中国科学院软件研究所学术年会’2021
暨计算机科学国家重点实验室开放周

学术论文

Inferring Switched Nonlinear Dynamical Systems
推断切换型非线性动力系统

Xiangyu Jin[1;2], Jie An[3;4], Bohua Zhan[1;2], Naijun Zhan[1;2] and Miaomiao Zhang[3]

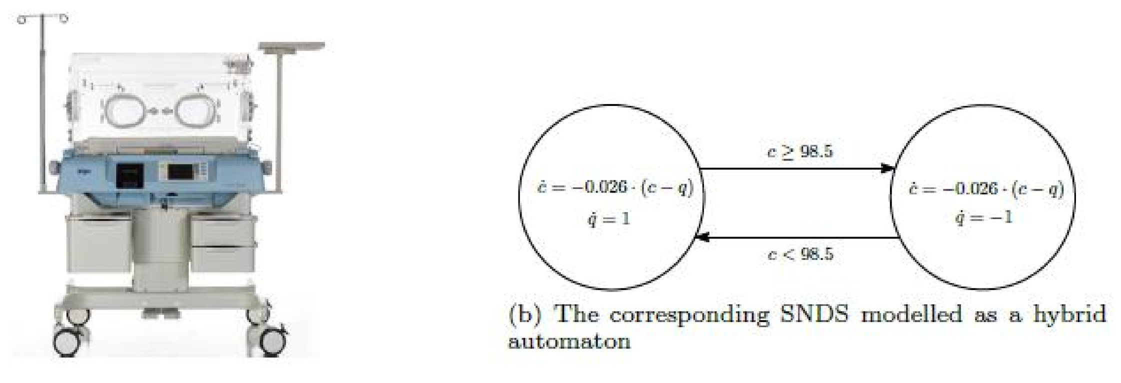
1. State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing, China
2. University of Chinese Academy of Sciences, Beijing, China
3. School of Software Engineering, Tongji University, Shanghai, China
4. Max Planck Institute for Software Systems, Kaiserslautern, Germany

Email: bzhan@ios.ac.cn

Formal Aspects of Computing

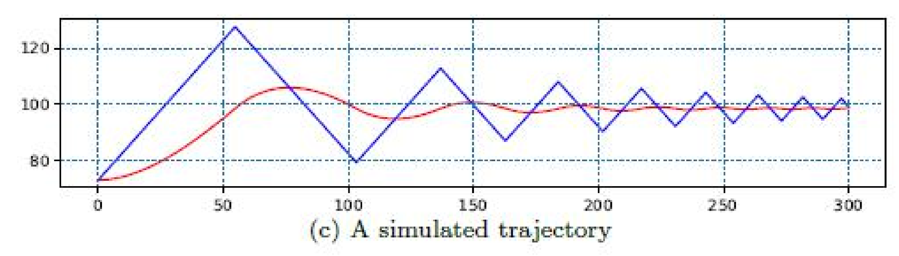
System Identification

- Purpose:
  Learning models for dynamical and hybrid systems from trajectories



(a) The Isolette system

(b) The corresponding SNDS modelled as a hybrid automaton



(c) A simulated trajectory

SNDS

Switched Nonlinear Dynamical Systems

$$\dot{x} = \begin{cases} f_1(x), & if\ x \in G_1(x) \\ f_2(x), & if\ x \in G_2(x) \\ \vdots \\ f_N(x), & if\ x \in G_N(x) \end{cases}$$

where f_i *are polynimials and*
$\forall i, j \in \{1,2,\cdots,N\},\ i \neq j \rightarrow G_i \cap G_j = \emptyset$

$trajectory: x(t_0), x(t_1), x(t_2), \cdots \qquad with \quad t_{i+1} - t_i = h$

Main Contributions

- A heuristic method to segment trajectories of a switched nonlinear dynamical system
- An inference procedure based on the segmented trajectories with a pruning search.
- An inference procedure by extending the identification methods of piecewise affine models.

