Identification of Finite Lattice Dynamical Systems

Junior Barrera

BIOINFO-USP
University of São Paulo, Brazil
Outline

• Introduction
• Lattice operator representation
• Lattice operator design
• Lattice dynamical systems (LDS)
• A simulator for chain dynamical systems
• LDS identification
• System identification examples
• Conclusion
Introduction
Dynamical Systems

\[
x_1 = y_1
\]

\[
x_2 = y_2
\]

Initial state \( u \)

Final state \( u' \)

Final state \( x_1 = y_1 \)
Sequential Switching Circuits

\[ \phi_1(x[i-5], x[i-4], x[i-3], x[i-2], x[i-1], x[i]) = \overline{x}_1[i-3] \cdot \overline{x}_1[i-2] \cdot \overline{x}_1[i-1] \cdot \overline{x}_2[i-5] \cdot \overline{x}_2[i-3] \cdot \overline{x}_2[i-1] \]

\[ \phi_2(x[i-5], x[i-4], x[i-3], x[i-2], x[i-1], x[i]) = \overline{x}_1[i-4] \cdot \overline{x}_2[i-5] \cdot \overline{x}_2[i-4] \cdot \overline{x}_2[i-3] \cdot \overline{x}_2[i-2] \cdot \overline{x}_2[i-1] \]
Mathematical Morphology

- studies operators between complete lattices, what includes switching functions
- lattice operators are decomposed in terms of simple morphological operators: erosion, dilation, anti-erosion, anti-dilation
- Any lattice operator can be decomposed in a canonical morphological representation
Lattice Dynamical Systems

- We present the notion of Lattice Dynamical System (LDS)
- Give a representation for LDSs, based on canonical morphological representations
- Formalize the problem of statistical identification of LDSs,
Operator Representation
\[ \psi : \text{Fun}[W, L] \rightarrow L \]

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>-2</td>
<td>-2</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
Intervals

- Let $a, b \in \text{Fun}[W,L]$, $a \leq b$ iff $a(x) \leq b(x)$, $x \in W$

  $|W| = 2$

  $\begin{array}{c}
  \text{a} \\
  \text{b}
  \end{array}$

- Interval $[a,b] = \{u \in \text{Fun}[W,L]: a \leq u \leq b\}$

  $\begin{array}{c}
  \text{a} \\
  \text{b}
  \end{array}$
Binary Sup-generating

- Sup-generating operator: \( \lambda_{a,b}(u) = 1 \iff u \in [a, b] \)
Kernel of $\psi$ at $y$: $K(\psi)(y) = \{ u \in \text{Fun}[W,L]: y \leq \psi(u) \}$

$$
\begin{array}{cccccc}
2 & 0 & 1 & 2 & 2 & 2 \\
1 & 0 & 1 & 2 & 2 & 2 \\
0 & -1 & 1 & 2 & 2 & 2 \\
-1 & -1 & 1 & 1 & 1 & 1 \\
-2 & -2 & -1 & -1 & -1 & -1 \\
\end{array}
$$

-2 -1 0 1 2

Kernel

$K(\psi)(-2)$  $K(\psi)(-1)$  $K(\psi)(0)$  $K(\psi)(1)$  $K(\psi)(2)$
Basis of $\psi$ at $y$: $B(\psi)$ is the set of maximal intervals contained in $K(\psi)$.
Canonical Representation

\[
\psi(u) = \bigcup \{ y \in M : \bigcup \{ \lambda_{a,b}(u) : [a,b] \in B(\psi)(y) \} = 1 \}
\]

\[
\psi(-1,-1) = 1
\]
Operator Design
Optimization problem

Design goal is to find a function $\psi_{opt} : \text{Fun}[W, L] \rightarrow L$ with minimum error.

Error (expected loss) of a function:

$$Er(\psi) = E[l(\psi(X), Y)]$$

$X$ is a random function

$Y$ is a random variable

Loss function

$$l : L \times L \rightarrow \mathbb{R}_+$$
Estimation problem

The distribution \( P(X,Y) \) is unknown

\( P(X,Y), \ Er(\psi) \) and \( \psi_{opt} \) should be estimated from realizations of \( X \) and \( Y \).

