in 3-dimensional space. In this case, the tool-tip can be programmed to follow the planned trajectory, whilst the bucket angle is separately adjusted to collect or release sand. In this regard, *Fig. 1* shows the laboratory excavator and its dimensions, i.e. θ_i (joint angles) and l_i (link lengths), where $i = 1, 2, 3, 4$ for the boom, dipper, bucket and slew respectively.

Kinematic analysis of any manipulator usually requires development of the homogeneous transformation matrix mapping the tool configuration of the arm. This is used to find the position, orientation, velocity and acceleration of the bucket with respect to the reference coordinate system, given the joint variable vectors [10]. Such analysis is typically based on the well-known Denavit-Hartenberg convention, which is mainly used for robot manipulators consisting of an open chain, in which each joint has one-degree of freedom, as is the case here [9].

3.1 Inverse kinematics

Given $\{X, Y, Z\}$ from the trajectory planning routine, i.e. the position of the end effector using a coordinate system originating at the workbench, together with the orientation of the bucket $\Theta = \theta_1 + \theta_2 + \theta_3$, the following inverse kinematic algorithm is derived by Shaban [9]. Here C_i and S_i denotes $\cos(i)$ and $\sin(i)$ respectively, whilst $C_{123} = \cos(\theta_1 + \theta_2 + \theta_3)$.

$$\bar{X} = \frac{X - l_4 C_4}{C_4} - l_3 C_{123} \tag{1}$$

$$\bar{Y} = Y - l_3 S_{123} \tag{2}$$

$$\theta_1 = \arctan\left[\frac{(l_1 + l_2 C_2)\,\bar{Y} - l_2 S_2 \bar{X}}{(l_1 + l_2 C_2)\,\bar{X} - l_2 S_2 \bar{Y}}\right] \tag{3}$$

$$\theta_2 = \pm\arccos\left[\frac{\bar{X}^2 + \bar{Y}^2 - l_1^2 - l_2^2}{2 l_1 l_2}\right] \tag{4}$$

$$\theta_3 = \Theta - \theta_1 - \theta_2 \tag{5}$$

$$\theta_4 = -\arctan\left[\frac{Z}{X}\right] \tag{6}$$

3.2 Trajectory planning

Excavation of a trench requires both ‘continuous path’ (CP) motion during the digging operation and a more primitive ‘point-to-point’ (PTP) motion when the bucket is moved out of the trench for discharging. In particular, each digging cycle can be divided into four distinct stages, as follows: positioning the bucket to penetrate the soil (PTP); the digging process in a horizontal straight line along the specified void length (CP);

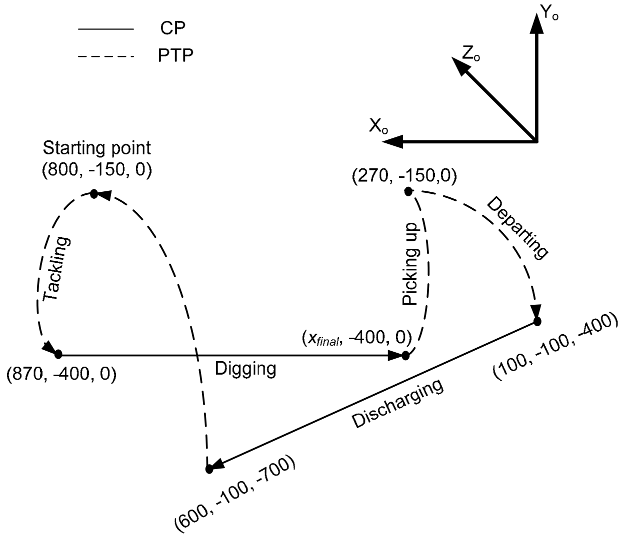

Figure 2: Trajectory planning for the laboratory excavator.

picking up the collected sand from the void to the discharge side (PTP); discharging the sand (CP).

For the present example, the CP trajectory can be traversed at a constant speed. Suppose $\overline{v}_0$ and $\overline{v}_f$ denote, respectively, the initial and final position vector for the end-effector and that the movement is required to be carried out in T seconds. In this case, the uniform straight-line trajectory for the tool-tip is,

$$\overline{v} = (1 - S_t)\overline{v}_0 + S_t\overline{v}_f \qquad 0 \leq t \leq T \tag{7}$$

Here, S_t is a differentiable speed distribution function, where $S_0 = 0$ and $S_T = 1$. Typically, the speed profile $\dot{S}_t$ first ramps up at a constant acceleration, before proceeding at a constant speed and finally ramping down to zero at a constant deceleration. In the case of uniform straight-line motion, the speed profile will take the form $\dot{S}_t = 1/T$. By integrating, the speed distribution function will be $S_t = t/T$.

