En introduktion till lärande reglering

Förstärkningsinlärning eller hur man tar fram en optimal tillståndsåterkoppling utan en modell av systemet

Martin Enqvist
2020-03-03

Inledning

En viktig grundmekanism i människors och djurs lärande verkar vara att testa olika handlingar och att anpassa dem baserat på den belöning eller bestraffning som handlingarna resulterar i. Några exempel på tidiga studier av denna princip för lärande är till exempel den amerikanske psykologen Edward Thorndikes (1874-1949) teorier och experiment om försök och misstag (eng. trial and error) och den ryske vetenskapsmannen och nobelpristagaren Ivan Pavlovs (1849-1936) experiment med hundar. Idag pratar med vanligen om *operant* eller *instrumentell betingning* när man beskriver lärande baserat på belöning eller bestraffning och ett vardagligt exempel är att många lär hundar att sitta genom att belöna det beteendet med en godisbit. Godisbiten är en så kallad *positiv förstärkning* av beteendet och i andra sammanhang kan det istället på samma sätt finnas *negativa förstärkningar*.

Inlärningsprincipen operant betingning med positiv eller negativ förstärkning har en motsvarighet inom *maskininlärning*, där målet är att få datorer att lära sig att lösa en uppgift utan att de på förhand har explicita regler för hur det ska göras. Inom maskininlärning är *förstärkningsinlärning* (eng. reinforcement learning) det vedertagna namnet på metoder som löser optimeringsproblem som involverar en agent eller aktör som interagerar med en omgivning och modifierar sina handlingar baserat på den respons som handlingarna ger.

Det här är en intressant princip inom många ingenjörsområden, bland annat inom *reglerteknik*, som handlar om att styra system som ofta är dynamiska genom att göra automatiska justeringar av systemets insignaler. Målet med styrningen är att det kompletta systemet, som består av både det ursprungliga systemet och regleralgoritmen (regulatorn), fungerar bättre än det ursprungliga systemet trots förekomsten av störningar i systemet och trots att systemets egenskaper delvis är okända. I många tillämpningar använder man en fix regleralgoritm för att styra systemet, till exempel för att man redan har eller kan skatta en relativt god modell av systemet och därför relativt enkelt kan bestämma en lämplig reglerstrategi eller för att man har ett behov av att kunna validera regleralgoritmens beteenden i olika situationer för att kunna ge säkerhetsgarantier. I vissa tillämpningar kan det dock vara svårt att ta fram en modell, till exempel för att systemet inte existerar ännu eller för att systemets egenskaper kan förändras på svårförutsägbara sätt, och här kan förstärkningsinlärning vara ett intressant alternativ.

I reglertekniska tillämpningar kan man tänka på aktören i beskrivningen av förstärkningsinlärning som en extra lärande algoritm i regulatorn och handlingarna som den reglerstrategi som används i regulatorn (funktionen som används för att beräkna styrsignalen från mätningar). Reglertekniska algoritmer som bygger på den här idén har studerats och använts sedan 1950-talet, oftast under namnet *adaptiv reglering*, och det finns också en koppling till det delområde som kallas *optimal styrning*. Förstärkningsinlärning för reglertekniska ändamål kan ses som ett exempel på adaptiv optimal styrning. Vi ska här studera en specifik metod för förstärkningsinlärning som ger ett modellfritt alternativ till en klassisk modellbaserad reglerteknisk metod som kallas *linjärkvadratisk reglering*. Framställningen här bygger främst på artiklarna [1] och [3].

