In[•]:=
```
Get["D:\\Dhruv\\MITACS_Summer_22\\Codes\\cartPendulum.m"]
```

In[•]:=

In[•]:=
```
ClearAll["Global`*"];
CalculateSMatrix[x1a_,xdot1a_,θ1a_,θdot1a_,u1a_,τ_,A_]:= Module[{x,L,RHS,xdot,θ,θdot,u,K,S,soltn

xState = {x,xdot,θ,θdot};
x2dot = 1/(1-A Cos[θ]^2) (u+A θdot^2 Sin[θ]+A Cos[θ] Sin[θ]);
θ2dot= 1/(1-A Cos[θ]^2) (-Sin[θ]-Cos[θ] (u+A θdot^2 Sin[θ]));
fx = {xdot,x2dot,θdot,θ2dot};
L = 1/2*u^2;
Af = Grad[fx,xState]; (* For nD stuff use Grad*)
Bf = D[fx,u] ;(*For 1D stuff use D*)
Q = Grad[Grad[L,xState],xState]; (* Fix this *)
Q = {{1,0,0,0},{0,1,0,0},{0,0,1,0},{0,0,0,1}};
Mf = Grad[D[L,u],xState];
R = D[L,{u,2}];
S0 = {{0,0,0,0},{0,0,0,0},{0,0,0,0},{0,0,0,0}};
RHS[t_] := (IdentityMatrix[4] + Af^T.S[t] + S[t].Af - KroneckerProduct[S[t].Bf,Bf^T.S[t]]) /. {x→
sol2 = S /. NDSolve[{S'[t]== RHS[t],S[0]==S0},S,{t,0,τ }];
S = sol2[[1]]
]
CalculateGains[x1a_,xdot1a_,θ1a_,θdot1a_,u1a_,time_,A_,τ_,S_]:= Module[{x,L,RHS,xdot,θ,θdot,u,K,
xState = {x,xdot,θ,θdot};
x2dot = 1/(1-A Cos[θ]^2) (u+A θdot^2 Sin[θ]+A Cos[θ] Sin[θ]);
θ2dot= 1/(1-A Cos[θ]^2) (-Sin[θ]-Cos[θ] (u+A θdot^2 Sin[θ]));
fx = {xdot,x2dot,θdot,θ2dot};
Bf = D[fx,u] ;(*For 1D stuff use D*)
K = (Bf^T.S[τ - time])/. {x→ x1a[time], xdot → xdot1a[time], θ →θ1a[time], θdot → θdot1a[time],
K
]
testWithFB[ICs_,τ_,τ1_,xff0_,xdotff0_,θff0_,θdotff0_,uff0_,A_]:=Module[{eq,init,θ,θdot,θff,θdotf
κ1=κ2=3;  (* lqr for q=r for balancing pendulum *)
κ3 = -0.1;κ4 = -0.65;
xff[t_]:=Piecewise[{{xff0[t],0≤t≤τ}},0];
xdotff[t_]:=Piecewise[{{xdotff0[t],0≤t≤τ}},0];
θff[t_]:=Piecewise[{{θff0[t],0≤t≤τ}},π];
θdotff[t_]:=Piecewise[{{θdotff0[t],0≤t≤τ}},0];
uff[t_]:=Piecewise[{{uff0[t],0≤t≤τ}},0];
S = CalculateSMatrix[xff,xdotff,θff,θdotff,uff,τ,A];
K[t_] := CalculateGains[xff,xdotff,θff,θdotff,uff,t,A,τ,S];
ufb[t_] := Piecewise[{{
```

```
K[t].{xff[t]-x[t],xdotff[t]-xdot[t],θff[t]-θ[t],θdotff[t]-θdot[t]},0≤t≤τ}},κ1(θff[t]-θ[t])+κ2

eq={x'[t]==xdot[t],xdot'[t]==1/(1-A Cos[θ[t]]^2) (u[t]+A θdot[t]^2 Sin[θ[t]]+A Cos[θ[t]] Sin[θ[

init={x[0]==ICs⟦1⟧,xdot[0]==ICs⟦2⟧,θ[0]==ICs⟦3⟧,θdot[0]==ICs⟦4⟧};

{xs,xdots,θs,θdots}=NDSolveValue[{eq,init},{x,xdot,θ,θdot},{t,0,τ1},Method→{"DiscontinuityProces
us[t_]:=uff[t]+Piecewise[{{K[t].{xff[t]-xs[t],xdotff[t]-xdots[t],θff[t]-θs[t],θdotff[t]-θdots[
J = NIntegrate[us[t]^2,{t,0,τ}];
{xs,xdots,θs,θdots,us,J}]
```

