Monograph:
Harrison, R.F. (2001) Asymptotically Optimal Stabilizing Quadratic Control of an Inverted Pendulum. Research Report. ACSE Research Report 824 . Department of Automatic Control and Systems Engineering

Asymptotically Optimal Stabilizing Quadratic Control of an Inverted Pendulum

Robert F Harrison

Department of Automatic Control and Systems Engineering

The University of Sheffield

Mappin Street, Sheffield, S1 3JD, UK

Research Report No. 824

Abstract

A new method for the design and synthesis of near-optimal, non-linear control laws is examined, based on a generalisation of LQ optimal control theory and which effectively provides a near-optimal gain-schedule. The method is simple to apply and affords greater design flexibility (via state-dependent weighting) than conventional approaches. The resulting regulator can, in principle, be implemented in real-time owing to the causal nature of the required computations. However, the need to solve an algebraic Riccati equation at every time-point is burdensome and a number of algorithms that would permit parallel computation is discussed. The problem of stabilizing an inverted pendulum is used to illustrate the method and proves an exacting task that highlights a number of issues.

1 Introduction

The aim of this paper is to examine a simple algorithm for the quadratic optimal control of a wide class of non-linear systems that is causal and can, in principle, be implemented in real-time. The method is based on a generalisation of the well-known linear quadratic optimal control theory. We concentrate on the use of the approach and its pros and cons, rather than the theoretical aspects of its derivation which can be found in [1-3]. The method is demonstrated via the well-established problem of balancing a single inverted pendulum on a cart.

Linear quadratic (LQ) optimal control theory is a highly developed approach for the synthesis of linear optimal control laws that has been widely applied. In particular, the infinite-time-horizon solution has appeal for the regulation of processes that are well modelled by linear time-invariant dynamics because the solution comprises a set of static gains that are calculated once, off-line, and are implemented causally thereafter. The finite-time-horizon solution, while being more general, and admitting of time-varying dynamics or weighting parameters, is essentially an off-line procedure because the associated (differential) Riccati equation must be solved backwards in time. Incorporation of such a scheme as a closed-loop, real-time process is not therefore possible. The receding horizon approach attempts to overcome this difficulty by repeatedly solving the open-loop, finite-time-horizon problem for short periods into the future, and using this for the feedback gain over the time step. It does this at the expense of optimality, which may or may not be important from the point of view of the practising engineer.

A further disadvantage of the LQ philosophy is that, being a linear feedback, the control signal is affected in the same way by small and large signals. In many applications it may be preferable to ignore small error signals (due, for instance, to measurement noise) as far as possible, while responding optimally to large errors. It

may also be desirable to switch attention between control objectives depending on their current values. The receding horizon strategy approximates the former behaviour to a certain extent, but does not address the latter.

For non-linear dynamics the situation is exacerbated, with few explicit solutions for their optimal quadratic control as yet known, except those based on series expansions. These are unrealisable, unless via truncation, leading to a loss of optimality and possibly stability. Normally optimal quadratic control for non-linear systems is conducted numerically and tends, inherently, to be non-causal.

In this paper we make use of a result that generalises the LQ theory to non-linear systems to provide a non-linear design method that overcomes some of the difficulties mentioned above [1,2] and also in [3] although [1] is not cited therein. This non-linear quadratic (NLQ) method applies to systems having a broad class of non-linear dynamics with state-dependent weighting matrices (the design degrees of freedom). In brief, it turns out that the infinite-time-horizon LQ regulator problem when solved afresh at every point on the state trajectory leads to an asymptotically optimal control policy in that it converges to the optimal control close to the origin [2]. For admissible system dynamics, the weighting parameters can be made to be functions of the state variables. Thus, in addition to handling non-linear dynamics, the design stage allows for the introduction of state-dependence in the weighting matrices, leading to a more flexible control strategy [4,5]. Although, in principle, time-varying weighting parameters are allowed in the (finite-time) LQ approach, lack of prior knowledge of potential disturbances and the acausal calculation for the solution makes the introduction of these difficult in practice. The required amplitude dependence can never, therefore, be achieved through the LQ approach. The solution to the proposed method is causal, but has considerable computational overhead. However, by using a solution to the Riccati equation based upon the matrix sign function [6], it is possible to derive a parallel algorithm [7] that may be suitable for real-time implementation.

1.1 Linear quadratic regulator (LQR)

The LQ optimal regulation problem is expressed as follows: minimise with respect to the control signal, $\mathbf{u}$, the cost function

$$J = \int_0^\infty \left(\mathbf{x}^t Q \mathbf{x} + \mathbf{u}^t R \mathbf{u} \right) dt \tag{1}$$

subject to the linear time invariant dynamics:

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} \tag{2}$$

where $\mathbf{x}$ is an n-vector of system states, $\mathbf{u}$ is an m-vector of control variables, A and B are matrices of appropriate dimension and the superscript, t, indicates transposition. The matrices Q and R are positive semi-definite and definite, respectively, and are used to penalise particular states and controls according to the engineering objective.

It is well known, e.g. [8], that the control policy which solves the above problem is a linear combination of the system states, given by:

$$\mathbf{u} = K\mathbf{x} \tag{3}$$

$$K = -R^{-1} B^t P \tag{4}$$

where P is the positive definite solution of the algebraic matrix Riccati equation (5). A unique, positive definite solution to the above exists if the pair (A, B) is stabilizable and (A, Γ) is detectable, with $Q = \Gamma^t \Gamma$.