For this particular application, the kinematic constraints of the laboratory excavator allow for digging a trench with length and depth not exceeding 600 mm and 150 mm, respectively. *Fig. 2* shows one complete digging cycle, illustrating the proposed path for the bucket. Note that each digging path is followed by picking up the soil to the point $(270, -150, 0)$ with an orientation of 180 degrees using PTP motion. This step is followed by another PTP motion to position the bucket inside the discharging area at coordinate $(100, -100, -400)$. The last step in the digging cycle is the discharging process which finishes at $(600, -100, -700)$ with an orientation of -30 degrees.

4 Teaching and learning

One of the most important features of engineering education is the combination of theoretical knowledge and practical experience. Laboratory experiments, therefore, play an important role in supporting student learning. However, there are several factors that often prevent students from having access to such ‘learning-by-doing’ interaction with robotic systems. These include their high cost, fragility and the necessary provision of skilled technical support. Nonetheless, the utilization of robots potentially offers an excellent basis for teaching in a number of different engineering disciplines, including mechanical, electrical, control and computer engineering; e.g. [11, 12, 13, 14]. Robots provide a fascinating tool for the demonstration of basic engineering problems and they also facilitate the development of skills in creativity, teamwork, engineering design, systems integration and problem solving.

In this regard, the 1/5th scale representation of LUCIE provides for the support of research and teaching in mechatronics at Lancaster University. It is a test bed for various approaches to signal processing and real-time control; and provides numerous learning opportunities and individual projects for both undergraduate and postgraduate research students. For example, since only a few minutes are needed to collect experimental data in open-loop mode, the robot arm provides a good laboratory example for demonstrating contrasting mechanistic and data-based approaches to system identification.

With regards to control system design, various classical and modern approaches are feasible. However, the present authors believe that PIP control offers an insightful introduction to modern control theory for students. Here, non-minimal state space (NMSS) models are formulated so that full state variable feedback control can be implemented directly from the measured input and output signals of the controlled process, without resort to the design and implementation of a deterministic state reconstructor or a stochastic Kalman Filter [6, 7]. Indeed, a MEng / MSc module in Intelligent Control taught in the Department covers all these areas, utilising the robot arm as a design example.

5 Control methodology

The benchmark PID controller for each joint is based on the well known Ziegler-Nichols methodology. The system is placed under proportional control and taken to the limit of stability by increasing the gain until permanent oscillations are achieved. The ‘ultimate gain’ obtained in this manner is subsequently used to determine the control gains. An alternative approach using a Nichols chart to obtain specified gain and phase margins is described by [15].

Linear PIP control is a model-based approach with a similar structure to PID control, with additional dynamic feedback and input compensators introduced when the process has second order or higher dynamics, or pure time delays greater than one sample interval. In contrast to classical methods, however, PIP design exploits the power of State Variable Feedback (SVF) methods, where the vagaries of manual tuning are replaced by pole assignment or Linear Quadratic (LQ) design [6, 7].

Finally, a number of recent publications describe an approach for nonlinear PIP control based on the identification of the following state dependent parameter (SDP) model [8],

$$y_k = \boldsymbol{w}_k^T \boldsymbol{p}_k \tag{8}$$

where,

$$\begin{aligned}
\boldsymbol{w}_k^T &= \left[\begin{array}{cccccc} -y_{k-1} & \cdots & -y_{k-n} & u_{k-1} & \cdots & u_{k-m} \end{array}\right] \\
\boldsymbol{p}_k &= \left[\begin{array}{cc} \boldsymbol{p}_{1,k} & \boldsymbol{p}_{2,k} \end{array}\right]^T \\
\boldsymbol{p}_{1,k} &= \left[\begin{array}{ccc} a_1\{\boldsymbol{\chi}_k\} & \cdots & a_n\{\boldsymbol{\chi}_k\} \end{array}\right] \\
\boldsymbol{p}_{2,k} &= \left[\begin{array}{ccc} b_1\{\boldsymbol{\chi}_k\} & \cdots & b_m\{\boldsymbol{\chi}_k\} \end{array}\right]
\end{aligned}$$

