Aggregation and Identification of Finite-Valued Networks via Bisimulation

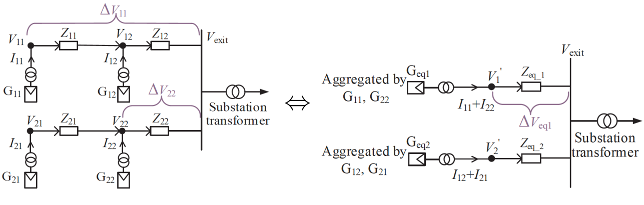
Zhengping Ji

Academy of Mathematics and Systems Science, Chinese Academy of Sciences

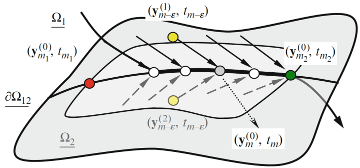
January 9, 2025
Kunming, China

Background: Externally Equivalent Systems

In systems engineering, externally equivalent states can be viewed as identical, which may simplify the system structures.



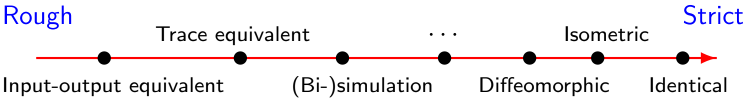
(a) Equivalent circuits in a photovoltaic power plant [4]

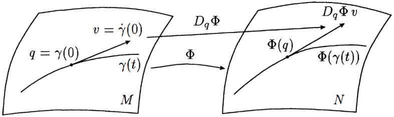


(b) Synchronized systems [7]

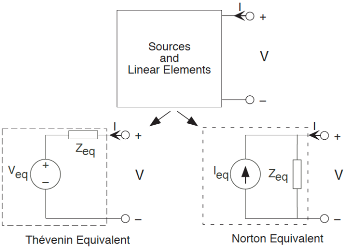
System Equivalence: Aims and Difficulties

We hope to find an appropriate way to characterize the similarity or equivalence between systems, neither too strict nor too rough.





(c) Diffeomorphic equivalence: too strict



(d) I/O equivalence: too rough [6]

Background: Model Reduction via Equivalence

Using equivalence, we may simplify the analysis of systems and achieve lower computational complexity.

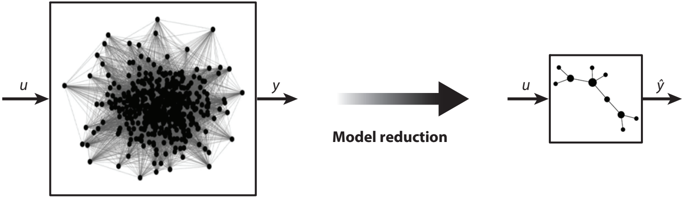


Figure 2: Model reduction of control networks

When the inner dynamic rule is known, the model may be reduced via equivalence; conversely, when only the input-output transition is known, one may construct equivalent systems to identify or realize the given I/O relation.

Background: System Identification and Realization

Given the input-output data of a system, we would like to reconstruct its inner state space.

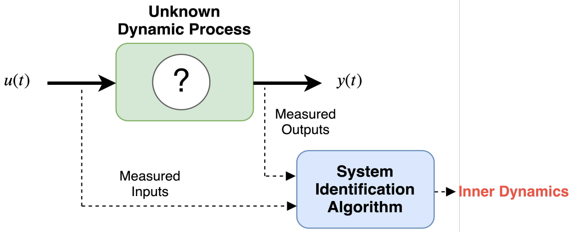


Figure 3: Identification of systems

Background: System Identification and Realization

Given the input-output data of a system, we would like to reconstruct its inner state space.