$$0 = PA + A^t P - PBR^{-1}B^t P + Q \tag{5}$$

1.2 Non-linear quadratic regulator (NLQR)

The extension of the above to non-linear systems looks identical [1-3], except that, instead of performing a single optimisation and applying the resulting gain-matrix for all time, the optimisation has to be carried out at every time step. Consider a non-linear dynamical system that can be expressed in the form:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}) = A(\mathbf{x})\mathbf{x} + B(\mathbf{x})\mathbf{u} \\ \mathbf{f}(\mathbf{0}, \mathbf{0}) &= \mathbf{0} \end{aligned} \tag{6}$$

where the Jacobians of $A(\mathbf{x})$ and $B(\mathbf{x})$ are subject to some bounded growth conditions (Lipschitz), then at each point, $\bar{\mathbf{x}}$, on the state trajectory, a linear system is defined with fixed $A = A(\bar{\mathbf{x}})$ and $B = B(\bar{\mathbf{x}})$. Provided that the matrices, A and B, converge to their conventional Taylor series linearisation at the origin it can be shown [2] that solving the infinite-time LQ optimal control problem, point-wise on the state trajectory and applying the resultant control, results in an asymptotically optimal, stabilising quadratic control policy for systems described by equation (6). The requirement

$$\begin{aligned} A(\mathbf{x}) &\rightarrow \partial \mathbf{f}(\mathbf{x}, u)/\partial \mathbf{x}\big|_{\mathbf{x}=0,\mathbf{u}=0} \\ B(\mathbf{x}) &\rightarrow \partial \mathbf{f}(\mathbf{x}, \mathbf{u})/\partial \mathbf{u}\big|_{\mathbf{x}=0,\mathbf{u}=0} \end{aligned} \tag{7}$$

is imposed to ensure that $P(\mathbf{x})$ tends to the solution for the linearised problem at the origin. Hence, in a small enough neighbourhood of the origin the feedback from the algorithm is arbitrarily close to the optimal feedback [2]. Appendix A contains a brief, heuristic justification of the theory [2].

So, by choosing the $\mathbf{u}$ that minimises the usual quadratic cost function at every time point, we have a near-optimal control policy for a very wide class of non-linear systems. Evidently, $A(\bar{\mathbf{x}}), B(\bar{\mathbf{x}}), Q$ and R are subject, point-wise, to the same conditions as for the linear case. It is clear that the proposed solution is identical to the one obtained from equations (3), (4) and (5) when the dynamics are linear.

Because the control synthesis takes place point-wise, the designer is now free to select Q and R in ways that may be more directly applicable to the control *engineering* objectives. These can be made to be functions of the instantaneous values of the state variables so that

$$J = \int_0^\infty \left(\mathbf{x}^t Q(\bar{\mathbf{x}})\mathbf{x} + \mathbf{u}^t R(\bar{\mathbf{x}})\mathbf{u} \right) dt \tag{8}$$

subject to the requirements for the solution of the Riccati equation and the invertability of R.

Ensuring that $A(\bar{\mathbf{x}}), B(\bar{\mathbf{x}}), R(\bar{\mathbf{x}})$ and $Q(\bar{\mathbf{x}})$ satisfy these requirements *a priori*, is difficult in general, however, for functions whose controllability (observability)

matrices are full rank except at isolated points (e.g. those admitting convergent Taylor series expansions), the required properties can only be lost at isolated points and may not, therefore, be harmful in practice provided no attempt is made to remain in their vicinity. We postulate that, since many systems can be well approximated in the region of interest by polynomial functions, loss of stabilizability (detectability) is unlikely to be generic.

2 Solving the matrix Riccati equation

In order to calculate the optimal solution, it is necessary to solve the matrix Riccati equation at each point in time. In practice this will be done in a computer and it will be necessary to solve the equation at each discrete time-step. The usual approach to the solution of this problem is via an eigen-decomposition of the Hamiltonian matrix for the system [9]. For sizeable dynamics such an approach is computationally intensive and may not be able to deliver solutions at the required sample rate (i.e. in "real-time"). It can also be numerically sensitive, depending as it does on the numerical solution of an eigen-problem.

The matrix-sign-function (MSF) is an appealing alternative to eigen-decomposition owing to its simplicity (Roberts, 1980), requiring only the operations of matrix inversion, addition, and scalar multiplication [6] as follows:[7]

$$Z_0 = H;\ Z_{i+1} = \frac{1}{2}\left(\frac{1}{c_i}Z_i + c_i Z_i^{-1}\right) \tag{9}$$

where $c_i = \left|\det(Z_i)\right|^{1/2n}$ is a scaling used to speed convergence, and n is the dynamical order. H is the Hamiltonian matrix of order $2n$, given by: $H = \begin{bmatrix} A^t & Q \\ BR^{-1}B^t & -A \end{bmatrix}$

Assuming H has no eigen-values on the imaginary axis Z_i converges to $\mathrm{Sign}(H) = S$, say, [6]. For the solution of the Riccati equation we require the quantity [10]

$S^+ = \mathrm{Sign}^+(S) = \frac{1}{2}(I_{2n} + S)$.

Now, by decomposing S^+ thus: $S^+ = \begin{bmatrix} V & W \end{bmatrix}$ we write, P, the solution of the Riccati equation (5) thus:

$$P = V^t W\left(W^t W\right)^{-1} \tag{10}$$

The convergence of the algorithm has been investigated in [10] and its global convergence is established in [11]. Its simplicity also suits the MSF algorithm to parallel computation [6]. In [7] an algorithm for diagonal pivoting factorisation – a form of Gaussian elimination with partial pivoting – is used to develop a parallel algorithm involving no sequential computations. Complexity is of order $(2n)^3$, which dominates the communication burden. The speed-up achieved over the conventional method [9] is demonstrated on a hypercube architecture. More recently, a Jacobi-like method has been proposed with a simple function of the matrix size to predict the number of iterations needed for convergence [12]. This, of course, does not guarantee that a solution can be found within any arbitrary sampling period.

We do not implement the parallel solution here. The discussion is included simply to underline that real-time operation may be possible.

3 Asymptotically optimal quadratic control of a single inverted pendulum

It is well known from energetic considerations that the motion of the inverted pendulum on a cart constrained to move in a plane (Figure 1) can be described thus:

$$\begin{aligned}(M+m)\ddot{x} &= mL\dot{\theta}^2\sin(\theta) - mL\cos(\theta)\ddot{\theta} + F \\ \left(J+mL^2\right)\ddot{\theta} &= mgL\sin(\theta) - mL\cos(\theta)\ddot{x}\end{aligned} \tag{11}$$

where M and m (kg) denote the mass of the cart and the pendulum respectively, L (m) is the length of the pendulum and J (kg m^2) is its moment of inertia about the centre of mass. $\theta(t)$ (rad) denotes angular displacement from the vertical and $x(t)$ (m), lateral displacement from the mid-point of the track. $F(t)$ (N) is the applied force.

