Prof K Glover

Engineering Tripos Part IIB, Module 4F3, OPTIMAL AND PREDICTIVE CONTROL SAMPLE SOLUTIONS TO EXAM MAY 2012

1. **Solution:**

(a) i. Bookwork to derive:

$$V(x,k) = \min_{u \in U} \{V(f(x,k), k+1) + c(x,u)\}\,; \quad V(x,h) = J_h(x).$$

ii. The advantage of this approach is that the minimisation over all $u's$ is replaced by a minimisation over one u at a time, but over all x. Sometimes the solution of each minimisation can be analytically found (e.g. in a quadratic case) and then a backward recursion is possible to find $V(x,k)$. In more complex cases when it might be tempting to quantise the state space the 'curse of dimensionality' comes into play and this approach is only feasible for a small state dimension.

(b) i. The H-J-B equation is:

$$-\frac{\partial V}{\partial t} = \max_{u(t)} \left\{ \sqrt{u(t)} + \frac{\partial V}{\partial x}(\alpha x(t) - u(t)) \right\}$$

[note the max in this equation because we are maximising the utility rather than minimising the cost - this can be verified by minimising $-V$ and following through the sign changes]

Differentiating with respect to the scalar, $u(t)$, to find the max gives,

$$\frac{1}{2\sqrt{u(t)}} - \frac{\partial V}{\partial x} = 0$$

and hence the optimal u is

$$u^*(t) = \frac{1}{4}\left(\frac{\partial V}{\partial x}\right)^{-2}$$

and

$$\begin{aligned} -\frac{\partial V}{\partial t} &= \frac{1}{2}\left(\frac{\partial V}{\partial x}\right)^{-1} + \frac{\partial V}{\partial x}\left(\alpha x(t) - \frac{1}{4}\left(\frac{\partial V}{\partial x}\right)^{-2}\right) \\ &= \frac{1}{4}\left(\frac{\partial V}{\partial x}\right)^{-1} + \alpha x(t)\frac{\partial V}{\partial x} \end{aligned}$$

The boundary condition will be $V(x,T) = 0$ since there is no utility in any residual funds. *[very few candidates actually solved for u at this point]*

ii. If we assume, $V(x,t) = \sqrt{w(t)}\sqrt{x}$ then

$$\frac{\partial V}{\partial t} = \frac{1}{2}(w(t))^{-1/2}\sqrt{x}\dot{w}(t)$$

and

$$\frac{\partial V}{\partial x} = \frac{1}{2}(x)^{-1/2}\sqrt{w(t)}$$

and substituting into the H-J-B equation gives

$$-\frac{1}{2}(w(t))^{-1/2}\sqrt{x}\dot{w}(t) = \frac{1}{4}2\sqrt{x}(w(t))^{-1/2} + \alpha x\frac{1}{2}(x)^{-1/2}\sqrt{w(t)}$$

The common factor $\sqrt{x}$ can be cancelled and we are left with,

$$\dot{w}(t) = -\alpha w(t) - 1, \quad w(T) = 0.$$

iii. The solution for $w(t)$ is in the form $A + Be^{-\alpha t}$ with $A + Be^{-\alpha T} = 0$ so $w(t) = B(-e^{-\alpha T} + e^{-\alpha t})$ and $\dot{w} = -B\alpha e^{-\alpha t} = -\alpha B(-e^{-\alpha T} + e^{-\alpha t}) - 1$. Hence $B = e^{\alpha T}/\alpha$.

$$w(t) = (e^{\alpha(T-t)} - 1)/\alpha$$

and

$$\begin{aligned} u^*(t) &= \frac{1}{4}\left(\frac{\partial V}{\partial x}\right)^{-2} \\ &= x(t)/w(t) \\ &= \frac{\alpha}{(e^{\alpha(T-t)} - 1)}x(t) \end{aligned}$$

2. **Solution:** *[very few attempts at this question]*

(a)

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 w(t) + B_2\hat{u}(t) \\ z(t) &= C_1 x(t) + D_{12}u(t) \end{aligned}$$

$$\hat{u} = Ru + Lx, \quad \hat{z} = Mz$$

Substituting for u and z gives

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 w(t) + B_2\left(R^{-1}\hat{u}(t) - R^{-1}Lx(t)\right) \\ &= (A - B_2 R^{-1}L)x + B_1 w + B_2 R^{-1}\hat{u} \\ \hat{z}(t) &= M\left[C_1 x(t) + D_{12}\left(R^{-1}\hat{u}(t) - R^{-1}Lx(t)\right)\right] \\ &= M\left[(C_1 - D_{12}R^{-1}L)x + D_{12}R^{-1}\hat{u}(t)\right] \end{aligned}$$

(b) If $M^T M = I$ then $\|\hat{z}\|_2 = \|z\|_2$ and hence the two norms will be the same.