For \( m > m(\varepsilon, \delta) \) examples

\[
Pr(|Er(\psi) - Er(\psi_{opt})| < \varepsilon) > 1 - \delta
\]

\( \varepsilon, \delta \in (0, 1) \)
The constrained estimation problem

\[ \Psi \]

\[ \psi_{opt} \]

Error

Sample size
Stack filter

- Spatially translation invariant
- Spatially locally defined
- Increasing
- Commutes with threshold
Noise elimination

training images
Apertures

- Spatially translation invariant
- Spatially locally defined
- Range translation invariant
- Range locally defined
Deblurring training images
Resolution Enhancement
Original

Aperture: 3x3x21x51

Linear

Bilinear
Zoom

Original

Aperture: 3x3x21x51

Linear

Bilinear
Independent Constraints

Constraints

Restriction of the operators space

\( K(\psi_{\text{opt}}) \in Q \subseteq P(P(W)) \)

Independent Constraint

Let be \( A, B \subseteq P(W) \) with \( A \subseteq B \):

\( h_\psi(x) = 1 \ \forall \ x \in A \) \& \( h_\psi(x) = 0 \ \forall \ x \notin B \),

\( \forall \psi : K(\psi) \in Q \)
Proposition: if $Q$ is an independent restriction then exist a pair of operators $(\alpha, \beta)$ such that, for any $\psi \in \Psi_w$

$$K(\psi) \in Q \iff \alpha \leq \psi \leq \beta$$

where $K(\alpha) = A$ and $K(\beta) = B$

- All independent constraint is characterized by two operators $\alpha$ and $\beta$
- The pair $(\alpha, \beta)$ is called “Envelope”
Noise Edge Detection

1. **Ground Image**
2. **Noise Addition**
3. **Noisy Image**
4. **Restoration**
5. **Filtered Image**
6. **Edge Detection**
7. **Edge Detected**

- **Direct Edge Detection**
Restoration

a) **Machine design** of the restoration

ψ_{pac} designed by examples

b) **Human-machine design** of the restoration

ψ_{con} = (ψ_{pac} ∩ β) ∪ α

α = δ_{B⊗B}ϵ_{B⊗B}δ_{B}ε_{B} and β = ε_{B⊗B}δ_{B⊗B}ε_{B}δ_{B}

α and β are alternating sequential filters with

P[ α(S) ≤ I ≤ β(S) ] ≈ 1

B is the 3x3 square
**Noise Edge Detection**

- **a)** Machine design over noisy images
  \[ \zeta_{\text{pac}} \text{ designed by examples from noisy images} \]

- **b)** Human design after restoration
  \[ \zeta = I_d - \varepsilon_B \]
  
  - B is the 3x3 square

- **c)** Machine design after restoration
  \[ \zeta_{\text{pac}} \text{ designed by examples from restored images} \]

---

<table>
<thead>
<tr>
<th>Method</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine design over noisy images</td>
<td>0.65 %</td>
</tr>
<tr>
<td>Human design after restoration</td>
<td>0.27 %</td>
</tr>
<tr>
<td>Machine design after restoration</td>
<td>0.24 %</td>
</tr>
</tbody>
</table>
Noise Edge Detection

Machine design over noisy images
Error = 0.65%

Human design after restoration
Error = 0.27%

Machine design after restoration
Error = 0.24%
Multiresolution Constraint

2 variables: $2^2 = 4$

4 variables: $2^4 = 16$

8 variables: $2^8 = 256$
Multiresolution Constraint

\[ D_1 = P(W_1) \]
\[ D_0 = P(W_0) \]
\[ z_i = p_i(x_{i1}, \ldots, x_{i9}), z = p(x), p = (p_1, \ldots, p_9) \]

Let \( \phi : D_1 \rightarrow \{0, 1\} \), it defines the operator \( \Psi_\phi \) on \( D_0 \) by
\[ \Psi_\phi (x) = \phi(p(x)) \]
The operator \( \Psi_\phi \) is constrained by resolution to \( D_1 \)

Equivalence classes defined by
\[ p(x) = p(y) \]
Multiresolution Constraint

The equivalence classes defined by $p$ may be different by the ones defined by $w$. 