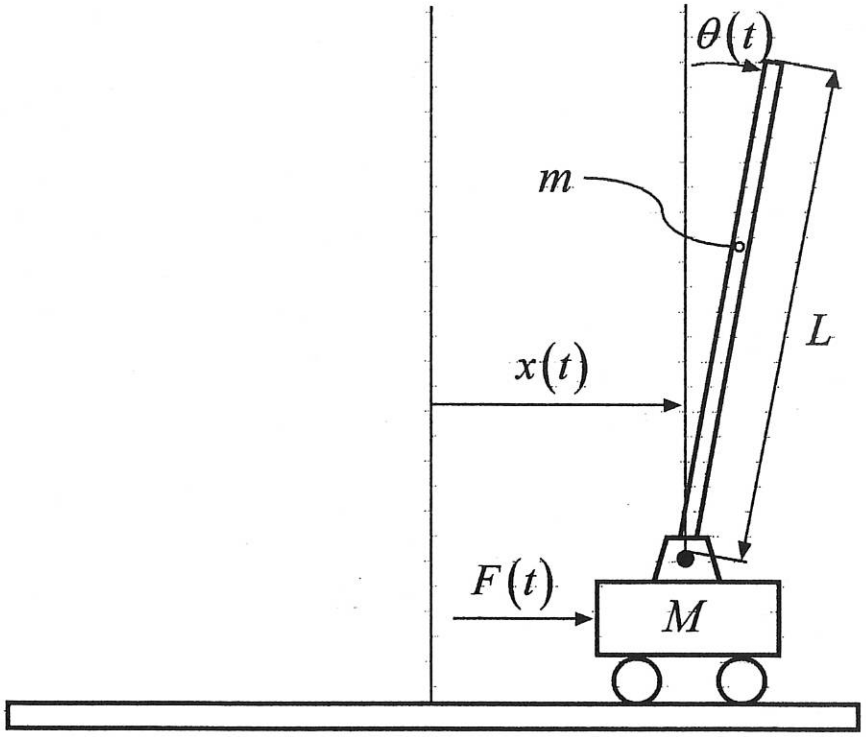

Figure 1 Schematic of a single inverted pendulum on a cart.

By substitution in (11) we obtain the explicit state space form:

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{-\left(mLx_4^2\sin(x_3)\left(J+mL^2\right) - m^2L^2g\sin(x_3)\cos(x_3) + \left(J+mL^2\right)u\right)}{m^2L^2\cos^2(x_3) - \left(J+mL^2\right)(M+m)} \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= \frac{-mL\left(g\sin(x_3)(M+m) - mLx_4^2\cos(x_3)\sin(x_3) - \cos(x_3)u\right)}{m^2L^2\cos^2(x_3) - \left(J+mL^2\right)(M+m)}\end{aligned} \tag{12}$$

where $x_1 = x$, $x_2 = \dot{x}$, $x_3 = \theta$, $x_4 = \dot{\theta}$ and $u = F$. Recall that we require the origin of state-space to be an equilibrium point and here the equilibrium of interest is $\underline{x} = \underline{0}$. For convenience we define: $d(x_3) = m^2L^2\cos^2(x_3) - (J + mL^2)(M + m)$ and write:

$$A(\mathbf{x}) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{-mL\sin(x_3)\left(x_4^2(J+mL^2) - mLg\cos(x_3)\right)}{d(x_3)x_3} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \dfrac{-mL\sin(x_3)\left(g(M+m) - mLx_4^2\cos(x_3)\right)}{d(x_3)x_3} & 0 \end{bmatrix} \tag{13}$$

$$B(\mathbf{x}) = \begin{bmatrix} 0 \\ \dfrac{-(J+mL^2)}{d(x_3)} \\ 0 \\ \dfrac{mL\cos(x_3)}{d(x_3)} \end{bmatrix} \tag{14}$$

or alternatively,

$$A(\mathbf{x}) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{m^2L^2g\sin(x_3)\cos(x_3)}{d(x_3)x_3} & \dfrac{-mL\sin(x_3)x_4(J+mL^2)}{d(x_3)} \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \dfrac{-mL\sin(x_3)g(M+m)}{d(x_3)x_3} & \dfrac{m^2L^2\sin(x_3)x_4\cos(x_3)}{d(x_3)} \end{bmatrix} \tag{15}$$

Evidently, the terms $\sin(x_3)/x_3$ in (13) are well behaved at the origin and $d(0)$ is non-zero. It can be easily verified that in either case, (13) or (15), $A(\mathbf{x}), B(\mathbf{x})$ satisfy the conditions imposed by (7). To evaluate the stabilizability of $A(\mathbf{x}), B(\mathbf{x})$ we simply form the determinant of the controllability matrix, given by:

$$D_c(\mathbf{x}) = \det\left(\begin{bmatrix} B & AB & A^2B & A^3B \end{bmatrix}\right) = \frac{\cos^2(x_3)\sin^2(x_3)L^4m^4g^2}{x_3^2d^4(x_3)}$$

which becomes zero only for $\pm k\pi/2$, $k = 1, \ldots$. In Figure 2 a plot of this determinant (choosing parameters as per the next section) indicates regions of poor controllability around the zeros of $D_c(\mathbf{x})$ (approximately $\pm(80-100°)$ and $|\theta| < (170°-180°)$. Further, local analyses of the system in these regions confirms that it is the unstable mode that becomes uncontrollable in both cases leading to a loss of stabilizability and hence a failure to fulfil the necessary requirements for the solution of the ARE. Trajectories entering these regions can be expected to cause numerical difficulties in the ARE solver and this will indeed be seen to be the case. Clearly, the detectability

requirement can be guaranteed, assuming a diagonal Q, by ensuring that all $q_{ii} > 0\ i = 1,\ldots,4$.