Here y_k and u_k are the output and input variables respectively, while $a_i\{\boldsymbol{\chi}_k\}\,(i=1,2,\ldots,n)$ and $b_j\{\boldsymbol{\chi}_k\}\,(j=1,\ldots,m)$ are state dependent parameters. The latter are assumed to be functions of a non-minimal state vector $\boldsymbol{\chi}_k^T$. For SDP-PIP control system design, it is usually sufficient to limit the model (8) to the case that $\boldsymbol{\chi}_k^T = \boldsymbol{w}_k^T$. The NMSS representation of (8) is,

$$\begin{aligned}
\boldsymbol{x}_{k+1} &= \boldsymbol{F}_k\boldsymbol{x}_k + \boldsymbol{g}_k u_k + \boldsymbol{d} y_{d,k} \\
y_k &= \boldsymbol{h}\boldsymbol{x}_k
\end{aligned} \tag{9}$$

where the non-minimal state vector is defined,

$$\boldsymbol{x}_k = \left[\begin{array}{ccccccc} y_k & \cdots & y_{k-n+1} & u_{k-1} & \cdots & u_{k-m+1} & z_k \end{array}\right]^T$$

and $z_k = z_{k-1} + [y_{d,k} - y_k]$ is the integral-of-error between the command input $y_{d,k}$ and the output y_k. Inherent type 1 servomechanism performance is introduced by means of this integral-of-error state. For brevity, $\{\boldsymbol{F}_k, \boldsymbol{g}_k, \boldsymbol{d}, \boldsymbol{h}\}$ are omitted here but are defined by e.g. [9, 16].

The state variable feedback control algorithm $u_k = -\boldsymbol{l}_k\boldsymbol{x}_k$ is subsequently defined by,

$$\boldsymbol{l}_k = \left[\begin{array}{ccccccc} f_{0,k} & \ldots & f_{n-1,k} & g_{1,k} & \ldots & g_{m-1,k} & -k_{I,k} \end{array}\right]$$

where $\boldsymbol{l}_k$ is the control gain vector obtained at each sampling instant by either pole assignment or optimisation of a Linear Quadratic (LQ) cost function. With regard to the latter approach, the present research uses a 'frozen-parameter' system defined as a sample member of the family of NMSS models $\{\boldsymbol{F}_k, \boldsymbol{g}_k, \boldsymbol{d}, \boldsymbol{h}\}$ to define the $\boldsymbol{P}$ matrix [9], with the discrete-time algebraic Riccati equation only used to update $\boldsymbol{l}_k$ at each sampling instant. Finally, note that while the NMSS/PIP linear controllability conditions are developed by [6], derivation of the complete controllability and stability results for the nonlinear SDP system is the subject on-going research by the authors.

6 Control design

For linear PIP design, open-loop experiments are first conducted for a range of applied voltages and initial conditions, all based on a sampling rate of 0.11 seconds. In this case, the Simplified Refined Instrumental Variable (SRIV) algorithm [17], suggests that a first order linear model with τ samples time delay, i.e. $y_k = a_1 y_{k-1} + b_\tau u_{k-\tau}$, provides an approximate representation of each joint. Here y_k is the joint angle and u_k is a scaled voltage in the range ± 1000, while $\{a_1, b_\tau\}$ are time invariant parameters. Note that the arm essentially acts as an integrator, since the normalised voltage has been calibrated so that there is no movement when $u_k = 0$. In fact, $a_1 = -1$ is fixed *a priori*, so that only the numerator parameter b_τ is estimated in practice for linear PIP design.

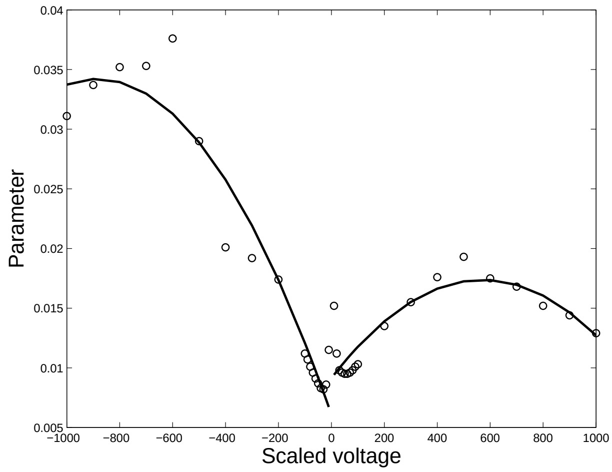

Figure 3: Variation of b_τ against input demand for the boom.