(c) We require

$$\begin{bmatrix} \hat{C}_1 & \hat{D}_{12} \end{bmatrix} = \begin{bmatrix} \hat{C}_{11} & 0 \\ 0 & I \end{bmatrix}$$

$$\begin{aligned} \hat{D}_{12} &= MD_{12}R^{-1} = MU_1\Sigma V^T R^{-1} \\ &= \begin{bmatrix} 0 \\ I \end{bmatrix} \text{ if } \mathrm{R} = \Sigma \mathrm{V}^{\mathrm{T}},\ \mathrm{M} = \begin{bmatrix} U_2^T \\ U_1^T \end{bmatrix} \end{aligned}$$

Also

$$\begin{aligned} \hat{C}_1 &= M(C_1 - D_{12}R^{-1}L) = \begin{bmatrix} U_2^T \\ U_1^T \end{bmatrix} (C_1 - U_1 L) \\ &= \begin{bmatrix} U_2^T C_1 \\ U_1^T C_1 - L \end{bmatrix} \\ \text{giving L} &= U_1^T C_1 \text{ and } \hat{\mathrm{C}}_{11} = \mathrm{U}_2^{\mathrm{T}} \mathrm{C}_1 \end{aligned}$$

(d) The solution to the revised problem can now be taken from the ARE in the data sheet:

$$\hat{A}^T X + X\hat{A} + \hat{C}_{11}^T \hat{C}_{11} - X\hat{B}_2\hat{B}_2^T X + \gamma^{-2} X\hat{B}_1\hat{B}_1^T X = 0$$

where the solution X is such that $(\hat{A} - \hat{B}_2\hat{B}_2^T X + \gamma^{-2}\hat{B}_1\hat{B}_1^T X)$ is stable and a satisfactory state feedback is given by $\hat{u}(t) = -\hat{B}_2^T X x(t)$. The controller for the original problem is now

$$\begin{aligned} u(t) &= R^{-1}\left(\hat{u}(t) - Lx(t)\right) \\ &= R^{-1}\left(-\hat{B}_2^T X x(t) - Lx(t)\right) = R^{-1}\left(-\hat{B}_2^T X - L\right) x(t) \\ &= -R^{-1}\left(R^{T^{-1}} B_2^T X + L\right) x(t) \end{aligned}$$

3. **Solution:**

(a) i. The receding horizon operates as follows:

- Generate an optimal and feasible open-loop sequence of inputs over a finite prediction horizon based on current state measurements or estimates;
- Apply only the first element of the sequence to the plant. Discard the rest of the sequence;
- Take a new measurement at the next sampling instant. Repeat.

ii. Predictive control might be advantageous when:

- hard (possibly non-convex) operational or physical constraints must be enforced; or
- the plant is MIMO with a high degree of cross-coupling or actuator redundancy; or
- the plant is nonlinear, and optimising over an infinite-horizon is not computationally feasible; or
- the control objective is clear, but it is important to adapt to changing plant model.

Examples might include:

- Minimising variance in paper thickness whilst enforcing constraints on tank levels.
- Manoeuvring a spacecraft with finite thrust capability along fuel-optimal trajectory, whilst avoiding collisions with obstacles.

(b)

$$\begin{aligned} x_0 &= x_0 \\ x_1 &= Ax_0 + Bu_0 \\ x_2 &= A^2x_0 + ABu_0 + Bu_1 \\ x_3 &= A^3x_0 + A^2Bu_0 + ABu_1 + Bu_2 \end{aligned}$$

Therefore:

$$\mathbf{x} = \underbrace{\begin{bmatrix} I \\ A \\ A^2 \\ A^3 \end{bmatrix}}_{\Phi} x_0 + \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ B & 0 & 0 \\ AB & B & 0 \\ A^2B & AB & B \end{bmatrix}}_{\Gamma} \mathbf{u}.$$

(c) i. Sufficient conditions to ensure convexity are:

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \geq 0, \quad R > 0, \quad P \geq 0.$$

[very few produced a correct condition on S]

ii.