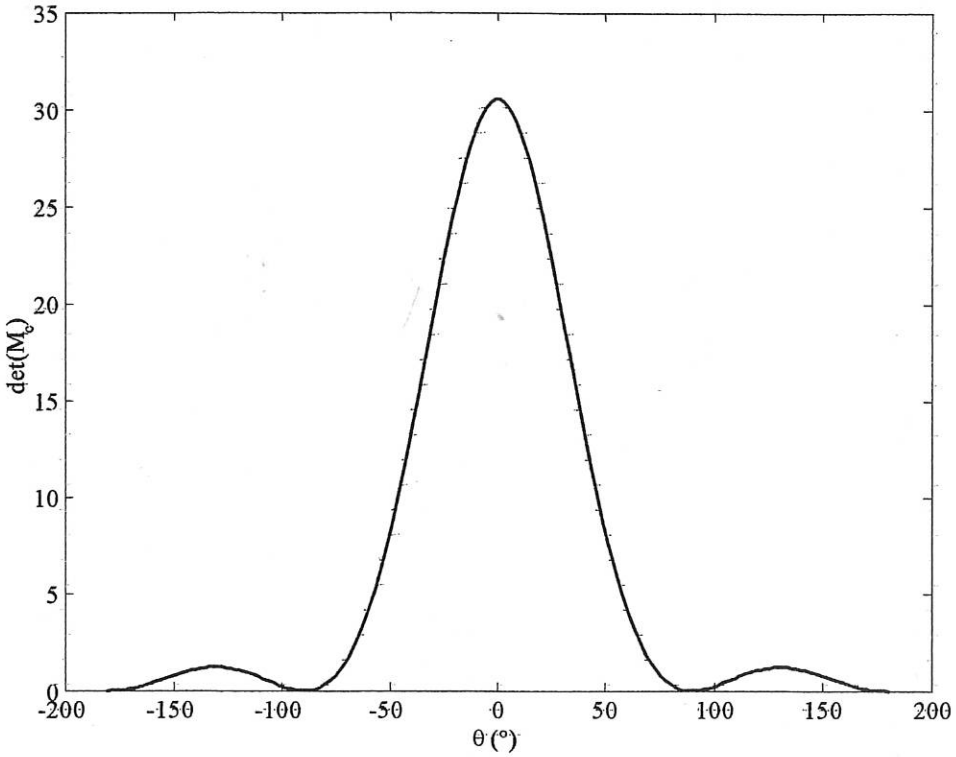

Figure 2 Determinant of the controllability matrix as a function of angular displacement.

4 Results

We use parameter settings corresponding to the inverted pendulum rig that resides in the Department of Automatic Control & Systems Engineering of the University of Sheffield, namely: $M = 1.92kg$, $m = 0.185kg$, $L = 0.508m$, and with $g = 9.81ms^{-2}$. The pendulum comprises a uniform pole and no "bob" giving $J = mL^2/3$.

4.1 Comparison with linear LQR solution

First we look simply at how the NLQR solution behaves in comparison with the LQR solution i.e. the non-linear system compensated with a set of linear gains computed at the operating point for the same values of Q,R. We choose, arbitrarily, $Q = \text{diag}\{100.0,\ 0.1,\ 1.0,\ 0.1\}$ and $R = 1$ to reflect that the linear displacement will be a major constraint on any practical implementation and consider only initial angular displacements, all other initial values being set to zero. After some experimentation we find that the LQR solution is unstable for initial conditions $|\theta(0)| > 55°$, while the NLQR solution remains stable for $|\theta(0)| < 74°$ and $98° < |\theta(0)| < 180°$. This indicates a degree of asymmetry in the numerical conditioning of the problem in that it is possible to approach the singularity at $\pm 180°$ much more closely than it is possible to approach $\pm 90°$.

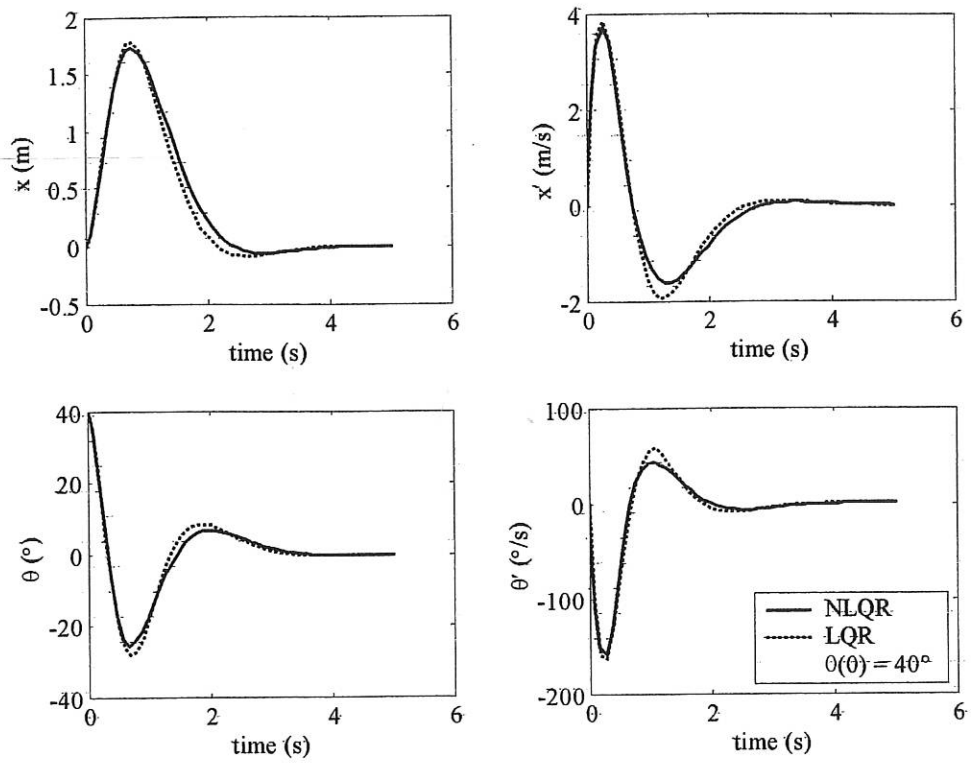

Figure 3: Comparison of state evolution for system under non-linear (NLQR) and linear (LQR) control for an initial angle of 40°.

As might be expected, for small excursions the NLQR and LQR solutions are approximately equal. Figure 3 illustrates this for an initial angular displacement of $40°$. This is the value at which the two solutions first begin to deviate. Notice that the non-linear controller generates less marked excursions in line with what is expected.

Moving now to the limit at which the current LQR design is able to stabilize the system $\left(\theta(0) = 55°\right)$ we note substantial differences in the two behaviours (Figure 4). While the non-linear regulator results in state trajectories qualitatively similar to those in Figure 3 – with correspondingly larger magnitudes – the linear compensator demonstrates wilder behaviour and a far higher total variation.

Figure 5 indicates that the behaviour of the control force can be similarly summarised. It is interesting to note that behaviour up to the first reversal in the states and, to a certain extent, the force, is approximately equal for both approaches.