Understanding Effect of Changing n

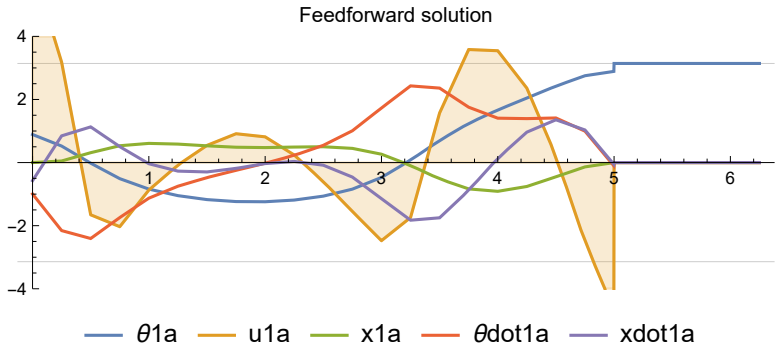
In[ ]:=

```
n = 20; τ = 5; τ1 = τ * 1.25; A = 0.2; order = 1; maxIter = 100;
xdotMin = -1;
xdotMax = 1;
θMin = -π;
θMax = π;
θdotMin = -1;
θdotMax = 1;

xdotInit = RandomReal[{xdotMin, xdotMax}];
θInit = RandomReal[{θMin, θMax}];
θdotInit = RandomReal[{θdotMin, θdotMax}];
ICs = {0, xdotInit, θInit, θdotInit}; (* Random Initialization *)
ICs = {0.89486028245609, -0.9468360111172656, -0.002994757534002989, 1.677668990900959};
(* Works *)
ICs = {0, -0.5735358669582524`, 0.8898706763193971`, -0.9946572285334812`};
(* Doesnt Work *)
{x1a, xdot1a, θ1a, θdot1a, u1a} = ffCartPendulum[ICs, n, τ, τ1, A, order, maxIter];
{x1b, xdot1b, θ1b, θdot1b, u1b, J1} = testSwingUp[ICs, τ, τ1, u1a, A];
{x1c, xdot1c, θ1c, θdot1c, u1c, J} =
 testWithFB[ICs, τ, τ1, x1a, xdot1a, θ1a, θdot1a, u1a, A];
p1a = Plot[{θ1a[t], u1a[t], x1a[t], θdot1a[t], xdot1a[t]}, {t, 0, τ1}, Filling → {2 → Axis},
  PlotRange → {-4, 4}, PlotLegends → {"θ1a", "u1a", "x1a", "θdot1a", "xdot1a"},
  PlotLabel → "Feedforward solution", AspectRatio → 1 / 3,
  ImageSize → 400, GridLines → {None, {-π, π}}]
p1b = Plot[{θ1b[t], u1a[t], x1b[t], θdot1b[t], xdot1b[t]},
  {t, 0, τ1}, PlotRange → {-4, 4}, Filling → {2 → Axis},
  PlotLegends → {"θ1b", "u1b", "x1b", "θdot1b", "xdot1b"}, PlotLabel → "Test on dynamics",
  AspectRatio → 1 / 3, ImageSize → 400, GridLines → {None, {-π, π}}]
p1c = Plot[{θ1c[t], u1c[t], x1c[t], θdot1c[t], xdot1c[t]}, {t, 0, τ1}, PlotRange → {-4, 4},
  Filling → {2 → Axis}, PlotLegends → {"θ1c", "u1c", "x1c", "θdot1c", "xdot1c"},
  PlotLabel → "LQR feedback solution", AspectRatio → 1 / 3,
  ImageSize → 400, GridLines → {None, {-π, π}}]
```