$$\begin{aligned} V(\mathbf{x}, \mathbf{u}) &= x_3^T P x_3 + \sum_{k=0}^{2} \left(x_k^T Q x_k + u_k^T R u_k + x_k^T S u_k + u_k^T S^T x_k \right) \\ &= \mathbf{x}^T \mathcal{Q}_e \mathbf{x} + \mathbf{u}^T \mathcal{R}_e \mathbf{u} + \mathbf{x}^T \mathcal{S}_e \mathbf{u} + \mathbf{u}^T \mathcal{S}_e^T \mathbf{x} \\ V(x_0, \mathbf{u}) &= (\Phi x_0 + \Gamma \mathbf{u})^T \mathcal{Q}_e (\Phi x_0 + \Gamma \mathbf{u}) + \mathbf{u}^T \mathcal{R}_e \mathbf{u} \\ &\quad + (\Phi x_0 + \Gamma \mathbf{u})^T \mathcal{S}_e \mathbf{u} + \mathbf{u}^T \mathcal{S}_e^T (\Phi x_0 + \Gamma \mathbf{u}) \\ &= x_0^T \Phi^T \mathcal{Q}_e \Phi x_0 + 2\mathbf{u}^T \left(\Gamma^T \mathcal{Q}_e + \mathcal{S}_e^T \right) \Phi x_0 \\ &\quad + \mathbf{u}^T \left(\Gamma^T \mathcal{Q}_e \Gamma + \Gamma^T \mathcal{S}_e + \mathcal{S}_e^T \Gamma + \mathcal{R}_e \right) \mathbf{u} \end{aligned}$$

Terms of $V(x_0, \mathbf{u})$ which do not include the decision variable $\mathbf{u}$ can be neglected as they do not affect the minimiser $\mathbf{u}^*$, so the QP in standard form is:

$$\min_{\mathbf{u}} \frac{1}{2}\mathbf{u}^T \left(\Gamma^T \mathcal{Q}_e \Gamma + \Gamma^T \mathcal{S}_e + \mathcal{S}_e^T \Gamma + \mathcal{R}_e\right) \mathbf{u} + x_0^T \Phi^T \left(\mathcal{S}_e + \mathcal{Q}_e \Gamma\right) \mathbf{u}$$

Subject to:

$$J\mathbf{u} \leq c + Wx_0$$
$$A^3 x_0 + \begin{bmatrix} A^2B & AB & B \end{bmatrix} \mathbf{u} = 0.$$

i.e.

$$\begin{bmatrix} 0 & 0 & 0 & I \end{bmatrix} [\Phi x_0 + \Gamma \mathbf{u}] = 0$$

(d)

- Solving a QP is more computationally demanding than evaluating a matrix-vector multiplication, but convex QPs can be solved "efficiently".
- Computation time might introduce a delay in the loop, to which the controller must be robust
- The numerical precision required to solve the QP might be higher than that required for evaluation of a matrix-vector multiplication
- MPC historically applied to slow processes where computation time was not an issue
- For faster processes, a sufficiently fast computer is needed (alternatively implementation on FPGAs or other dedicated hardware is a current research topic)
- Shorter prediction horizons can reduce computational demands
- Use move-blocking to reduce the number of decision variables
- Can explicitly include delays in the prediction model to allow longer for computation
- (Explicit MPC uses multi-parametric programming to precompute the solution as a piecewise-affine function of the state)

4. **Solution:**

(a) i. The constraint admissible set, $S \subseteq \mathbb{R}^{n_x}$ is the set of states for which the state and input under the specified control law satisfy the specified constraints. For constraints $Z \subset \mathbb{R}^{n_x} \times \mathbb{R}^{n_u}$, S is constraint admissible if:

$$(x, Kx) \in Z \,\forall x \in S.$$

For the above system, there are only input constraints, but these are a function of the state. Therefore, the constraint admissible set is:

$$-5 \leq K\hat{x}_k \leq 5.$$

ii. The stabilising K that meets the specification is $K = -\frac{5}{3}$. *[many solutions forgot about the stability requirement]*

$(A + BK) = (1.2 - 5/3) \approx -0.47$. Therefore for a given initial condition x_0:

$$\begin{aligned} x_n &= (A + BK)^n x_0 \\ &= (-0.47)^n x_0 \\ |x_n| &= 0.47^n |x_0|. \end{aligned}$$

Therefore the magnitude of x_n is strictly decreasing, so if $|x_0| \leq 3$ then $|x_n| \leq 3$, and $|u_n| \leq 3 \times \frac{5}{3} = 5, \forall n \in \{0, \ldots, \infty\}$.

(b) i. The sequences $\mathbf{u}^* = \begin{bmatrix} u_0^* & u_1^* & \ldots & u_{N-1}^* \end{bmatrix}^T$ and $\mathbf{x}^* = \begin{bmatrix} x_0 & x_1^* & \ldots & x_N^* \end{bmatrix}^T$ are optimal and feasible at $k = 0$.

After the first control action has been applied, a candidate solution for the subsequent optimisation problem is: $\mathbf{u}^+ = \begin{bmatrix} u_1^* & u_2^* & \ldots & u_{N-1}^* & Kx_N^* \end{bmatrix}^T$ and $\mathbf{x}^+ = \begin{bmatrix} x_1 & x_2^* & \ldots & x_N^* & (1.2 + K)x_N^* \end{bmatrix}^T$.