Inferring Dynamics

- linear multistep method to estimate the derivatives on discrete points

$$f(\boldsymbol{x}(t_n)) \approx \frac{1}{h}\Big(\frac{137}{60}\boldsymbol{x}(t_n) - \frac{300}{60}\boldsymbol{x}(t_{n-1}) + \frac{300}{60}\boldsymbol{x}(t_{n-2}) - \frac{200}{60}\boldsymbol{x}(t_{n-3}) + \frac{75}{60}\boldsymbol{x}(t_{n-4}) - \frac{12}{60}\boldsymbol{x}(t_{n-5})\Big)$$

$$f(\boldsymbol{x}(t_n)) \approx \frac{1}{h}\Big(-\frac{137}{60}\boldsymbol{x}(t_n) + \frac{300}{60}\boldsymbol{x}(t_{n+1}) - \frac{300}{60}\boldsymbol{x}(t_{n+2}) + \frac{200}{60}\boldsymbol{x}(t_{n+3}) - \frac{75}{60}\boldsymbol{x}(t_{n+4}) + \frac{12}{60}\boldsymbol{x}(t_{n+5})\Big)$$

- linear regression to infer the coefficients of the dynamic polynomials

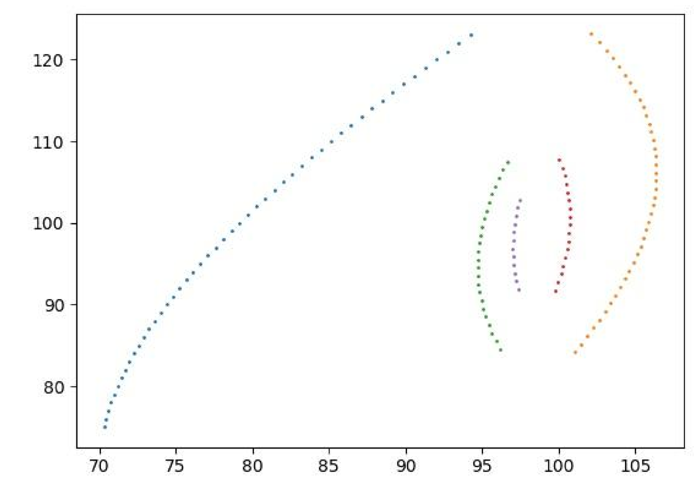
$$f(x_1, x_2) = \begin{bmatrix} a_{0,0} + a_{1,0}x_1 + a_{0,1}x_2 + a_{2,0}x_1^2 + a_{1,1}x_1x_2 + a_{0,2}x_2^2 + \cdots \\ b_{0,0} + b_{1,0}x_1 + b_{0,1}x_2 + b_{2,0}x_1^2 + b_{1,1}x_1x_2 + b_{0,2}x_2^2 + \cdots \end{bmatrix}^T$$

$$\begin{bmatrix} 1 & x_1(t_5) & x_2(t_5) & x_1(t_5)^2 & x_1(t_5)x_2(t_5) & x_2(t_5)^2 & \cdots \\ 1 & x_1(t_6) & x_2(t_6) & x_1(t_6)^2 & x_1(t_6)x_2(t_6) & x_2(t_6)^2 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_1(t_{N-1}) & x_2(t_{N-1}) & x_1(t_{N-1})^2 & x_1(t_{N-1})x_2(t_{N-1}) & x_2(t_{N-1})^2 & \cdots \\ 1 & x_1(t_N) & x_2(t_N) & x_1(t_N)^2 & x_1(t_N)x_2(t_N) & x_2(t_N)^2 & \cdots \end{bmatrix} \bullet \begin{bmatrix} a_{0,0} & b_{0,0} \\ a_{1,0} & a_{1,0} \\ a_{0,1} & b_{0,1} \\ a_{2,0} & b_{2,0} \\ a_{1,1} & b_{1,1} \\ a_{0,2} & b_{0,2} \\ \vdots & \vdots \end{bmatrix} = \begin{bmatrix} f(x_1(t_5), x_2(t_5)) \\ f(x_1(t_6), x_2(t_6)) \\ \vdots \\ f(x_1(t_{N-1}), x_2(t_{N-1})) \\ f(x_1(t_N), x_2(t_N)) \end{bmatrix}$$