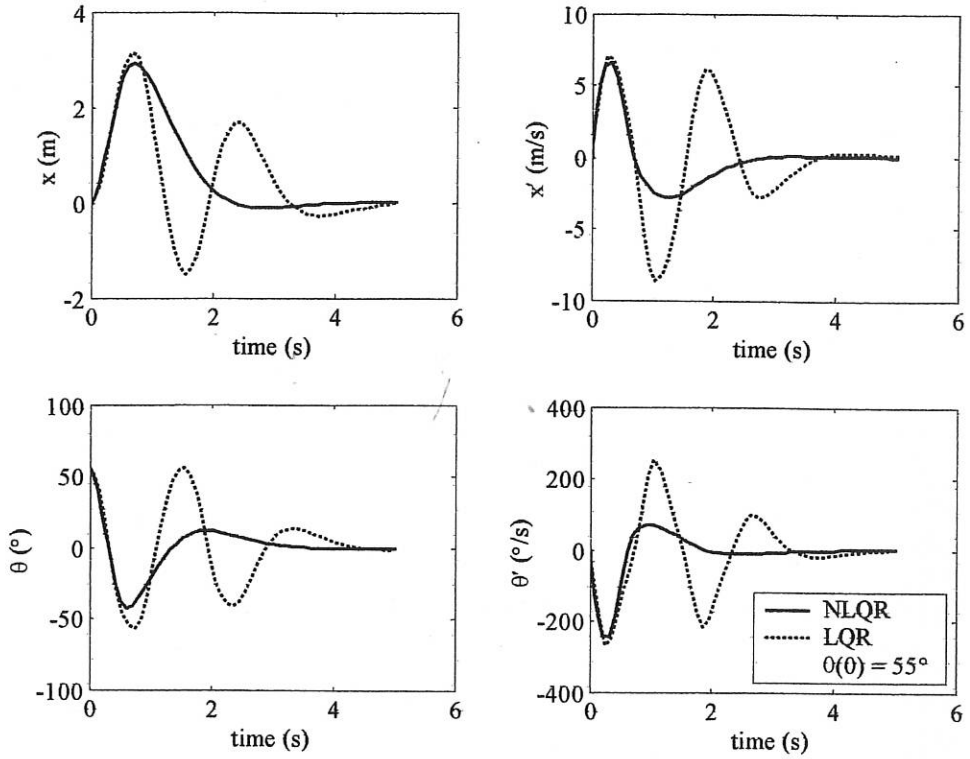

Figure 4: Comparison of state evolution for system under non-linear (NLQR) and linear (LQR) control for an initial angle of 55°.

Beyond the 55° limit we can only report the behaviour of the NLQR solution. As noted earlier, that, while only isolated points cause a failure to meet the required stabilizability conditions, numerical sensitivity excludes larger regions from effective control. Figure 6 shows the state evolution for three large initial angles, $\theta(0) = 70°, 110°, 140°$ avoiding these difficult regions. In the first instance, the physical situation corresponds more closely with the previous configurations, i.e. the pendulum starts above horizontal. The non-linear regulator attempts, therefore, to lift the rod to the vertical, demanding a large initial control force to do this. The state and control force evolution is qualitatively similar to that of the earlier examples but is more vigorous still. For initial angles that leave the rod hanging below the horizontal we see qualitatively different behaviour. The early linear motion is in the opposite direction leading to a more nearly monotonic reduction in angular displacement rather than the previous oscillatory approach to zero. While the state trajectories appear reasonably smooth in these cases, the required control force is highly non-smooth. This arises from the need to pass through regions of low stabilizability and, indeed, to cross the singularity at $\theta = 90°$. This causes an impulsive force to be generated and a reversal in sign. Evidently system inertia serves to smooth this effect but, nonetheless, it can be seen as a serious limitation. The fact that the system was able to recover reinforces our earlier assertion that it may still be practically possible to pass through regions of low stabilizability provided that it is done quickly enough. Figure 7 is difficult to read but shows the control forces for the three cases.

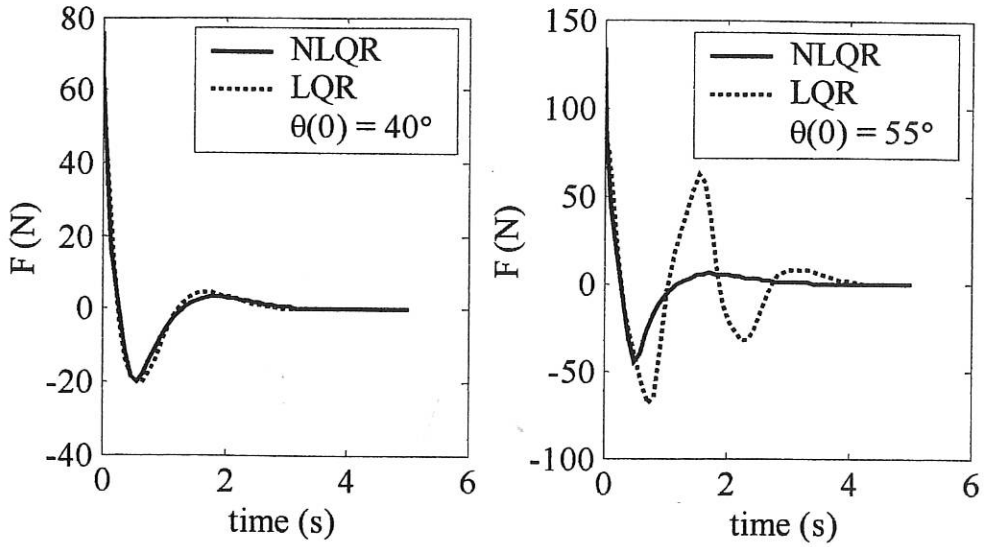

Figure 5: Comparison of control force for system under non-linear (NLQR) and linear (LQR) control for an initial angles of 40° and 55° .