The beginning of both sequences is feasible by construction due to the feasiblity of the previous optimisation. The value appended to the input sequence, Kx_N^* is feasible because the terminal constraint guarantees the controller $u = Kx$ is constraint admissible when $k = N$. The terminal set is also invariant, so $(1.2 + K)x_N$ is also constraint admissible.

The cost associated with these candidate sequences is:

$$\begin{aligned} V(\mathbf{x}^+, \mathbf{u}^+) = V^*(x_0) & \underbrace{-qx_0^2 - ru_0^2 - px_N^2}_{\text{Terms removed}} \\ & \underbrace{+qx_N^2 + K^2 r x_N^2 + (1.2 + K)^2 p x_N^2}_{\text{Terms added}}. \end{aligned}$$

Hence the optimal cost for the new problem must be less than or equal to the cost of the candidate feasible solution:

$$V^*(x_1) \leq V(\mathbf{x}^+, \mathbf{u}^+).$$

ii.

$$(1.2 + K)^2 p - p < -(q + K^2 r)$$

The receding horizon control law stabilises the plant to the origin asymptotically.

(c) A stabilising controller does not exist if $|x_k| \geq 25$ since in this case $|x_{k+1}| = |1.2x_k + u_k| \geq 1.2 \times 25 - 5 = 25$, so that $|x_n|$ will always be ≥ 25 for all future n.

If this is a reasonable state to expect, then the plant must be modified (e.g. adding redundant actuators, or replacing the actuators with some with a larger range), or the open-loop dynamics must be modified.

KG 2012

Examiner's comments:

Q1.

Answered by most candidates. Part (a) was straightforward and answered well but part (b) where they were required to set up and solve the Hamilton-Jacobi-Belman pde for an example they had not seen before, gave significant difficulty for most candidates.

Q2.

A most unpopular question with relatively simple matrix manipulations but they required a deeper understanding of the material.

Q3.

A popular and straightforward question on MPC with most attempts pretty complete and understanding the main points. The discussion parts were more varied in the length and depth of the answers. Fortunately minor manipulation slips had minimal consequences. One point that was almost completely missed was conditions on the cross term, S, to make the problem convex, with most candidates requiring S>0 in spite of it not even being square!

Q4.

Attempted by all candidates. The manipulations were done pretty well but the precision of various of various statements and definitions were variable.

2012 paper 4F3 Optimal and Predictive Control - Answers

1.(b)(i) $-\frac{\partial V}{\partial t} = \frac{1}{4}\left(\frac{\partial V}{\partial x}\right)^{-1} + \alpha x(t)\frac{\partial V}{\partial x}$

(b)(ii) $\dot{w}(t) = -\alpha w(t) - 1, \quad w(T) = 0.$

(b)(iii) $u^*(t) = \frac{\alpha}{(e^{\alpha(T-t)}-1)}x(t)$

2. (a) $\hat{A} = (A - B_2R^{-1}L), \hat{B}_1 = B_1, \hat{B}_2 = B_2R^{-1},$
$\hat{C}_1 = M(C_1 - D_{12}R^{-1}L), \hat{D}_{12} = MD_{12}R^{-1}.$

(b) $M^TM = I.$

(c) $R = \Sigma V,\ L = U_1^TC_1,\ M = \begin{bmatrix} U_2^T \\ U_1^T \end{bmatrix}.$

(d) The ARE is: $\hat{A}^TX + X\hat{A} + \hat{C}_{11}^T\hat{C}_{11} - X\hat{B}_2\hat{B}_2^TX + \gamma^{-2}X\hat{B}_1\hat{B}_1^TX = 0$, where $\hat{C}_{11} = U_2^TC_1$. $u(t) = -R^{-1}\left(R^{T^{-1}}B_2^TX + L\right)x(t)$.

3.(c)(i) $\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \geq 0, \quad R > 0, \quad P \geq 0.$

(c)(ii)

$$\min_{\mathbf{u}} \frac{1}{2}\mathbf{u}^T\left(\Gamma^T\mathcal{Q}_e\Gamma + \Gamma^T\mathcal{S}_e + \mathcal{S}_e^T\Gamma + \mathcal{R}_e\right)\mathbf{u} + x_0^T\Phi^T\left(\mathcal{S}_e + \mathcal{Q}_e\Gamma\right)\mathbf{u}$$

Subject to:

$$J\mathbf{u} \leq c + Wx_0$$
$$\begin{bmatrix} 0 & 0 & 0 & I \end{bmatrix}[\Phi x_0 + \Gamma\mathbf{u}] = 0$$

4.(a)(ii) $K = -\frac{5}{3}$.

(b)(ii) $(1.2 + K)^2p - p < -(q + K^2r)$

(c) Cannot be stabilised if $|x_k| \geq 25$.