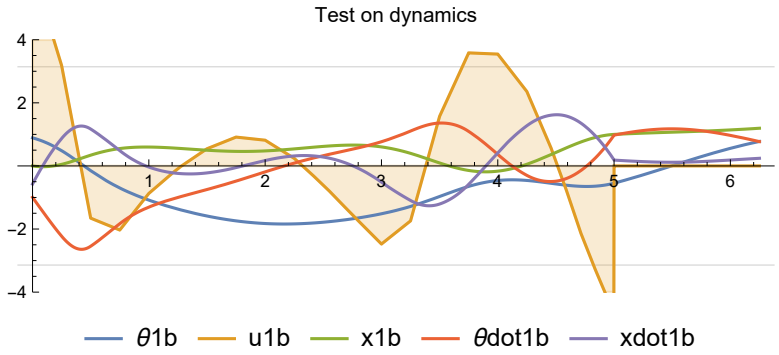
••• FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

••• NIntegrate: NIntegrate failed to converge to prescribed accuracy after 9 recursive bisections in t$536993 near {t$536993} = {3.25188}. NIntegrate obtained 35.723225659375935` and 0.00016528899694322267` for the integral and error estimates.

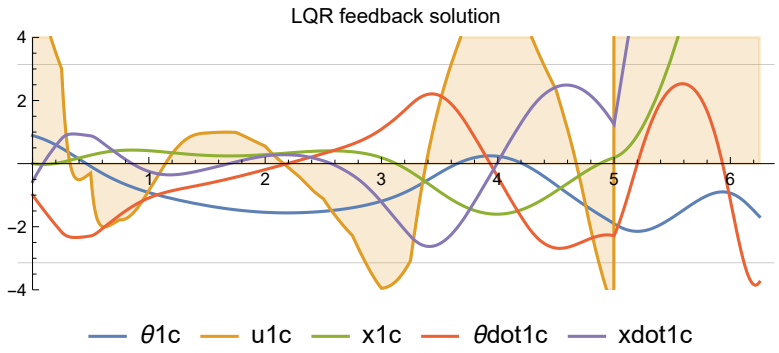
Out[•]=


Feedforward solution
θ1a
u1a
x1a
θdot1a
xdot1a

Out[•]=


Test on dynamics
θ1b
u1b
x1b
θdot1b
xdot1b

Out[•]=


LQR feedback solution
θ1c
u1c
x1c
θdot1c
xdot1c

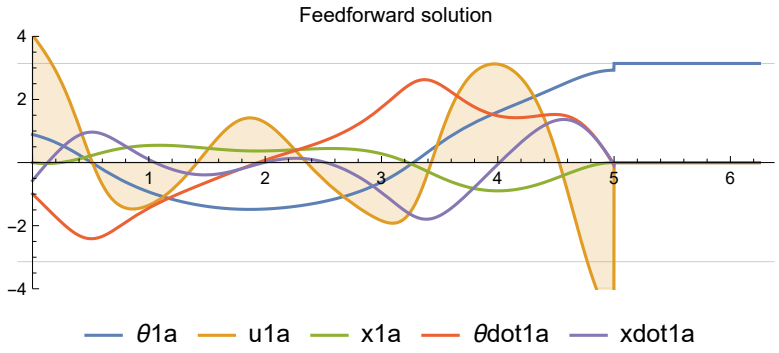
In[•]:=
```
n = 234; τ = 5; τ1 = τ * 1.25; A = 0.2; order = 1; maxIter = 100;
xdotMin = -1;
xdotMax = 1;
θMin = -π;
θMax = π;
θdotMin = -1;
θdotMax = 1;

xdotInit = RandomReal[{xdotMin, xdotMax}];
θInit = RandomReal[{θMin, θMax}];
θdotInit = RandomReal[{θdotMin, θdotMax}];
ICs = {0, xdotInit, θInit, θdotInit}; (* Random Initialization *)
ICs = {0.89486028245609, -0.9468360111172656, -0.002994757534002989, 1.677668990900959};
(* Works *)
ICs = {0, -0.5735358669582524`, 0.8898706763193971`, -0.9946572285334812`};
(* Doesnt Work *)
{x1a, xdot1a, θ1a, θdot1a, u1a} = ffCartPendulum[ICs, n, τ, τ1, A, order, maxIter];
{x1b, xdot1b, θ1b, θdot1b, u1b, J1} = testSwingUp[ICs, τ, τ1, u1a, A];
{x1c, xdot1c, θ1c, θdot1c, u1c, J} =
 testWithFB[ICs, τ, τ1, x1a, xdot1a, θ1a, θdot1a, u1a, A];