\[
D_2 = P(W_2), \quad D_1 = P(W_1), \quad D_0 = P(W_0) \\
x \in D_0, \quad z \in D_1, \quad v \in D_2, \\
z_i = p_i(x_{i,1}, \ldots, x_{i,9}), \quad z = p(x), \\
p = (p_1, \ldots, p_{81}) \\
v_i = w_i(x_{i,1}, \ldots, x_{i,81}), \quad v = w(x), \\
w = (w_1, \ldots, w_9)
\]
Multiresolution Noise

image

noise

image + noise
Dynamical Systems
Finite Lattice Dynamical System

\[ x: T \rightarrow \mathcal{L}^n \quad y: T \rightarrow \mathcal{L}^m \quad x[t] \in \mathcal{L}^n \]

\[ u: T \rightarrow \mathcal{L}^n \]

\[ x[t + 1] = \Phi_t(x[t - N], \ldots, x[t], \ldots, x[t + N], u[t - N], \ldots, u[t], \ldots, u[t + N]) \]

\[ y[t] = \Psi_t(x[t - N], \ldots, x[t], \ldots, x[t + N], u[t - N], \ldots, u[t], \ldots, u[t + N]) \]

\[ S(\Phi_t, \Psi_t) \]

\[ \Phi_t : \mathcal{L}^{2(2N+1)n} \rightarrow \mathcal{L}^n \]

\[ \Psi_t : \mathcal{L}^{2(2N+1)n} \rightarrow \mathcal{L}^m \]
Representation

\[ x_j[t + 1] = \phi_{t,j}(x[t - N], \ldots, x[t], \ldots, x[t + N], u[t - N], \ldots, u[t], \ldots, u[t + N]) \]

\[ y_k[t] = \psi_{t,k}(x[t - N], \ldots, x[t], \ldots, x[t + N], u[t - N], \ldots, u[t], \ldots, u[t + N]) \]

\[ x_j[t] \in \mathcal{L} \]

\[ \phi_{t,j} : \mathcal{L}^{2(2N+1)n} \to \mathcal{L} \]

\[ \psi_{t,k} : \mathcal{L}^{2(2N+1)n} \to \mathcal{L} \]

The component functions have canonical morphological representations
System: input-output

\[ y[t] = S(x[0],...,x[N],...,x[2N])(\Phi_t, \Psi_t)(u[t - N],...,u[t],...,u[t + N]) \]

\[ y = S(x[0],...,x[N],...,x[2N])(\Phi_t, \Psi_t)(u) \]
Filter

\[ u : T \rightarrow \mathcal{L}^n \quad \quad y : T \rightarrow \mathcal{L}^m \]

\[ \Gamma : \mathcal{L}^{(2N+1)n} \rightarrow \mathcal{L}^m \]

\[ y[t] = \Gamma_t(u[t - N], ..., u[t], ..., u[t + N]) \]

For example, processing of motion images.
Input-free systems

\[ x[t + 1] = \Phi_t(x[t - N], ..., x[t], ..., x[t + N]) \]

\[ y[t] = \Psi_t(x[t - N], ..., x[t], ..., x[t + N]) \]
Causal systems

\[
x[t + 1] = \Phi_t(x[t - N], ..., x[t], u[t - N], ..., u[t])
\]

\[
y[t] = \Psi_t(x[t - N], ..., x[t], u[t - N], ..., u[t])
\]
Time translation invariant systems

\[ \Phi : L^2(2N+1)^n \to L^n \]
\[ \Psi : L^2(2N+1)^n \to L^{m^*} \]
\[ S(\Phi, \Psi) \]
\[ \Phi_t = \Phi \]
\[ \Psi_t = \Psi \]
Causal, time translation invariant
A simulator for chain dynamical systems
Simulator Architecture
Functions Representation

![Graphical representation of functions]

- 0
- +1
- -1
System Description


# Function definitions
# ------------------

def f1: [0000000000..1000101010]: 1;
def f2: [0000000000..0111100000]: 1;