With $\tau = 1$, the dipper and bucket joints appear relatively straightforward to control using linear PIP methods. In this case, the algorithm reduces to a PI structure [6], hence the implementation results are similar to the PI algorithm tuned using classical frequency methods. As would be expected, the difference between the classical and PIP methods for these joints is qualitative. Such differences relate only to the relative ease of tuning the algorithm to meet the stated control objectives.

By contrast, with $\tau = 2$, the slew and boom joints are better controlled using PIP methods since (as shown in numerous earlier publications) the latter automatically handles the increased time delay [9]. Of course, an alternative solution to this problem would be to introduce a Smith Predictor into the PI control structure. The authors are presently investigating the relative robustness of such an approach in comparison to PIP methods.

However, further analysis of the open-loop data reveals limitations in the linear model above. In particular, the value of b_τ changes by a factor of 10 or more, depending on the applied voltage used, as illustrated in *Fig. 3* for the case of the boom. Here, numerous experiments are conducted for a range of applied voltages and, in each case, SRIV methods used to estimate linear models. *Fig. 3* illustrates these estimates of b_τ plotted against the magnitude of the step input (the solid trace represents a straightforward polynomial fit).

In fact, SDP analysis suggests that a more appropriate model for the boom takes the form of equation (8) with,

$$\begin{aligned}
\boldsymbol{w}_k^T &= \left[\begin{array}{ccc} -y_{k-1} & u_{k-1} & u_{k-2} \end{array}\right] \\
\boldsymbol{p}_k &= \left[\begin{array}{ccc} a_1\{\boldsymbol{\chi}_k\} & 0 & b_2\{\boldsymbol{\chi}_k\} \end{array}\right]^T
\end{aligned} \tag{10}$$

where,

$$\begin{aligned}
a_1\{\boldsymbol{\chi}_k\} &= 0.238 \times 10^{-6} u_{k-2}^2 - 1 \\
b_2\{\boldsymbol{\chi}_k\} &= -5.8459 \times 10^{-6} u_{k-2} + 0.01898
\end{aligned}$$

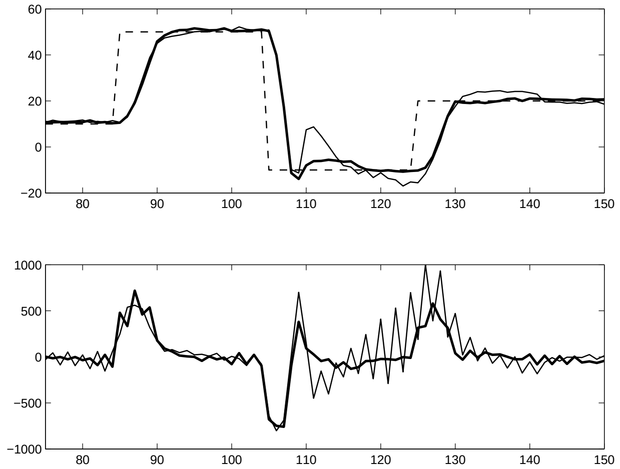

Figure 4: Top: linear PIP (thin trace), nonlinear SDP-PIP (thick) and command input (dashed) for the boom angle, plotted against sample number. Bottom: equivalent control inputs.

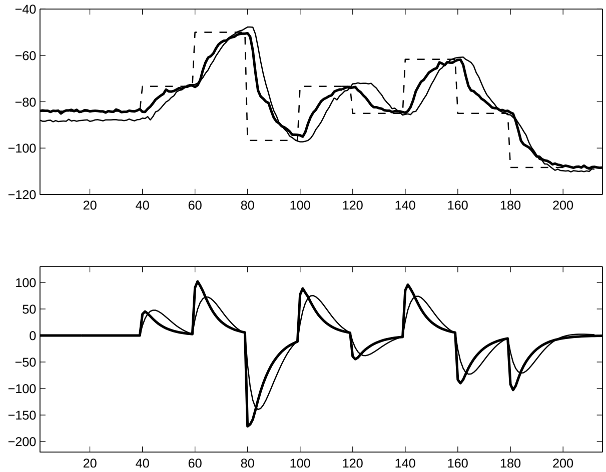

Figure 5: Top: linear PIP (thin trace), nonlinear SDP-PIP (thick) and command input (dashed) for the dipper angle, plotted against sample number. Bottom: equivalent control inputs.

