1 Polynomially asymptotically autonomous systems – 2D

Let $\sigma \in \mathbb{N}$, $\sigma \geq 2$. Consider the polynomially asymptotically autonomous equation $\dot{x} = f(t,x)$, $x \in \mathbb{R}^n$, where $f, g \in C^{\sigma+1}$ and $f(t,x) \to g(x)$ in the following sense: for each compact set $K \subset \mathbb{R}^n$ and each $\delta > 0$ there is a T with

$$\begin{aligned} \|\partial^\gamma f(t,x) - \partial^\gamma g(x)\| \, t^\epsilon &< \delta \text{ for all } \gamma \text{ with } |\gamma| \leq \sigma \text{ and } \gamma_0 = 0, \\ \|\partial^\gamma f(t,x)\| \, t^{\epsilon(\sigma+1)+\gamma_0} &< \delta \text{ for all } \gamma \text{ with } |\gamma| \leq \sigma \text{ and } \gamma_0 \neq 0. \end{aligned}$$

for all $x \in K$ and all $t > T$, where

$$0 < \epsilon < \frac{1}{\sigma - 1}.$$

Let $x(t) = 0$ be an exponentially stable solution.

Time transformation

Define $h(\tau) = (-\tau)^{-1/\epsilon}$. The transformed system is

$$\begin{aligned} \dot{\tau} &= \epsilon(-\tau)^{1/\epsilon+1} \\ \dot{x} &= \begin{cases} f((-\tau)^{-1/\epsilon}, x) & \text{if } \tau \neq 0 \\ g(x) & \text{if } \tau = 0 \end{cases} \end{aligned}$$

Local Lyapunov function

Solve the matrix equation

$$Dg(0)Q^T + QDg(0) = -I$$

for $Q \in \mathbb{R}^{n \times n}$. Set

$$v_{loc}(x) = x^T \, Q \, x.$$

Find R^* such that all points in

$$E := \{x \in \mathbb{R}^n \mid v_{loc}(x) \leq R^*\}$$

satisfy $v'_{loc}(x) = \nabla V_{loc}(x) \cdot g(x) < 0$.

Radial Basis Functions

Choose a Radial Basis Function ϕ. For $n = 1$ or $n = 2$ we choose $c > 0$ and $\phi(r) = \psi_{4,2}(c \cdot r) = \begin{cases} (1-cr)^6[35c^2r^2 + 18cr + 3] & \text{for } r \leq \frac{1}{c} \\ 0 & \text{otherwise.} \end{cases}$

Choose grid points $X \subset (-\infty, 0] \times \mathbb{R}^n$ approximately in the expected basin of attraction – do not include $(0, 0)$.

Find R such that all points in

$$K_1 := \{(\tau, x) \in (-\infty, 0] \times \mathbb{R}^n \mid v(\tau, x) = R\}$$

satisfy $v'(\tau, x) < 0$, and all points in

$$K_2 := \{(0, x) \in \{0\} \times \mathbb{R}^n \mid v(\tau, x) \leq R\}$$

satisfy $v'(0, x) < 0$ or $x \in E$.

Then K is a subset of the basin of attraction.

Transform K back:

$$\tilde{K} := \left\{(t, x) \in (-\infty, \infty) \times \mathbb{R}^n \mid v(-t^{-\epsilon}, x) \leq R\right\}$$

Matlab Files

The plots for the Matlab files are designed for dimension $n = 2$.

- **f_fun.m** is the right-hand side $f(t, x)$ of the original system.
- **g_fun.m** is the right-hand side $g(x)$ of the limiting system.
- **F.m** is the right-hand side $F(\tau, x)$ of the transformed system, depending on the transformation and in particular ϵ.
- **hinv.m** is the inverse of the transformation h, i.e. $h^{-1}(t) = -t^{-\epsilon}$.
- **locv.m** is the local Lyapunov function $x^T Q x$ where Q solves $Dg(0)^T Q + QDg(0) = -I$.
- **gradlocv.m** is the gradient of $locv(x)$ with respect to x
- **plotlocv.m**

  plotlocv(MAXx,xnumber,MAXy,ynumber,valloc),

  plots the level set $v_{loc}(x)$ =valloc in Figure 6 and 3 for $\tau \in [-0.2, 0]$ and the function $v_{loc}(x)$ in Figure 1 on a grid in $[-0.2, 0] \times [-MAXx, MAXx] \times [-MAXy, MAXy]$ where the x-interval is divided into 2 xnumber equal steps and the y-interval is divided into 2 ynumber equal steps

- **plotlocvs.m**

  plotlocvs(MAXx,xnumber,MAXy,ynumber),

  plots the level set $v'_{loc}(x) = 0$ in Figure 3 for $\tau \in [-0.2, 0]$ on a grid in $[-0.2, 0] \times [-MAXx, MAXx] \times [-MAXy, MAXy]$ where the x-interval is divided into 2 xnumber equal steps and the y-interval is divided into 2 ynumber equal steps

- **phi.m, phi1.m, phi2.m**

  phi(r) is the Radial Basis Function $\psi_{4,2}(c \cdot r)$, phi1 is defined by $\frac{d\,phi}{dr}/r$, and phi2 is defined by $\frac{d\,phi1}{dr}/r$