Linjärkvadratisk reglering i diskret tid

Ett tidsdiskret linjärt tidsinvariant system kan skrivas på tillståndsform som

$$x_{k+1} = Ax_k + Bu_k, \tag{1}$$

där x_k är en vektor som innehåller systemets tillståndsvariabler, u_k är systemets insignal och A och B är matriser. Antag att vår regleruppgift är att styra tillbaka tillstånden x_k till origo från ett nollskilt

starttillstånd och att vi kan mäta hela tillståndsvektorn. Ett populärt sätt att hitta en lämplig styrsignal är att definiera reglerproblemet som att man ska hitta u_i, $i = k, k+1, \ldots$, så att målfunktionen

$$J_k = \frac{1}{2}\sum_{i=k}^{\infty} x_i^T Q x_i + u_i^T R u_i \tag{2}$$

minimeras, något som brukar kallas linjärkvadratisk reglering eftersom systemet är linjärt och målfunktionen är kvadratisk. Matriserna Q och R i (2) är viktmatriser som man kan välja för att prioritera snabb reglering av tillstånden (genom att sätta stora värden i Q) eller små styrsignaler (med stora värden i R). Det är värt att notera att en samtidig skalning av Q och R med samma konstant inte förändrar styrsignalen eftersom det är förhållandet mellan Q och R som är intressant. Om styrsignalen är skalär väljer man därför ofta att sätta $R = 1$ för att inte ha onödigt många parametrar att välja. Det här är den tidsdiskreta motsvarigheten till den tidskontinuerliga linjärkvadratiska reglering som, för just fallet $R = 1$, beskrivs i avsnitt 9.3 i kursboken [2].

Antag att vi befinner oss i en godtycklig tidpunkt k och har ett förslag på en linjär tillståndsåterkoppling

$$u_i = -Lx_i \tag{3}$$

och använder den på systemet (1) vid den aktuella och alla framtida tidpunkter. Detta ger oss ett slutet system

$$x_{i+1} = (A - BL)x_i \tag{4}$$

och genom att använda denna ekvation flera gånger kan vi uttrycka ett godtyckligt framtida tillstånd x_{k+N} som

$$x_{k+N} = (A - BL)^N x_k. \tag{5}$$

Insättning av (3) och (5) i (2) ger oss nu

$$J_{L,k} = x_k^T \underbrace{\frac{1}{2}\sum_{i=0}^{\infty} (A - BL)^{i,T}(Q + L^T R L)(A - BL)^i}_{=:P_L} x_k = \frac{1}{2} x_k^T P_L x_k =: V_L(x_k), \tag{6}$$

det vill säga målfunktionen är alltså en kvadratisk funktion $V_L(x_k)$ av det aktuella tillståndet x_k. Varje tillståndsåterkoppling som resulterar i ett asymptotiskt stabilt slutet system (4) kommer att ge ett ändligt värde på målfunktionen och vi kan till exempel jämföra prestandan hos (3) med den hos en annan tillståndsåterkoppling

$$u_i = -\tilde{L}x_i \tag{7}$$

genom att jämföra $V_L(x_k)$ med $V_{\tilde{L}}(x_k)$.

Det visar sig att det optimala värdet på styrsignalen (i meningen att den minimerar målfunktionen (2)) ges av tillståndsåterkopplingen

$$u_k = -L_{LQ}x_k = -(B^T P B + R)^{-1} B^T P A x_k \tag{8}$$

där P är den positiva semidefinita lösningen till den tidsdiskreta algebraiska riccatiekvationen

$$A^T P A - P + Q - A^T P B (B^T P B + R)^{-1} B^T P A = 0. \tag{9}$$

En modell (matriserna A och B) krävs dock för att kunna beräkna återkopplingen på det här sättet. I nästa avsnitt ska vi diskutera hur vi kan komma fram till samma reglerstrategi som i (8) med hjälp av förstärkningsinlärning.

Förstärkningsinlärning

Givet en viss reglerstrategi

$$u_i = -Lx_i$$

så kan vi skriva om uttrycket för målfunktionen som

$$\begin{aligned} V_L(x_k) &= \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2}\sum_{i=k+1}^{\infty} x_i^T Q x_i + u_i^T R u_i \\ &= \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k) + V_L(x_{k+1}). \end{aligned}$$