p1a = Plot[{θ1a[t], u1a[t], x1a[t], θdot1a[t], xdot1a[t]}, {t, 0, τ1}, Filling → {2 → Axis},
  PlotRange → {-4, 4}, PlotLegends → {"θ1a", "u1a", "x1a", "θdot1a", "xdot1a"},
  PlotLabel → "Feedforward solution", AspectRatio → 1 / 3,
  ImageSize → 400, GridLines → {None, {-π, π}}]
p1b = Plot[{θ1b[t], u1a[t], x1b[t], θdot1b[t], xdot1b[t]},
  {t, 0, τ1}, PlotRange → {-4, 4}, Filling → {2 → Axis},
  PlotLegends → {"θ1b", "u1b", "x1b", "θdot1b", "xdot1b"}, PlotLabel → "Test on dynamics",
  AspectRatio → 1 / 3, ImageSize → 400, GridLines → {None, {-π, π}}]
p1c = Plot[{θ1c[t], u1c[t], x1c[t], θdot1c[t], xdot1c[t]}, {t, 0, τ1}, PlotRange → {-4, 4},
  Filling → {2 → Axis}, PlotLegends → {"θ1c", "u1c", "x1c", "θdot1c", "xdot1c"},
  PlotLabel → "LQR feedback solution", AspectRatio → 1 / 3,
  ImageSize → 400, GridLines → {None, {-π, π}}]
```

••• FindRoot: Failed to converge to the requested accuracy or precision within 100 iterations.

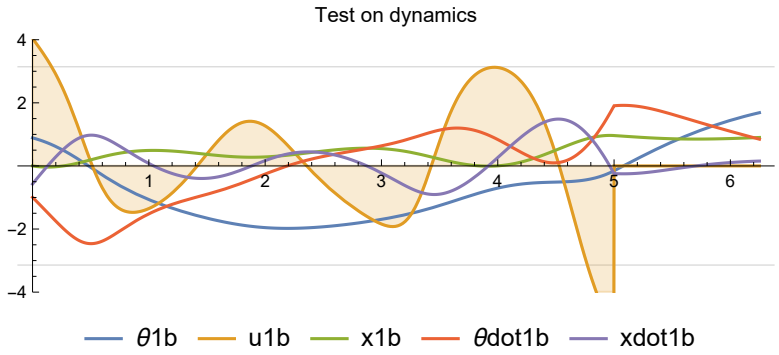
••• NIntegrate: Numerical integration converging too slowly; suspect one of the following: singularity, value of the integration is 0, highly oscillatory integrand, or WorkingPrecision too small.

••• NIntegrate: NIntegrate failed to converge to prescribed accuracy after 9 recursive bisections in t$543645 near {t$543645} = {3.27141}. NIntegrate obtained 62.66405602988513` and 0.00046666133920119507` for the integral and error estimates.