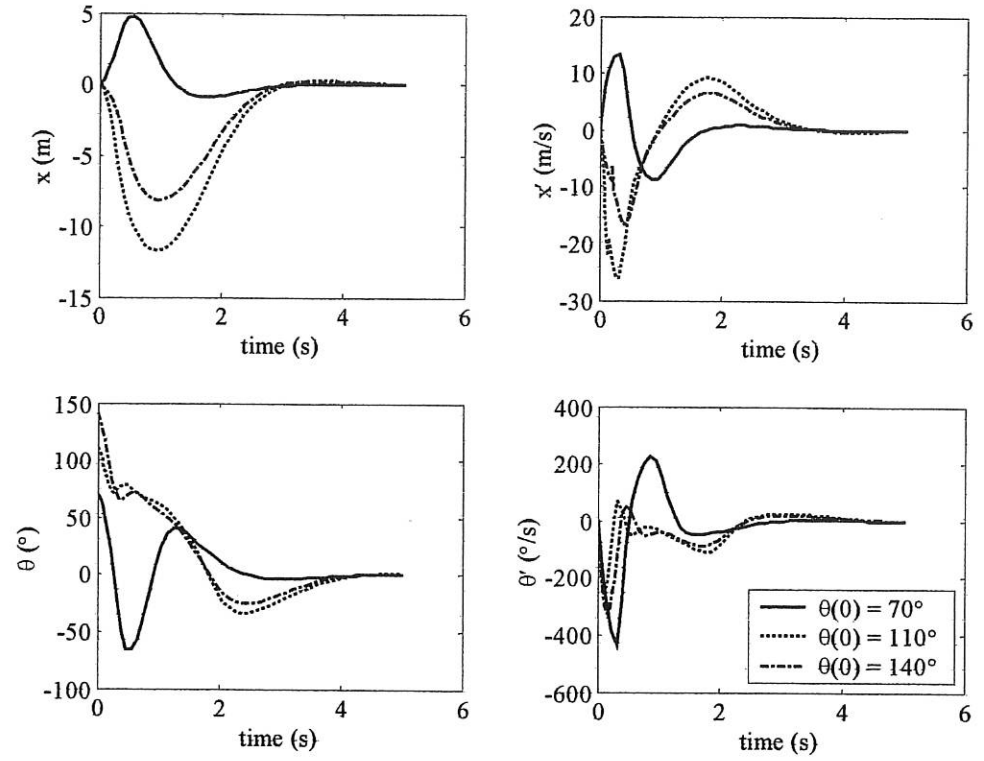

Figure 6: Comparison of state evolution under non-linear control for *large* initial angles.

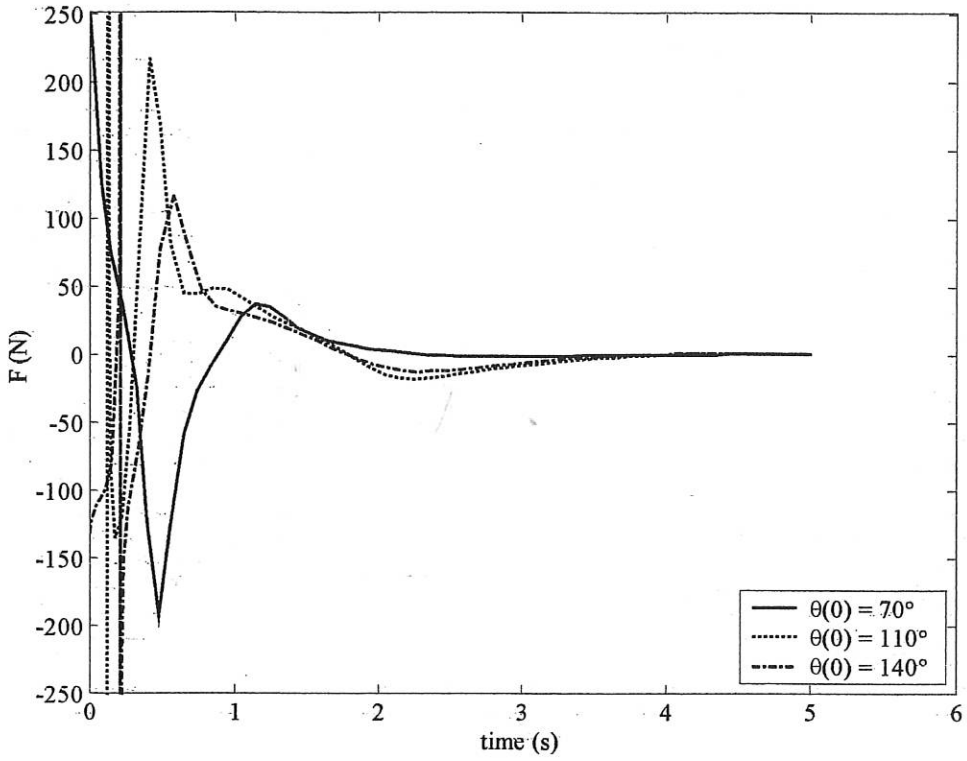

Figure 7: Control forces generated for the simulations of Figure 6. Note that for initial angles greater than 90° large impulsive forces are required at times. These are associated with regions on the state trajectory corresponding to poor stabilizability and generate very sharp reversals in linear velocity.

4.2 Non-linearity in the control – saturation

Realistically there will be hard constraints on the control force that can be achieved and this is usually modelled as a piecewise linear, saturation non-linearity. From (6) it is clear that control non-linearities are not permitted and to overcome this we solve a closely related problem that introduces integral action through state augmentation. We outline this for the case of scalar inputs only. Let the control input be given by $u = \phi(x_{n+1})$ then, defining $\dot{x}_{n+1} = \nu$ we can re-write the model in (6) thus:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{x}_{n+1} \end{bmatrix} = \begin{bmatrix} A(\mathbf{x}) & \dfrac{B(\mathbf{x})\phi(x_{n+1})}{x_{n+1}} \\ 0_{1\times n} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ x_{n+1} \end{bmatrix} + \begin{bmatrix} 0_{n\times 1} \\ 1 \end{bmatrix} \nu \tag{16}$$

and find the optimal policy that minimises:

$$J = \int_0^\infty \left(\mathbf{x}_a^t Q_a(\bar{\mathbf{x}}_a)\mathbf{x}_a + \nu^t R_a(\bar{\mathbf{x}}_a)\nu \right) dt \tag{17}$$

where we have used the subscript a to indicate the augmented system. The state weighting matrix now becomes:

$$Q_a = \begin{bmatrix} Q & 0 \\ 0 & \phi^2(x_{n+1})R \end{bmatrix}$$

and is itself state-dependent. Subject to the point-wise detectability and stabilizability conditions being satisfied, a problem that is close to the original (assuming R_a is small) can be solved.