Definiera nu en ny funktion $\mathcal{Q}(x_k, u_k)$ som

$$\mathcal{Q}(x_k, u_k) = \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k) + V_L(x_{k+1}) \tag{10}$$

där u_k är en godtycklig styrsignal vid tiden k. Funktionen $\mathcal{Q}(x_k, u_k)$ beskriver alltså kostnaden för att starta i tillståndet x_k och använda en godtycklig styrsignal u_k vid tiden k för att därefter använda styrsignalerna $u_{k+j} = -Lx_{k+j}$ vid alla efterföljande tidpunkter. Ett möjligt styrsignalval vid tiden k är förstås $u_k = -Lx_k$ och med det valet återfår vi vår målfunktion $V_L(x_k)$, det vill säga

$$V_L(x_k) = \mathcal{Q}(x_k, -Lx_k). \tag{11}$$

Det intressanta med funktionen $\mathcal{Q}(x_k, u_k)$ är dock att vi kan använda den till att utvärdera olika modifieringar av en föreslagen reglerstrategi. Om vi känner matriserna A och B i tillståndsbeskrivningen för systemet kan vi skriva om $\mathcal{Q}(x_k, u_k)$ som

$$\begin{aligned} \mathcal{Q}(x_k, u_k) &= \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2}(Ax_k + Bu_k)^T P_L (Ax_k + Bu_k) \\ &= \frac{1}{2}\begin{pmatrix} x_k \\ u_k \end{pmatrix}^T \begin{pmatrix} A^T P_L A + Q & A^T P_L B \\ B^T P_L A & B^T P_L B + R \end{pmatrix} \begin{pmatrix} x_k \\ u_k \end{pmatrix} \end{aligned}$$

Givet x_k så är $\mathcal{Q}(x_k, u_k)$ en kvadratisk funktion i u_k och vi kan beräkna den styrsignal som minimerar $\mathcal{Q}(x_k, u_k)$ genom att lösa ekvationen

$$\frac{\partial \mathcal{Q}(x_k, u_k)}{\partial u_k} = 0$$

vilket ger oss

$$B^T P_L A x_k + (B^T P_L B + R) u_k = 0 \quad \Rightarrow \quad u_k = -(B^T P_L B + R)^{-1} B^T P_L A x_k \tag{12}$$

förutsatt att $B^T P_L B + R$ är inverterbar.

Även om vi inte känner matriserna A och B så kan vi ibland ta fram ett allmänt uttryck för funktionen $\mathcal{Q}(x_k, u_k)$ och minimera det med avseende på u_k. Med

$$\mathcal{Q}(x_k, u_k) = \frac{1}{2}\begin{pmatrix} x_k \\ u_k \end{pmatrix}^T \begin{pmatrix} S_{xx} & S_{xu} \\ S_{ux} & S_{uu} \end{pmatrix} \begin{pmatrix} x_k \\ u_k \end{pmatrix}, \tag{13}$$

där S_{xx} och S_{uu} är symmetriska matriser och $S_{xu}^T = S_{ux}$, kan vi återigen lösa ekvationen

$$\frac{\partial \mathcal{Q}(x_k, u_k)}{\partial u_k} = 0$$

vilket nu ger oss

$$S_{ux} x_k + S_{uu} u_k = 0 \quad \Rightarrow \quad u_k = -\underbrace{S_{uu}^{-1} S_{ux}}_{=:L_{ny}} x_k \tag{14}$$

förutsatt att S_{uu} är inverterbar.

Eftersom den framtagna reglerstrategin är definierad genom minimering av $\mathcal{Q}$ har vi att

$$\mathcal{Q}(x_k, -L_{ny} x_k) \leq \mathcal{Q}(x_k, -Lx_k) = V_L(x_k),$$

där vi har använt (11) i den sista likheten. Det här innebär att vi får en bättre eller minst lika bra reglerprestanda genom att byta tillståndsåterkoppling från L till L_{ny} vid tiden k för att därefter använda L.