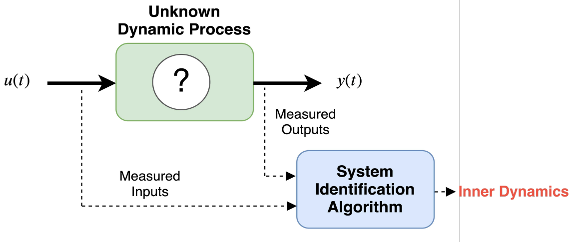


Figure 3: Identification of systems

Question:
What if the system is not observable? What if the inner space is of unknown dimension?

Outline of the Talk

- Basic notions and examples of bisimulation
- Application 1: aggregated (bi-)simulation of Boolean networks
- Application 2: identification of Boolean networks

Basic Setting: Discrete Transition Systems

Definition 1 (Transition Systems)

A tuple $T = (X, U, \Sigma, O, h)$ is called a transition system, where

(i) X is the set of states,

(ii) U is the set of inputs (controls or actions),

(iii) $\Sigma : X \times U \to 2^X$ is a transition mapping,

(iv) O is the observations,

(v) $h : X \to O$: observation mapping.

If $|\Sigma(x, u)| \leq 1$, T is said to be deterministic.

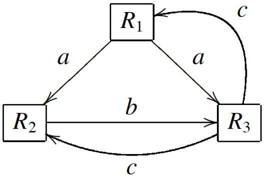


Figure 4: A transition system

Bisimulation of Transition Systems

Definition 2 (Simulation and Bisimulation)

Consider two transition systems $T_i = (X_i, U_i, \Sigma_i, O_i, h_i)$, $i = 1, 2$. If there exists a relation $\mathcal{R} \subset X_1 \times X_2$, s.t.

- $\forall x_1 \in X_1, \exists x_2 \in X_2$, s.t. $(x_1, x_2) \in \mathcal{R}$;
- $\forall (x_1, x_2) \in \mathcal{R}$, $\forall u_1$, $\exists u_2$, s.t. $(\Sigma_1(x_1, u_1), \Sigma_2(x_2, u_2)) \cap \mathcal{R} \neq \varnothing$,

then $\mathcal{R}$ is called a simulation of T_1 by T_2. If further $\mathcal{R}^{-1} \subset X_2 \times X_1$ is a simulation of T_2 by T_1, then $\mathcal{R}$ is called a bisimulation of T_1, T_2.

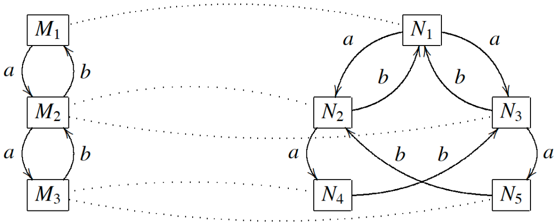


Figure 5: Bisimulated systems [8]

Quotient Systems of Transition Systems

A special type of the simulation is the quotient system under observational equivalence.

Definition 3 (Quotient Systems)

Let $T = (X, U, \Sigma, O, h)$ be a transition system.
$T/\sim := (X/\sim, U, \Sigma_\sim, O, h_\sim)$ is called the quotient system of T under observational equivalence, where

(i) $X/\sim = \{\bar{x} | \; x \in X\}$ is the set of observational equivalence classes, i.e. $x_1 \sim x_2 \Leftrightarrow h(x_1) = h(x_2)$. $\bar{x} := \{y | y \sim x\}$.

(ii) U: (original) set of inputs.

(iii) $\Sigma_\sim : X/\sim \times U \to 2^{X/\sim}$ defined as follows: Assume $\bar{x}_i$, $\bar{x}_j \in X/\sim$. $\bar{x}_j \in \Sigma_\sim(\bar{x}_i, u)$, if and only if $\exists x_i \in \bar{x}_i$, $x_j \in \bar{x}_j$, s.t. $x_j \in \Sigma(x_i, u)$.

(iv) O: (original) set of observations.