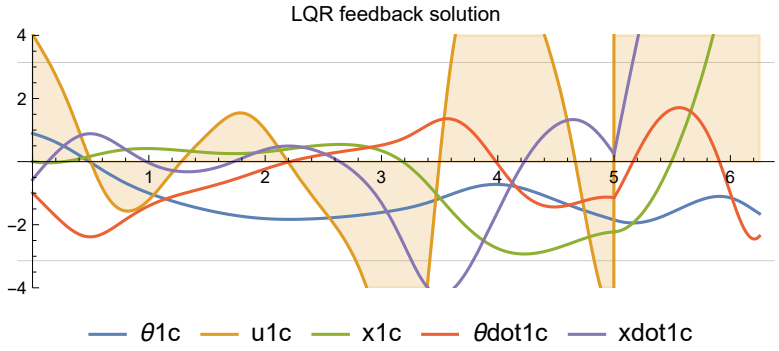
Out[•]=



Out[•]=



Out[•]=

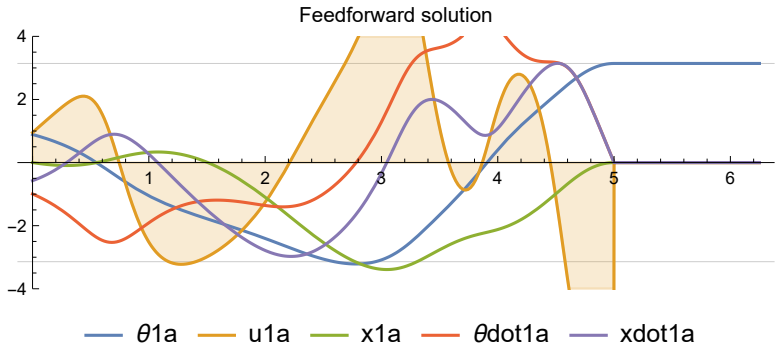


By Changing n from 234 to 235 we move to a completely new solution!!!

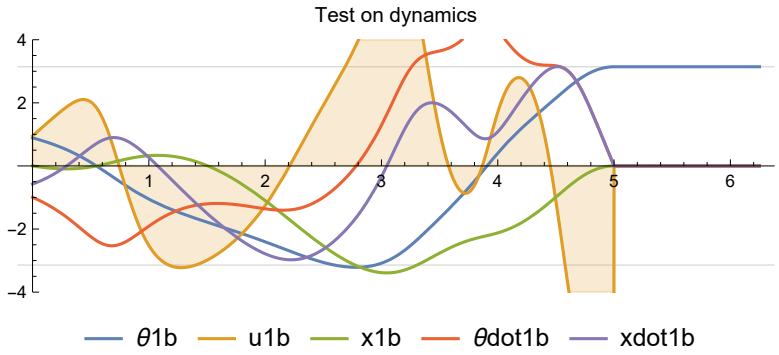
In[ ]:=
```
n = 235;
τ = 5;
τ1 = τ * 1.25;
A = 0.2;
order = 1;
maxIter = 100; (* Order of Interpolation doesnt matter for such large n*)
xdotMin = -1;
xdotMax = 1;
θMin = -π;
θMax = π;
θdotMin = -1;
θdotMax = 1;

xdotInit = RandomReal[{xdotMin, xdotMax}];
θInit = RandomReal[{θMin, θMax}];
θdotInit = RandomReal[{θdotMin, θdotMax}];
ICs = {0, xdotInit, θInit, θdotInit}; (* Random Initialization *)
ICs = {0.89486028245609, -0.9468360111172656, -0.002994757534002989, 1.677668990900959};
(* Works *)
ICs = {0, -0.5735358669582524`, 0.8898706763193971`, -0.9946572285334812`};
(* Doesnt Work *)
{x1a, xdot1a, θ1a, θdot1a, u1a} = ffCartPendulum[ICs, n, τ, τ1, A, order, maxIter];
{x1b, xdot1b, θ1b, θdot1b, u1b, J1} = testSwingUp[ICs, τ, τ1, u1a, A];
{x1c, xdot1c, θ1c, θdot1c, u1c, J} =
 testWithFB[ICs, τ, τ1, x1a, xdot1a, θ1a, θdot1a, u1a, A];
p1a = Plot[{θ1a[t], u1a[t], x1a[t], θdot1a[t], xdot1a[t]}, {t, 0, τ1}, Filling → {2 → Axis},
  PlotRange → {-4, 4}, PlotLegends → {"θ1a", "u1a", "x1a", "θdot1a", "xdot1a"},
  PlotLabel → "Feedforward solution", AspectRatio → 1 / 3,
  ImageSize → 400, GridLines → {None, {-π, π}}]
p1b = Plot[{θ1b[t], u1a[t], x1b[t], θdot1b[t], xdot1b[t]},
  {t, 0, τ1}, PlotRange → {-4, 4}, Filling → {2 → Axis},
  PlotLegends → {"θ1b", "u1b", "x1b", "θdot1b", "xdot1b"}, PlotLabel → "Test on dynamics",
  AspectRatio → 1 / 3, ImageSize → 400, GridLines → {None, {-π, π}}]
p1c = Plot[{θ1c[t], u1c[t], x1c[t], θdot1c[t], xdot1c[t]}, {t, 0, τ1}, PlotRange → {-4, 4},
  Filling → {2 → Axis}, PlotLegends → {"θ1c", "u1c", "x1c", "θdot1c", "xdot1c"},
  PlotLabel → "LQR feedback solution", AspectRatio → 1 / 3,
  ImageSize → 400, GridLines → {None, {-π, π}}]
```