$$X \cdot \theta = y$$

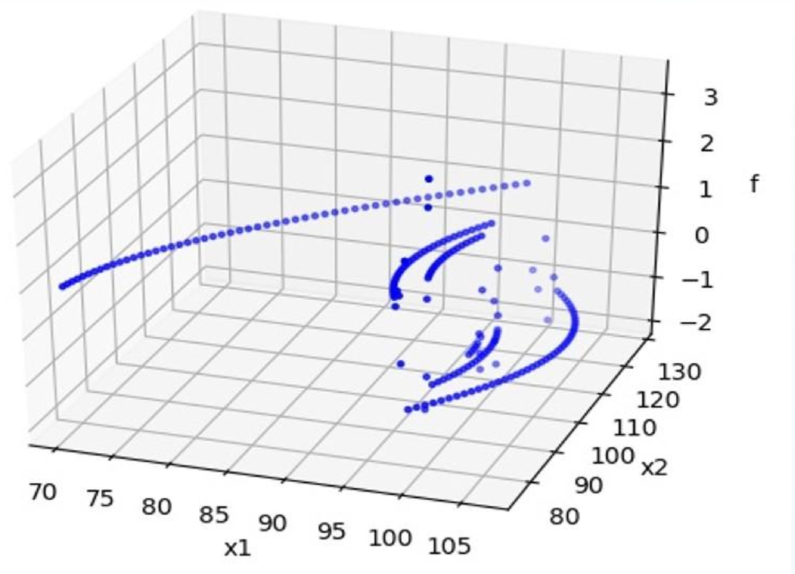
Segmenting and Searching

Segmenting the trajectory by comparing the relative difference between the two estimates of derivatives.



Pruning searching to find the best clustering and then infer dynamics.

Identification of piecewise affine models



Without prior segmentation, the estimated derivatives are captured by a piecewise affine model except the change points. The dynamics can be inferred by identification of this pwa model.

Experiments

- The Isolette example

  Origin system

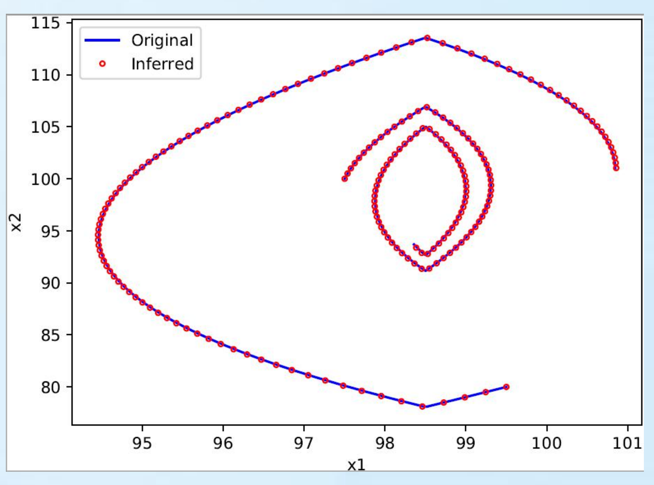
$$\begin{bmatrix} -0.026 & 0 \\ 0.026 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -0.026 & 0 \\ 0.026 & 0 \\ 0 & -1 \end{bmatrix}$$

$$x_1 = 98.5$$

  Inferred system

$$\begin{bmatrix} -2.60000 \times 10^{-2} & -7.93683 \times 10^{-14} \\ 2.60000 \times 10^{-2} & -2.58099 \times 10^{-14} \\ -6.58505 \times 10^{-13} & 1.00000 \end{bmatrix} \quad \begin{bmatrix} -2.60056 \times 10^{-2} & -1.08319 \times 10^{-4} \\ 2.60007 \times 10^{-2} & 1.42407 \times 10^{-5} \\ 4.86417 \times 10^{-4} & -9.90636 \times 10^{-1} \end{bmatrix}$$

$$x_1 - 0.00077x_2 = 98.42313$$



- Other examples