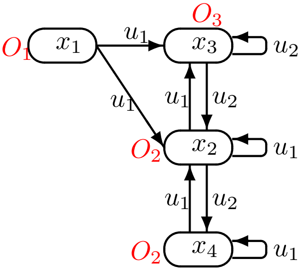
(v) $h_\sim : X/\sim \to O$ is defined as $h_\sim(\bar{x}) := h(x)$, $\forall \bar{x} \in X/\sim$.

An Example of Quotient System

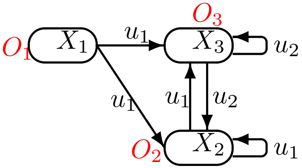
Consider a transition system $T = (X, U, \Sigma, O, h)$ where

$$X = \{x_1, x_2, x_3, x_4\},\ U = \{u_1, u_2\},\ O = \{O_1, O_2, O_3\},$$
$$\Sigma(x_1, u_1) = \{x_2, x_3\},\ \Sigma(x_2, u_1) = \{x_2, x_3\},\ \Sigma(x_2, u_2) = \{x_4\},$$
$$\Sigma(x_3, \sigma_2) = \{x_2, x_3\},\ \Sigma(x_4, \sigma_1) = \{x_2, x_4\},$$
$$h(x_1) = O_1,\ h(x_2) = h(x_4) = O_2,\ h(x_3) = O_3,$$

Construct its quotient system, as depicted in figures 4(a)4(b):



(a) A transition system T



(b) Its quotient system $T/\sim$

Bisimulation From Observational Equivalence

Definition 4

Consider a transition system $T = (X, U, \Sigma, O, h)$. Assume $x_1 \sim x_2$ are observational equivalent, if $\forall u \in U$ and $x_1' \in \Sigma(x_1, u)$, there exists an $x_2' \in \Sigma(x_2, u)$ such that $x_1' \sim x_2'$, then we say $x_1 \approx x_2$. If $\forall x_1, x_2$, $x_1 \sim x_2 \Rightarrow x_1 \approx x_2$, then $T/\sim$ is called a bisimulation of T, denoted by $T/\approx$.

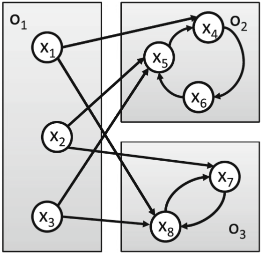
Obviously, the above definition coincides with the general definition of bisimulation by taking $\mathcal{R} := \{(x, \bar{x}) | x \in X\} \subset T \times T/\sim$.

Quotient Systems as Bisimulations

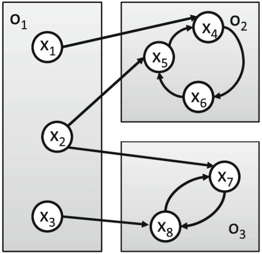
Theorem 5 [3]

The observational equivalence $\sim$ is a bisimulation for a finite transition system $T = (X, U, \Sigma, O, h)$ if and only if $\forall \bar{x} \in X/\sim$, $\forall u \in U$, $P(\bar{x}, u)$ is either empty or a finite union of equivalent classes, where

$$P(\bar{x}, u) := \{y \in X | \exists x' \sim x,\ x' \in \Sigma(y, u)\}.$$

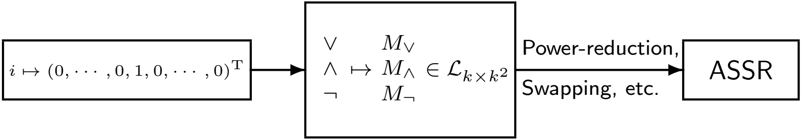


(a) A bisimulation

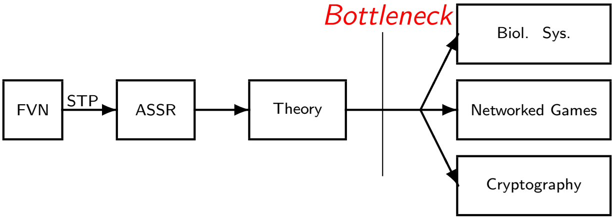


(b) Not a bisimulation

Aggregation of Boolean Networks: Motivation



A chief bottleneck of algebraic state space representation (ASSR) approach to finite-valued networks (FVN) is the curse of dimension.