Returning now to the example and modelling the saturation as a smooth function:

$$\sigma_\lambda(x)=\begin{cases}\lambda & x>\lambda\\ \lambda\sin\left(\dfrac{\pi x}{2\lambda}\right) & |x|\le\lambda\\ -\lambda & x<-\lambda\end{cases} \tag{18}$$

we apply a saturation limit of 40 N, choose $R_a = 0.001$ and re-address the situation of Figure 3.

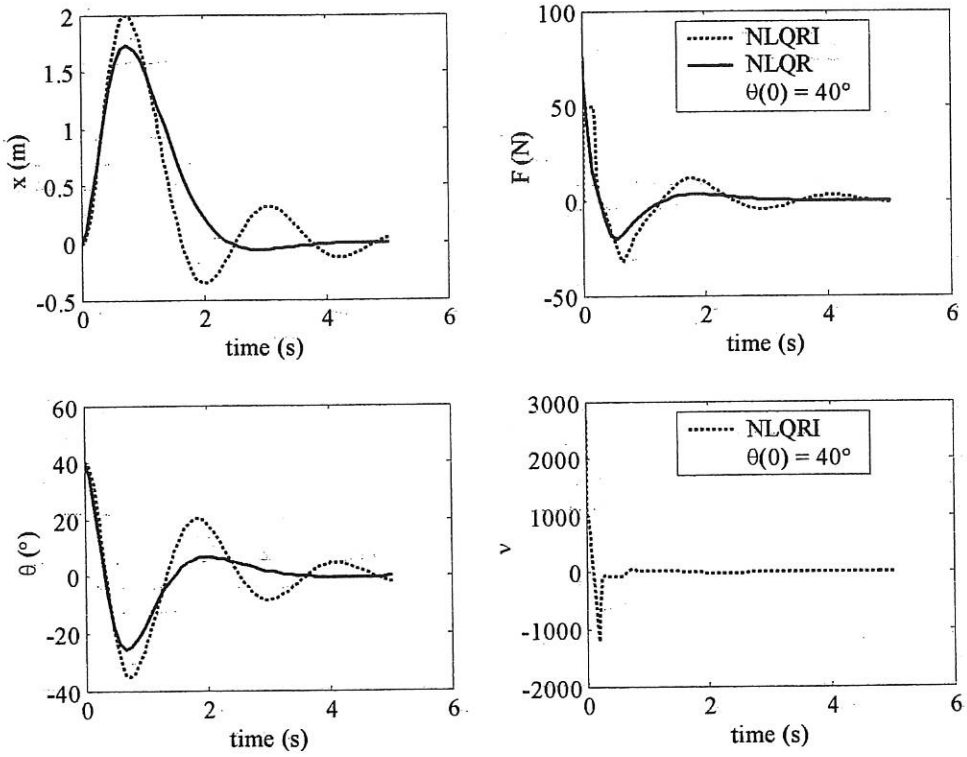

Figure 8: Comparison of state and control evolution uder non-linear (NLQR) and non-linear integral (NLQRI) control for an initial angle of 40° .

It is clear from Figure 8 that the demanded force (top right) now never exceeds what can be delivered by the hardware, the trade-off being larger excursions and greater total variation in the angular and linear displacements (we do not show the corresponding velocities for reasons of space). It is possible to achieve a closer match to the original problem by judicious tuning of the parameters R and R_a but this is not considered further.

5 Conclusions

A new method for the design and synthesis of near-optimal, non-linear control laws is examined, based on a generalisation of LQ optimal control theory. The method is

simple to apply and affords greater design flexibility, via state-dependent weighting than conventional approaches (including receding horizon control). The resulting regulator can, in principle, be implemented in real-time owing to the causal nature of the required computations. However, the need to solve an ARE every time-step is burdensome and a number of algorithms that would permit a parallel solution are discussed. Notwithstanding the technical difficulties of real-time implementation, it is safe to say that the essential problem of real-time, non-linear optimal control, *viz.* the acausal nature of the solution, can be overcome by this approach and a way has been opened towards real-time optimal quadratic regulation.

A common benchmark problem – stabilisation of a simple inverted pendulum – is used to illustrate the method. The application throws up some interesting points and proves an exacting task. In particular, the requirement for local stabilizability breaks down at a number of points in state space, points that are necessarily visited for large initial angular displacements. While these points are isolated they induce neighbourhoods of poor stabilizability that cause numerical ill-conditioning in the ARE solver. Nonetheless, the NLQR algorithm manages to recover in these situations, albeit with an impractically large force requirement. Clearly it would be important to ensure that such problems are avoided in any real application and so one should think of the method as widening the range of operation rather than as providing a global solution. A method to overcome non-linearity in the control signal is presented and demonstrated for a saturation non-linearity in the target problem.

Finally, for practical implementation, the question of state estimation arises as an adjunct to the present work. The Extended Kalman Filter is a natural candidate here but work has been carried out that exploits the duality of regulation and estimation problems [13] and which provides a non-linear dual estimator that fits the present framework. Future work will incorporate both regulator and estimator in the control of the physical device referred to above.

Reference List

[1] Banks,SP and Mhana,KJ, Optimal control and stabilization for non-linear systems. *IMA Journal on Mathematical Control and Information*, 1992; 9: 179-196.

[2] McCaffrey,D and Banks,SP, Lagrangian manifolds and asymptotically optimal stabilizing feedback control. *Systems & Control Letters*, 2001; 43: 219-224.

[3] Mracek,CP and Cloutier,JR, Control designs for the non-linear bench-mark problem via the state-dependent Riccati equation method. *Journal of Robust and Non-linear Control*, 1998; 8: 401-433.

[4] Harrison,RF and Banks,SP, A new non-linear design method for active vehicle suspension systems. Proceedings of the Intelligent Components for Vehicles, ICV'98, Sevilla, Spain, 1998, pp 417-422.

[5] Harrison,RF and Banks,SP, Towards real-time, non-linear quadratic optimal regulation. Proceedings of the Algorithms and Architectures for Real-Time Control, AARTC'98, Cancun, Mexico, 1998, pp 107-112.

[6] Gardiner,JD and Laub,AJ, A generalisation of the matrix-sign-function for algebraic Riccati equations. *International Journal of Control*, 1986; 44: 823-832.