NIntegrate: NIntegrate failed to converge to prescribed accuracy after 9 recursive bisections in t$584614 near {t$584614} = {3.26164}. NIntegrate obtained 51.54465638857705` and 0.0009468894614348335` for the integral and error estimates.

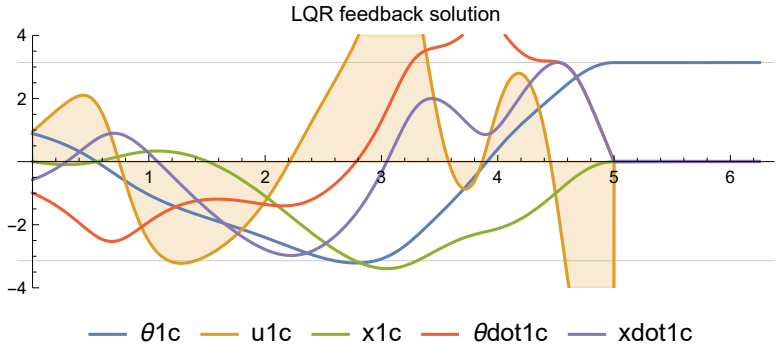
Out[•]=



Out[•]=



Out[•]=



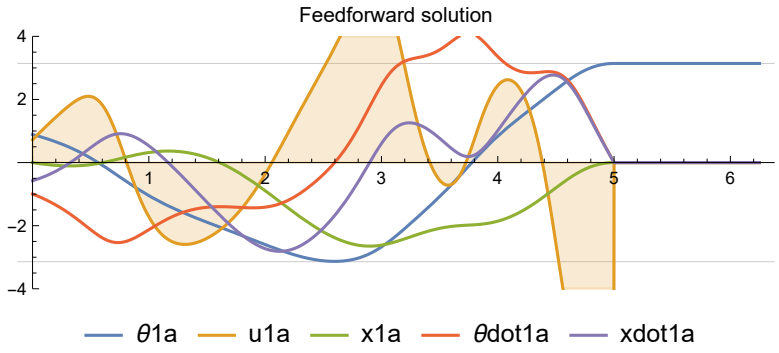
Furthermore reducing the max iterations to 20 also works as the feedback corrects the errors