The associated SDP-PIP control algorithm takes the form,

$$u_k = -\left[\begin{array}{ccc} f_{0,k} & g_{1,k} & -k_{I,k} \end{array}\right] \cdot \left[\begin{array}{ccc} y_k & u_{k-1} & z_k \end{array}\right]^T \qquad (11)$$

where the gains $f_{0,k}$, $g_{1,k}$ and $k_{I,k}$ are updated at each sampling instant in the manner of a scheduled controller. Full details of this approach and the equivalent SDP-PIP algorithms for the dipper, bucket and slew joints are given by Shaban [9].

7 Implementation

Typical implementation results for the boom arm are illustrated in *Fig. 4*, where it is clear that the SDP-PIP algorithm is more robust than the fixed gain, linear PIP algorithm (or equivalent classical PI controller) to large steps in the command level. Furthermore, the nonlinear approach yields a considerably smoother control input signal.

Note that the linear and nonlinear controllers are designed to yield a similar speed of response in the theoretical case, i.e. the differences seen in *Fig. 4* are due to the variation in b_2 (*Fig. 3*) which is only taken account of in the SDP-PIP case. It should pointed out that the response time for this example has been deliberately increased to the practical limit of robust linear PIP design, in order to emphasis these differences.

Fig. 5 shows control of the dipper arm for a similar experiment. Although the differences between the linear and nonlinear approaches are often relatively small when each joint is examined in isolation for movement in air, as in *Fig. 5*, such differences are multiplied up when the bucket position is finally resolved in the sandpit. In this regard, Table 1 compares the response time of the linear PIP and SDP-PIP approaches, represented by the number of seconds taken to complete three complete trenches, each consisting of 9 digging cycles. Here, the improved joint angle control allows for a faster SDP-PIP design, typically yielding a 10% improvement in the digging time.

Table 1: Time taken to complete one trench.

Trench	Linear PIP	SDP-PIP
1	338.46s	369.01s
2	334.39s	370.43s
3	336.13s	372.86s

Finally, *Fig. 6* illustrates typical SDP-PIP implementation results for one cycle of the bucket showing a 3D co-ordinate plot of the end-effector. This graph shows the bucket being first lowered into and subsequently being dragged through the sand, followed by extraction, displacement and release.

8 Conclusions

This paper has described the control of a laboratory robot arm, representing a 1/5th scale model of an autonomous excavator, developed for both research and teaching at Lancaster University. In contrast to earlier research [15], the present paper considers the implementation of the complete control system for automatically digging a trench in the sandpit.

Classical and modern approaches have been evaluated for joint control, including: Proportional-Integral (PI) control tuned by conventional Ziegler-Nichols rules; linear Proportional-Integral-Plus (PIP) control; and a novel, nonlinear PIP design based on the identification of State Dependent Parameter (SDP) models. Here, linear PI algorithms tuned using either classical or PIP methods initially appear sufficient for control of the dipper and bucket angles. By contrast, the slew and boom joints are better controlled using the higher order PIP algorithm that automatically handles the pure time delays.

However, inherent nonlinearities in all these joints prove problematic when the feedback controllers are combined with kinematic equations to control the position of the end effector, particularly once the bucket in moved into the sand. In fact, the increased complexity of the nonlinear SDP-PIP approach appears justified here, given the improved closed-loop performance. In particular, the time taken to complete an entire digging cycle is reduced by $\approx 10\%$. Finally, the experience gained using the laboratory excavator has recently been exploited for the development of a full scale vibro-lance system used for ground improvement on a construction site; see e.g. [16].