Resonemanget kan upprepas vid nästa tidpunkt, $k+1$, och slutsatsen är då att vi tjänar, eller åtminstone inte förlorar, på att använda den nya tillståndsåterkopplingen vid tidpunkterna k och $k+1$. En fortsatt användning ger med hjälp av induktion att

$$V_{L_{ny}}(x_k) \leq V_L(x_k),$$

det vill säga att ett permanent byte till den nya tillståndsåterkopplingen resulterar i en bättre eller minst lika bra reglerprestanda eftersom målfunktionen antingen avtar eller är konstant.

Kommentar (överkurs): Om modellen är känd kan den nya målfunktionen beräknas som

$$V_{ny}(x_k) = \frac{1}{2}x_k^T P_{L_{ny}} x_k = \frac{1}{2}(x_k^T Q x_k + x_k^T L_{ny}^T R L_{ny} x_k + x_k^T (A - BL_{ny})^T P_{ny}(A - BL_{ny})x_k),$$

vilket tillsammans med (12) ger

$$\begin{aligned} P_{L_{ny}} &= Q + L_{ny}^T R L_{ny} + (A - BL_{ny})^T P_{ny}(A - BL_{ny}) \\ &= Q + ((B^T P_L B + R)^{-1} B^T P_L A)^T R (B^T P_L B + R)^{-1} B^T P_L A \\ &\quad + (A - B(B^T P_L B + R)^{-1} B^T P_L A)^T P_{ny}(A - B(B^T P_L B + R)^{-1} B^T P_L A) \end{aligned} \tag{15}$$

Antag att vi har hittat en stationär punkt, $P_{ny} = P_L$ och $L_{ny} = L$. Uppdateringen (15) kan då skrivas

$$P_L = A^T P_L A + Q - A^T P_L B (B^T P_L B + R)^{-1} B^T P_L A,$$

vilket är ekvivalent med riccatiekvationen (9). Om vi uppdaterar tillståndsåterkopplingen rekursivt med hjälp av (12) eller (14) och det resulterar i en strikt avtagande målfunktion kommer vi alltså att närma oss den optimala återkopplingen som ges av (8) och (9).

Den centrala frågan är nu hur vi ska kunna få information om funktionen $\mathcal{Q}(x_k, u_k)$ när vi inte har en modell av systemet. Lösningen visar sig vara att samla in data från experiment med det riktiga systemet. Sambandet (11) gäller vid en godtycklig tidpunkt och vid tiden $k+1$ har vi därför

$$V_L(x_{k+1}) = \mathcal{Q}(x_{k+1}, -Lx_{k+1}). \tag{16}$$

Insättning av (16) i (10) ger

$$\mathcal{Q}(x_k, u_k) = \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k) + \mathcal{Q}(x_{k+1}, -Lx_{k+1}) \tag{17}$$

vilket också kan skrivas

$$\mathcal{Q}(x_k, u_k) - \mathcal{Q}(x_{k+1}, -Lx_{k+1}) = \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k) \tag{18}$$

Högerledet i (18) kan enkelt beräknas eftersom det innehåller de båda kända matriserna Q och R (som ju är designparametrar som vi väljer själva) samt x_k som är tillståndsvektorn i systemet, som vi fortsatt antar att vi kan mäta, och styrsignalen u_k som ju bestäms av oss.

Vänsterledet i (18) är kvadratiskt i våra mätta tillstånd och kända styrsignaler men linjärt beroende på de okända parametrarna i S_{xx}, S_{xu} och S_{uu} från (13). Det här innebär att vi kan skriva $\mathcal{Q}$ som

$$\mathcal{Q}(x, u) = \varphi(x, u)^T \theta, \tag{19}$$

där vektorn θ innehåller alla unika parametrar från S_{xx}, S_{xu} och S_{uu} och $\varphi(x, u)$ är en vektor med motsvarande unika kvadratiska termer i x:s och u:s element (till exempel x_1^2, $x_1 x_2$ och $x_1 u_1$). Exemplet nedan visar hur den här omskrivningen ser ut för ett system med två tillståndsvariabler och en insignal.