In[ ]:=
```
n = 235;
τ = 5;
τ1 = τ * 1.25;
A = 0.2;
order = 1;
maxIter = 20; (* Order of Interpolation doesnt matter for such large n*)
xdotMin = -1;
xdotMax = 1;
θMin = -π;
θMax = π;
θdotMin = -1;
θdotMax = 1;

xdotInit = RandomReal[{xdotMin, xdotMax}];
θInit = RandomReal[{θMin, θMax}];
θdotInit = RandomReal[{θdotMin, θdotMax}];
ICs = {0, xdotInit, θInit, θdotInit}; (* Random Initialization *)
ICs = {0.89486028245609, -0.9468360111172656, -0.002994757534002989, 1.677668990900959};
(* Works *)
ICs = {0, -0.5735358669582524`, 0.8898706763193971`, -0.9946572285334812`};
(* Doesnt Work *)
{x1a, xdot1a, θ1a, θdot1a, u1a} = ffCartPendulum[ICs, n, τ, τ1, A, order, maxIter];
{x1b, xdot1b, θ1b, θdot1b, u1b, J1} = testSwingUp[ICs, τ, τ1, u1a, A];
{x1c, xdot1c, θ1c, θdot1c, u1c, J} =
 testWithFB[ICs, τ, τ1, x1a, xdot1a, θ1a, θdot1a, u1a, A];
p1a = Plot[{θ1a[t], u1a[t], x1a[t], θdot1a[t], xdot1a[t]}, {t, 0, τ1}, Filling → {2 → Axis},
  PlotRange → {-4, 4}, PlotLegends → {"θ1a", "u1a", "x1a", "θdot1a", "xdot1a"},
  PlotLabel → "Feedforward solution", AspectRatio → 1 / 3,
  ImageSize → 400, GridLines → {None, {-π, π}}]
p1b = Plot[{θ1b[t], u1a[t], x1b[t], θdot1b[t], xdot1b[t]},
  {t, 0, τ1}, PlotRange → {-4, 4}, Filling → {2 → Axis},
  PlotLegends → {"θ1b", "u1b", "x1b", "θdot1b", "xdot1b"}, PlotLabel → "Test on dynamics",
  AspectRatio → 1 / 3, ImageSize → 400, GridLines → {None, {-π, π}}]
p1c = Plot[{θ1c[t], u1c[t], x1c[t], θdot1c[t], xdot1c[t]}, {t, 0, τ1}, PlotRange → {-4, 4},
  Filling → {2 → Axis}, PlotLegends → {"θ1c", "u1c", "x1c", "θdot1c", "xdot1c"},
  PlotLabel → "LQR feedback solution", AspectRatio → 1 / 3,
  ImageSize → 400, GridLines → {None, {-π, π}}]
```

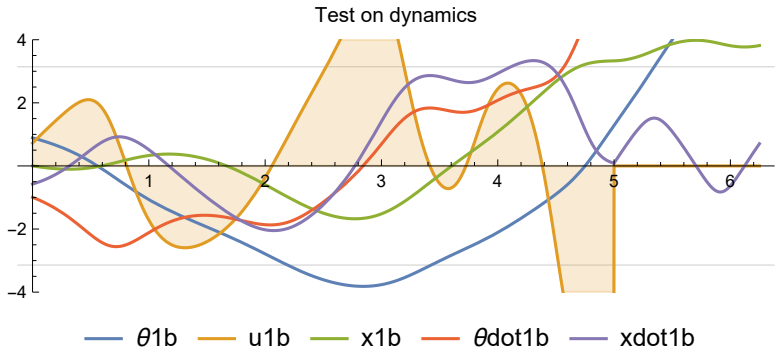
FindRoot: Failed to converge to the requested accuracy or precision within 20 iterations.

NIntegrate: NIntegrate failed to converge to prescribed accuracy after 9 recursive bisections in t$591587 near {t$591587} = {2.93938}. NIntegrate obtained 48.77007624715693` and 0.00034474967229073414` for the integral and error estimates.

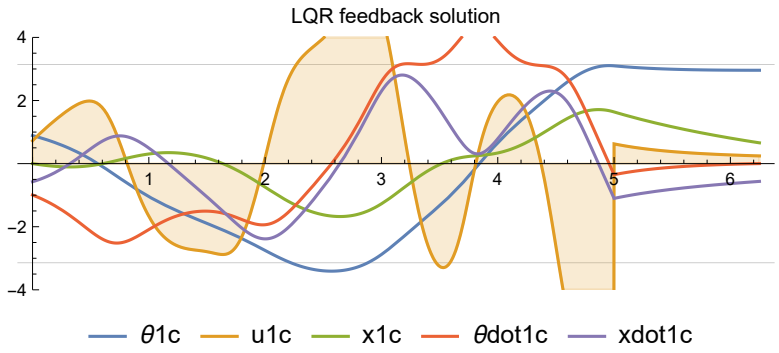
Out[•]=


Feedforward solution
θ1a
u1a
x1a
θdot1a
xdot1a

Out[•]=


Test on dynamics
θ1b
u1b
x1b
θdot1b
xdot1b

Out[•]=


LQR feedback solution
θ1c
u1c
x1c
θdot1c
xdot1c