- **coefficients.m**

  [alpha,points,N]=coefficients(T,tN,xN,yN,xdist,ydist,c,n,epsilon)

  calculates the coefficients alpha of the approximation v

  - $T < 0$ and $[T, 0]$ is the time interval
  - tN denotes the number of steps in time direction from $-T$ to 0
  - xdist is the distance of a step in x-direction
  - xN denotes the number of steps in x-direction
  - ydist the distance in y-direction
  - yN denotes the number of steps in y-direction
  - c denotes the constant in the RBF
  - n is the dimension
  - points are the points (τ_k, x_k) in $[T, 0] \times \mathbb{R}^n$, N points

- **v.m** [v]=v(c,alpha,points,epsilon,t,x,y)

  calculates the value of the approximation v at point (t, x, y)

- **vs.m** [vs]=vs(c,alpha,points,epsilon,t,x,y)

  calculates the orbital derivative v' of approximation v at (t, x, y)

- **plotv.m**

  plotv(T,MAXx,MAXy,c,n,epsilon,alpha,points,tnumber,xnumber,ynumber,val),

  plots the level set $v(\tau, x, y)$ =val in Figure 6 on a grid in $[T, 0] \times [-MAXx, MAXx] \times [-MAXy, MAXy]$ where the time interval is divided into tnumber, the x-interval is divided into 2 xnumber equal steps and the y-interval is divided into 2 ynumber equal steps

- **plotvs.m**

  plotvs(T,MAXx,MAXy,c,n,epsilon,alpha,points,tnumber,xnumber,ynumber),

  plots the level set $v'(\tau, x, y) = 0$ in Figure 6 on a grid in $[T, 0] \times [-MAXx, MAXx] \times [-MAXy, MAXy]$ where the time interval is divided into tnumber, the x-interval is divided into 2 xnumber equal steps and the y-interval is divided into 2 ynumber equal steps

- **plotvorg.m**

  plotvorg(T1,T2,MAXx,MAXy,c,n,epsilon,alpha,points,tnumber,xnumber,ynumber),

  plots the level set $v(t, x, y)$ =val in Figure 7 on a grid in $[T1, T2 \times [-MAXx, MAXx] \times [-MAXy, MAXy]$ where the time interval is divided into tnumber, the x-interval is divided into 2 xnumber equal steps and the y-interval is divided into 2 ynumber equal steps

- **plotvsorg.m**

  plotvsorg(T1,T2,MAXx,MAXy,c,n,epsilon,alpha,points,tnumber,xnumber,ynumber),

  plots the level set $v'(t, x, y) = 0$ in Figure 7 on a grid in $[T1, T2 \times [-MAXx, MAXx] \times [-MAXy, MAXy]$ where the time interval is divided into tnumber, the x-interval is divided into 2 xnumber equal steps and the y-interval is divided into 2 ynumber equal steps

Figures

3. Level set $v'_{loc}(x, y) = 0$ (red), $v_{loc}(x, y) = valloc$ (green)
6. Level set $v'(\tau, x) = 0$ (red), $v(\tau, x) = val$ (black) and $v_{loc}(\tau, x) = valloc$ (green)
7. Level set $v'(t, x) = 0$ (red), $v(t, x) = val$ (black) and $v_{loc}(t, x) = valloc$ (green) in the original time t

What to do

- Choose the constant ϵ and modify f_fun.m, g_fun.m, locv.m and gradlocv.m.
  - **f_fun**: define $f(t, x, y)$
  - **g_fun**: define $g(x, y)$
  - **locv**: Solve $Dg(0)^T Q + QDg(0) = -I$. Output is $(x, y)Q \begin{pmatrix} x \\ y \end{pmatrix}$.
  - **gradlocv** is the gradient of $locv(x, y)$ with respect to x, y

- Use plotlocv.m and plotlocvs.m. Choose the value **valloc** in plotlocv so small that the sublevel set of v_{loc} is in the area where v'_{loc} is negative (Figure 3)
- Choose constants T, tN, xN, xdist, c and n
- Calculate alpha using coefficients.m
- Use plotv.m and plotvs.m. Choose the value **val** in plotv so small that the level set of v is in the area where v' is negative, and the sublevel set of v at $\tau = 0$ is either in the area where v' is negative or which is covered by the sublevel set of v_{loc}
- If not possible, put more points into the grid (tN, xN larger and T,xdist smaller) and/or make area covered smaller (T,xdist smaller) – (Figure 6)
- Use plotvorg.m, plotvsorg.m and plotlocvor.m to obtain a plot with respect to the original time t (Figure 7)

1.1 Example

Example 5.2 from [1].

```
T=-0.9;tN=3;xN=4;xdist=0.15;yN=4;ydist=0.15;c=2;n=2;epsilon=1/11;
[alpha,points,N]=coefficients(T,tN,xN,yN,xdist,ydist,c,n,epsilon);

valloc=0.25;
plotlocv(1,80,1,80,valloc);
plotlocvs(1,2,54,54);

val=0;
plotvs(-1,0.8,0.8,c,n,epsilon,alpha,points,10,20,20);
plotv(-1,0.8,0.8,c,n,epsilon,alpha,points,10,15,15,val);

plotvorg(2.5,15,0.8,0.8,c,n,epsilon,alpha,points,15,15,15,val);
plotvsorg(2.5,15,0.8,0.8,c,n,epsilon,alpha,points,15,15,15);
```

Acknowledgement This work was supported by the Engineering and Physical Research Council [EP/H051627/1, EP/I000860/1].

References

[1] P. Giesl and H. Wendland, *Numerical determination of the basin of attraction for asymptotically autonomous dynamical systems*, submitted.