Exempel

Antag att vi studerar ett system med två tillstånd och en insignal. Funktionen $\mathcal{Q}(x_k, u_k)$ definieras då av matriserna

$$S_{xx} = \begin{pmatrix} s_1 & s_2 \\ s_2 & s_3 \end{pmatrix}, \quad S_{ux} = \begin{pmatrix} s_4 & s_5 \end{pmatrix} \quad \text{och} \quad S_u = \begin{pmatrix} s_6 \end{pmatrix}$$

och vi har att

$$\begin{aligned}\mathcal{Q}(x_k,u_k) &= \frac{1}{2}\begin{pmatrix}x_k\\u_k\end{pmatrix}^T\begin{pmatrix}S_{xx} & S_{xu}\\S_{ux} & S_{uu}\end{pmatrix}\begin{pmatrix}x_k\\u_k\end{pmatrix}\\ &= \frac{1}{2}(s_1x_{k,1}^2 + 2s_2x_{k,1}x_{k,2} + s_3x_{k,2}^2 + 2s_4u_kx_{k,1} + 2s_5u_kx_{k,2} + s_6u_k^2)\\ &= \underbrace{\frac{1}{2}\begin{pmatrix}x_{k,1}^2 & 2x_{k,1}x_{k,2} & x_{k,2}^2 & 2u_kx_{k,1} & 2u_kx_{k,2} & u_k^2\end{pmatrix}}_{=\varphi(x_k,u_k)^T}\underbrace{\begin{pmatrix}s_1 & s_2 & s_3 & s_4 & s_5 & s_6\end{pmatrix}^T}_{=\theta}.\end{aligned}$$

Med (19) så kan vi skriva om (18) som

$$(\varphi(x_k,u_k) - \varphi(x_{k+1},-Lx_{k+1}))^T\theta = \frac{1}{2}(x_k^TQx_k + u_k^TRu_k). \tag{20}$$

Om vi har tillgång till systemet kan vi först mäta x_k, därefter använda en viss styrsignal u_k och till sist mäta x_{k+1}. Dessa tre storheter kan därefter sättas in i (20) så att vi får ett samband där parametervektorn θ är den enda obekanta storheten. Ett enda samband av den här typen räcker dock inte för att bestämma de okända parametrarna eftersom de i de flesta fall är fler än en till antalet. Genom att samla ihop flera samband med data från olika tidpunkter kan man däremot få tillräckligt med information för att kunna beräkna en minstakvadratskattning av θ (det vill säga ta fram minstakvadratlösningen för ett överbestämt ekvationssystem). Denna skattning ger oss information om S_{uu} och S_{ux} i $\mathcal{Q}$, vilket gör att vi kan räkna ut en ny förbättrad tillståndsåterkoppling med uttrycket (14). Det här sättet att successivt ta fram en bättre reglerstrategi kallas för förstärkningsinlärning (eng. reinforcement learning) och en komplett algoritm beskrivs nedan.

Algoritm 1: Förstärkningsinlärning (eng. reinforcement learning) för linjärkvadratisk reglering (LQ)

Välj en initial tillståndsåterkoppling $u = -L^0x$ för $j = 0$. Denna återkoppling måste stabilisera systemet men är annars godtycklig. Om man vet att systemet är stabilt kan man välja $L^0 = 0$.

Steg j:

1. Skattning av parametrarna θ:
   (a) Vid tiden $k = jN_{LS}+1, jN_{LS}+2, \ldots, (j+1)N_{LS}$: Mät systemets tillstånd x_k och använd styrsignalen $u_k = -L^jx_k + e_k$ på systemet vid tiden k. Tillägget e_k till styrsignalen är en störsignal som man kan designa själv (till exempel som en realisering av en stokastisk variabel) och som ger en bättre excitation av systemet. Mät även systemets nästa tillstånd x_{k+1}.
   (b) Vid tiden $k = (j+1)N_{LS}+1$: Bilda matriserna

$$\Phi_j = \begin{pmatrix}(\varphi(x_n,u_n) - \varphi(x_{n+1},-Lx_{n+1}))^T\\(\varphi(x_{n+1},u_{n+1}) - \varphi(x_{n+2},-Lx_{n+2}))^T\\\vdots\\(\varphi(x_{n+N_{LS}-1},u_{n+N_{LS}-1}) - \varphi(x_{n+N_{LS}},-Lx_{n+N_{LS}}))^T\end{pmatrix}$$