Acknowledgments

The authors are grateful for the support of the Engineering and Physical Sciences Research Council (EPSRC). The statistical tools used in this paper have been assembled as the CAPTAIN toolbox [18] within the Matlab$^{\text{TM}}$ software environment, available for download at: http://www.es.lancs.ac.uk/cres/captain/

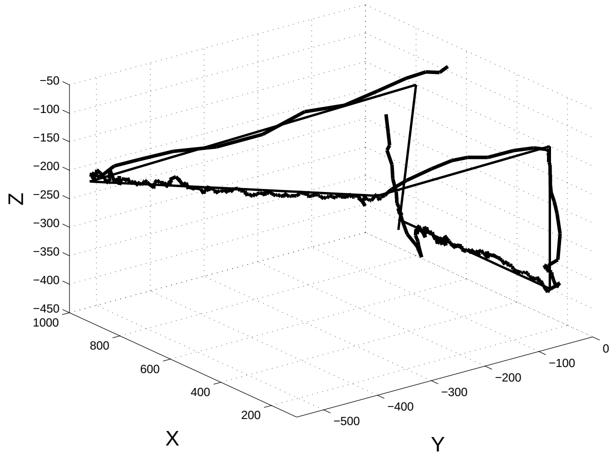

Figure 6: Resolved position $\{X, Y, Z\}$ of the end-effector in 3-D space, with the set-point shown as straight lines (mm).

References

[1] R. L. Tucker. Construction automation in the USA. In *Proceedings 16th Internation Symposium on Automation and Robotics in Construction*, Madrid, Spain, 1999.

[2] Q. Ha, Q. Nguyen, D. Rye, and H. Durrant-Whyte. Impedance control of a hydraulic actuated robotic excavator. *J. Automation in Construction*, 9:421–435, 2000.

[3] E. Budny, M. Chlosta, and W. Gutkowski. Load-independent control of a hydraulic excavator. *Journal of Automation in Construction*, 12:245–254, 2003.

[4] J. Gu, C. J. Taylor, and D. W. Seward. The automation of bucket position for the intelligent excavator LUCIE using the proportional-integral-plus (PIP) control strategy. *Journal of Computer-Aided Civil and Infrastructure Engineering*, 12:16–27, 2004.

[5] D.A. Bradley and D.W. Seward. The development, control and operation of an autonomous robotic excavator. *Journal of Intelligent Robotic Systems*, 21:73–97, 1998.

[6] P. C. Young, M. A. Behzadi, C. L. Wang, and A. Chotai. Direct digital and adaptive control by input-output, state variable feedback pole assignment. *International Journal of Control*, 46:1867–1881, 1987.

[7] C. J. Taylor, P. C. Young, and A. Chotai. State space control system design based on non-minimal state-variable feedback : Further generalisation and unification results. *International Journal of Control*, 73:1329–1345, 2000.

[8] P. C. Young. Stochastic, dynamic modelling and signal processing: Time variable and state dependent parameter estimation. In W. J. Fitzgerald, editor, *Nonlinear and Nonstationary Signal Processing*. Cambridge University Press, Cambridge, 2000.

[9] E. M. Shaban. *Nonlinear control for construction robots (in preparation)*. PhD thesis, Lancaster University, Engineering Department, 2006.

[10] K. Yoram. *Robotics for engineers*. McGraw-Hill Book Company, 1985.

[11] D. Gustafson. Using robotics to teach software engineering. In *Frontiers in Education Conference (FIE'98)*, Tempe, Arizona, 1998.

[12] R. S. Dannelly. Use of a mobile robot in a data structures course. *The Journal of Computing in Small Colleges*, 15(3):85–90, 2000.

[13] C. Paul, V. Hafner, and J.C. Bongard. Teaching new artificial intelligence using constructionist edutainment robots. In *Workshop on Edutainment Robots'00*, Sankt Augustin, Germany, 2000.

[14] R. A. C. Bianchi and A. La Neve. Studying electrical and computer engineerng through the construction of robotic teams and systems. In *International Conference on Engineering Education (ICEE'02)*, Manchester, UK, 2002.

[15] R. Dixon, C. J. Taylor, and E. M Shaban. Comparison of classical and modern control applied to an excavator-arm. In *IFAC 16th Triennial World Congress*, Prague, Czech Republic, July 2005.

[16] C. J. Taylor, E. M. Shaban, A. Chotai, and S. Ako. Nonlinear control system design for construction robots using state dependent parameter models. In *UK-ACC International Conference*, Glasgow, UK, September 2006.

[17] P. C. Young. *Recursive Estimation and Time Series Analysis*. Springer-Verlag, Berlin, 1984.

[18] C. J. Taylor, D. J. Pedregal, P. C. Young, and W. Tych. Environmental time series analysis and forecasting with the Captain toolbox. *Environmental Modelling and Software*, 2006 (in press).