[7] Gardiner,JD and Laub,AJ, Parallel algorithms for algebraic Riccati equations. *International Journal of Control*, 1991; 54: 1317-1333.

[8] Friedland,B, Control system design: an introduction to state-space methods. McGraw-Hill Book Co., New York, 1987.

[9] Laub,AJ, A Schur method for solving algebraic Riccati equations. *IEEE Transactions On Automatic Control*, 1979; 24: 913-921.

[10] Roberts,JD, Linear model reduction and solution of the algebraic Riccati equation by use of the sign function. *International Journal of Control*, 1980; 32: 677-687.

[11] Balzer,LA, Accelerated convergence of the matrix sign function method of solving Lyapunov, Riccati and other matrix equations. *International Journal of Control*, 1980; 32: 1057-1078.

[12] Bunse-Gerstner,A and Faßbender,H, A Jacobi-like method for solving algebraic Riccati equations on parallel computers. *IEEE Transactions On Automatic Control*, 1997; 42: 1071-1084.

[13] McCaffrey,D, Harrison,RF, and Banks,SP, Asymptotically optimal nonlinear filtering. Research Report 734, The University of Sheffield, Sheffield, 1998.

[14] van der Schaft,AJ, On a state space approach to H_∞ non-linear control. *Systems & Control Letters*, 1991; 16: 1-8.

Appendix A

A brief outline of the NLQR derivation [2] follows. Consider the following infinite-time, optimal control problem

$$J(\mathbf{x}_0) = \inf_{\mathbf{u}\in L_2(0,\infty)} \frac{1}{2}\int_0^\infty \mathbf{x}^t Q(\mathbf{x})\mathbf{x} + \mathbf{u}^t R(\mathbf{x})\mathbf{u}\, dt \tag{1.1}$$

subject to the system dynamics $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + B(\mathbf{x})\mathbf{u};\ \mathbf{x}\in\mathbb{R}^n; \mathbf{u}\in\mathbb{R}^m; \mathbf{f}(\mathbf{0}) = \mathbf{0}$. The functions f, B, $Q > 0\ \forall\mathbf{x}$, and $R > 0\ \forall\mathbf{x}$ are analytic and of appropriate dimension. The cost functional, $J(\mathbf{x})$, is a solution of the Hamiton-Jacobi-Bellman equation

$$\max_{\mathbf{u}}\left\{-\nabla_{\mathbf{x}} J(\mathbf{x})\left(\mathbf{f}(\mathbf{x}) + B(\mathbf{x})\mathbf{u}\right) - \frac{1}{2}\mathbf{x}^t Q(\mathbf{x})\mathbf{x} - \frac{1}{2}\mathbf{u}^t R(\mathbf{x})\mathbf{u}\right\} = 0 \tag{1.2}$$

where $\nabla_x \triangleq \left[\partial/\partial x_1 \quad \partial/\partial x_2 \quad \cdots \quad \partial/\partial x_n\right]$. If $\left(\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\big|_{\mathbf{x}=\mathbf{0}}, B(\mathbf{0}), Q^{1/2}(\mathbf{0})\right)$ is stabilizable and detectable a smooth solution, $J(\mathbf{x})$, of (1.2) is known to exist [14]. Let $\mathbf{f}(\mathbf{x}) = A(\mathbf{x})\mathbf{x}$ such that $\lim_{\mathbf{x}\to\mathbf{0}} A(\mathbf{x}) \to \nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x})\big|_{\mathbf{x}=\mathbf{0}}$ then, in the region where $J(\mathbf{x})$ is a smooth solution to (1.2), the maximum is attained when $u = -R^{-1}(\mathbf{x})B^t(\mathbf{x})\nabla_{\mathbf{x}}^t J(\mathbf{x})$ and $J(\mathbf{x})$ is the solution of

$$-\nabla_{\mathbf{x}} J(\mathbf{x})\mathbf{f}(\mathbf{x}) + \frac{1}{2}\nabla_{\mathbf{x}} J(\mathbf{x})B(\mathbf{x})R^{-1}(\mathbf{x})B^t(\mathbf{x})\nabla_{\mathbf{x}}^t J(\mathbf{x}) - \frac{1}{2}\mathbf{x}^t Q(\mathbf{x})\mathbf{x} = 0 \tag{1.3}$$

by substitution.

Under the "linearisation", $f(\mathbf{x}) = A(\mathbf{x})\mathbf{x}$, $\nabla_{\mathbf{x}}\mathbf{f}(\mathbf{x}) = A(\mathbf{x})$ and noting that since $\nabla_{\mathbf{x}}^t J(\mathbf{x})\big|_{\mathbf{x}=\mathbf{0}} = \mathbf{0}$ [14], we can write $\nabla_{\mathbf{x}}^t J(\mathbf{x}) = P(\mathbf{x})\mathbf{x}$. Substituting into (1.3) and noting that $2\mathbf{x}^t P(\mathbf{x})A(\mathbf{x})\mathbf{x} = \mathbf{x}^t\left(P(\mathbf{x})A(\mathbf{x}) + A^t(\mathbf{x})P(\mathbf{x})\right)\mathbf{x}$ we get

$$\mathbf{x}^t\left(-P^t(\mathbf{x})A(\mathbf{x}) - A^t(\mathbf{x})P(\mathbf{x}) + P^t(\mathbf{x})B(\mathbf{x})R^{-1}(\mathbf{x})B(\mathbf{x})P - Q(\mathbf{x})\right)\mathbf{x} = 0 \tag{1.4}$$

The approximation to the solution ignores the fact that $P(\mathbf{x})\mathbf{x}$ should equal the gradient of some function and, instead, assumes that $P(\mathbf{x})$ is symmetrical. Then, at each $\mathbf{x}$, the algorithm simply finds the positive semi-definite solution to the state-dependent ARE

$$P(\mathbf{x})A(\mathbf{x}) + A^t(\mathbf{x})P(\mathbf{x}) - P(\mathbf{x})B(\mathbf{x})R^{-1}(\mathbf{x})B(\mathbf{x})P + Q(\mathbf{x}) = 0 \tag{1.5}$$

and applies, at that $\mathbf{x}$, the control $u = -R^{-1}(\mathbf{x})B^t(\mathbf{x})P(\mathbf{x})\mathbf{x}$.

It should be noted that while smoothness in the solution is used in the derivation, it is not a requirement for application.

$\infty\ H_\infty H_\infty$