och

$$Y_j = \begin{pmatrix}x_n^TQx_n + u_n^TRu_n\\x_{n+1}^TQx_{n+1} + u_{n+1}^TRu_{n+1}\\\vdots\\x_{n+N_{LS}-1}^TQx_{n+N_{LS}-1} + u_{n+N_{LS}-1}^TRu_{n+N_{LS}-1}\end{pmatrix},$$

där $n = jN_{LS}+1$ och beräkna

$$\hat{\theta}_j = (\Phi_j^T\Phi_j)^{-1}\Phi_j^TY_j$$

2. Uppdatering av tillståndsåterkopplingen: Skapa matriserna $S_{uu,j}$ och $S_{ux,j}$ som ingår i uttrycket för $\mathcal{Q}$ med hjälp av parametrarna i $\hat{\theta}_j$ och beräkna en ny tillståndsåterkoppling

$$L^{j+1} = S_{uu,j}^{-1} S_{ux,j}$$

Sätt $j := j + 1$ och gå till steg j.

Avslutningsvis ska vi se hur den här algoritmen fungerar i ett numeriskt exempel.

Exempel

Betrakta en likströmsmotor som med god noggrannhet kan beskrivas med en andra ordningens tillståndsmodell

$$\dot{x}(t) = \begin{pmatrix} 0 & 1 \\ 0 & -2 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(t),$$

där det första tillståndet är motorvinkeln, det andra är vinkelhastigheten och insignalen är spänningen över motorn. Vi ska styra den här motorn med en samplande regulator med sampeltiden $T_S = 0.05\,\text{s}$ och om vi antar att styrsignalen är styckvis konstant kan vi ta fram en exakt tidsdiskret tillståndsmodell

$$x_{k+1} = \begin{pmatrix} 1.0000 & 0.0476 \\ 0.0000 & 0.9048 \end{pmatrix} x_k + \begin{pmatrix} 0.0012 \\ 0.0476 \end{pmatrix} u_k,$$

där $x_k = x(kT_S)$ och $u_k = u(kT_S)$.

Antag att vi har ett initialtillstånd

$$x_1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

och att vi vill använda linjärkvadratisk reglering med

$$Q = \begin{pmatrix} 10 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{och} \quad R = 1$$

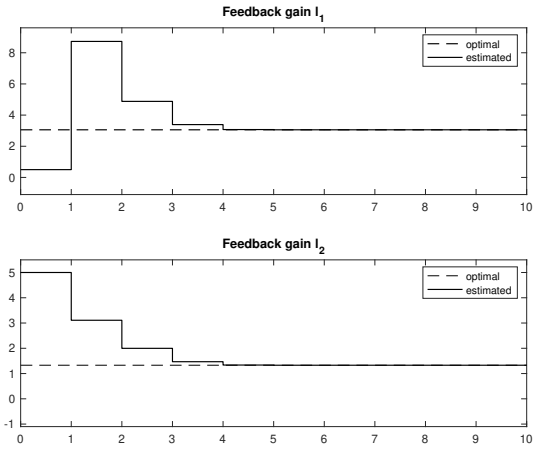
trots att modellen ovan är okänd. Förstärkningsinlärning enligt algoritm 1 är då ett alternativ. För att använda den algoritmen behöver vi dock välja tre parametrar, den initiala tillståndsåterkopplingen L^0, steglängden N_{LS} under vilken vi samlar in data för att kunna skatta en ny tillståndsåterkoppling och störsignalen e_k.