# History
# -------------

hist g1: [0 0 0 0 0 0 0 0 0 0 0 0];
hist g2: [0 0 0 0 0 0 0 0 0 0 0 0];

end
• Cell cycle simulation
Oscilador de Período 10: FUNCIONAMENTO GERAL (parte_B-t4A-o10.sim)
Oscilador de Periodo 2: FUNCIONAMENTO GERAL (parte_B-t4A-o2.sim)
Sinal Periodico 7 ligados 1desligado: FUNCIONAMENTO GERAL (parte_B:t4-o8-7of8.sim)
A model for system identification
Model

$S_{opt} (\Phi, \Psi)$

vector of functions
each component function has a basis
System Error

\[ Er(S(\Phi, \Psi)) = E\left[ l(S(x_0, ..., x_N)(\Phi, \Psi)(U[t-N], ..., U[t-1], U[t]), I[t]) \right] \]

\[ l : \mathcal{L}^m \times \mathcal{L}^m \to \mathbb{R}^+ \]
Stationary Conditions

\[ P(x[t - N], ..., x[t - 1], x[t], u[t - N], ..., u[t - 1], u[t], i[t]) = p \]

\[ E_r(S(\Phi, \Psi)) = E\left[l(S(x[0], ..., x[N]), (\Phi, \Psi)(U[t - N], ..., U[t - 1], U[t]), I[t])\right] \]

\[ = \sum_{(x[t - N], ..., x[t], u[t - N], ..., u[t], i[t]) \in \mathcal{L}^{2(N+1)n} \times \mathcal{L}^m} l(S(x[t - N], ..., x[t]), (\Phi, \Psi)(u[t - N], ..., u[t]), i[t]) \times p(x[t - N], ..., x[t], u[t - N], ..., u[t], i[t]) \]
Component Error

\[ E_{r_k} [S(\Phi, \Psi)] = E \left[ l_k \left( S_{(x[0], \ldots, x[N])}(\Phi, \Psi)(U[t - N], \ldots, U[t - 1], U[t])_k, I_k[t] \right) \right]. \]

\[ l_k : \mathcal{L} \times \mathcal{L} \rightarrow \mathbb{R}^+ \]
Additive Loss Function

\[ l = \sum_{k=1}^{m} c_k l_k \quad c_k \in \mathbb{R}^+ \]

\[ Er(S(\Phi, \Psi)) = \sum_{k=1}^{m} Er_k [ S(\Phi, \Psi) ] \]

\[ e_{MAE}(a, b) = \sum_{k=1}^{m} | a_k - b_k | \]

\[ a, b \in \{0,1\}^m \]

\[ e_{MAE} = \sum_{k=1}^{m} e_k_{MAE} \]

\[ e_k_{MAE}(a, b) = |a - b| \]
Independence Condition

- Under additive loss function optimize the system error is equivalent to optimize the system components error.
- The problem of system identification is reduced to a family of problems of lattice operator design.
Identification of Dynamical Systems
A Boolean System

\[ x_1[t + 1] = 1 \iff \begin{cases} 
    x_1[t] = 0 \\
    \text{and} \\
    \left( (x_3[t] = 1 \text{ or } x_3[t - 1] = 1 \text{ or } x_3[t - 2] = 1) \text{ and } \\
    (x_4[t] = 1 \text{ or } x_4[t - 1] = 1 \text{ or } x_4[t - 2] = 1) \right) \\
    \text{or} \\
    \left( x_3[t] = x_3[t - 1] = x_3[t - 2] = x_3[t - 3] = x_3[t - 4] = 0 \text{ and } \\
    x_4[t] = x_4[t - 1] = x_4[t - 2] = x_4[t - 3] = x_4[t - 4] = 0 \right) 
\end{cases} \]
System simulation
System identification: system error
System identification: transition error
Ideal

Est.: 100 training examples

Est.: 500 training examples

Est.: 1500 training examples
Motion Segmentation
Filtering

Color Composition
Watershed

Color Composition
Motion Segmentation
Motion Segmentation
Conclusion
• Presented the notion of Lattice Dynamical System
• Proposed a model for LDS identification
• Under additive condition, system identification reduces to a family of problems of lattice operator design
• Some examples were presented
• This perspective unifies theories such as switching theory, discrete automatic control and reinforcement learning.