Genom mer eller mindre slumpmässiga tester på systemet kan vi hitta en stabiliserande tillståndsåterkoppling

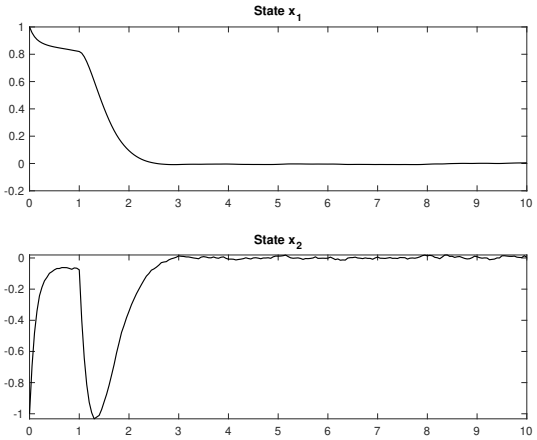
$$L^0 = \begin{pmatrix} 0.5 & 5 \end{pmatrix}$$

och vi väljer $N_{LS} = 20$ för att med lite marginal samla in mer data än vad som minimalt behövs för att skatta θ. I vårt fall har vi ju 6 parametrar att skatta (se det tidigare exemplet). Störsignalen e_k väljs som oberoende realiseringar av en normalfördelad stokastisk variabel med standardavvikelse 0.1.

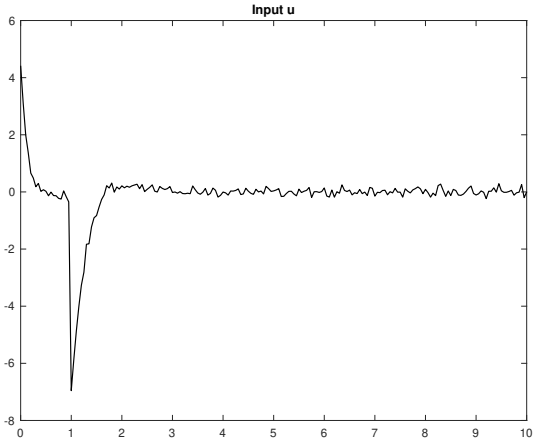
Resultatet av att använda algoritm 1 på en simulerad version av det aktuella systemet i ett 10 s (200 sampel) långt experiment visas i figur 1, 2 och 3. I figur 1 kan man se att den initiala tillståndsåterkopplingen är ganska långt ifrån den optimala (som är framräknad med hjälp av modellen och ekvation (8) och (9)) men att algoritmen för förstärkningsinlärning efter 4 iterationer har hittat de optimala regulatorparametrarna. Figur 2 visar att tillstånden svänger in till origo på ett par sekunder, men med en inte helt intuitiv transient som beror på den stora förändringen i reglerstrategi efter den första iterationen. Samma beteende syns i styrsignalen i figur 3. De kvarvarande variationerna i tillstånden och styrsignalen beror här på störsignalen e_k. Ett alternativ skulle kunna vara att låsa tillståndsåterkopplingen när vi ser att förstärkningsinlärningen har konvergerat och då skulle beroendet på e_k förstås försvinna.



Figur 1: *Parametrarna i tillståndsåterkopplingen som fås med förstärkningsinlärning (heldragna) och de optimala (modellbaserade) regulatorparametrarna (streckade) som fås i likströmsmotorexemplet.*



Figur 2: *Tillstånden $x_{k,1}$ (motorvinkeln) och $x_{k,2}$ (vinkelhastigheten) som fås när man använder förstärkningsinlärning i likströmsmotorexemplet.*



Figur 3: *Styrsignalen u_k som fås när man använder förstärkningsinlärning i likströmsmotorexemplet.*

Referenser

[1] S. J. Bradtke. Reinforcement learning applied to linear quadratic regulation. I: *Advances in Neural Information Processing Systems (NIPS)*, ss 295–302, 1993.

[2] T. Glad och L. Ljung. *Reglerteknik – Grundläggande teori*. Studentlitteratur, fjärde utgåvan, 2006.

[3] F. L. Lewis, D. Vrabie och K. G. Vamvoudakis. Reinforcement learning and feedback control – using natural decision methods to design optimal adaptive controllers. *IEEE Control Systems Magazine*, 32(6):76–105, 2012.