Review: ASSR of Boolean Control Networks

Consider a Boolean control network

$$\begin{cases} x_i(t+1) = f_i(x_1(t), \cdots, x_n(t); u_1(t), \cdots, u_m(t)), \\ y_k(t) = g_k(x_1(t), \cdots, x_n(t)) \end{cases} \quad (1)$$

where $\{x_i\}_{i=1}^n$, $\{y_k\}_{k=1}^p$, $\{u_j\}_{j=1}^m \subset \mathcal{D}_2$. Denote by L_i, H_k the structure matrices of the functions f_i, g_k, $i = 1, \cdots, n$, $k = 1, \cdots, p$ respectively, let $x := \ltimes_{i=1}^n x_i$, $u := \ltimes_{j=1}^m u_j$, $y := \ltimes_{k=1}^p y_k$, then the ASSR of (1) is

$$\begin{cases} x(t+1) = Lu(t)x(t), \\ y(t) = H(t)x(t), \end{cases} \quad (2)$$

where $L := L_1 * \cdots * L_n$, $H := H_i * \cdots * H_p$.

Generalization: ASSR of Finite-Valued Transition Systems

Proposition 6

Consider a finite-valued transition system $T = (X, U, \Sigma, O, h)$ with $|X| = n$, $|U| = m$, $|O| = \ell$. Using vector form expressions that $X \sim \Delta_n$, $U \sim \Delta_m$, and $O \sim \Delta_\ell$, the dynamics of T can be expressed into its ASSR as

$$\begin{cases} x(t+1) = Lu(t)x(t), \\ y(t) = Hx(t), \end{cases} \tag{3}$$

where $x(t) \in \mathcal{B}^n$, $u(t) \in \Delta_m$, $y(t) \in \mathcal{B}^\ell$ and $L \in \mathcal{B}_{n \times mn}$, $H \in \mathcal{L}_{\ell \times n}$.

ASSR of Quotient Networks as Transition Systems

Theorem 7

Consider a transition system

$$\begin{cases} x(t+1) = Lu(t)x(t), \\ y(t) = Hx(t), \end{cases} \tag{4}$$

where $x(t)$, $y(t) = \mathcal{B}^p$ are Boolean vectors, $u(t) \in \Delta_m$ is a logical vector, $L \in \mathcal{B}_{n\times mn}$, $H \in \mathcal{L}_{p\times n}$. Then the quotient system is

$$\begin{cases} X(t+1) = L_q u(t)X(t), \\ y(t) = H_q X(t), \end{cases} \tag{5}$$

where $X_i \in X/\sim$ is the equivalence class of y_i, $i \in [1, q]$,

$$L_q = H \times_{\mathcal{B}} L \times_{\mathcal{B}} (I_m \otimes H^T), \quad H_q = I_p. \tag{6}$$

where $\times_{\mathcal{B}}$ is the Boolean product of matrices.

Partition and Aggregation of BCNs

Proposition 8

Assume $A \subset N$ is a block of nodes with $\{x_{i_1}, \cdots, x_{i_\alpha}\}$ as its block inputs, and $\{\{x_{j_1}, \cdots, x_{j_\beta}\}$ as its block outputs. Then the dynamic subnetwork of A can be expressed as a controlled network Σ_A with block control $v_\ell := x_{i_\ell}$, $\ell \in [1, \alpha]$, and block output $y_k := x_{j_k}$, $k \in [1, \beta]$, replacing block A in Σ by this block control system Σ_A does not affect the dynamics of the rest part of Σ.

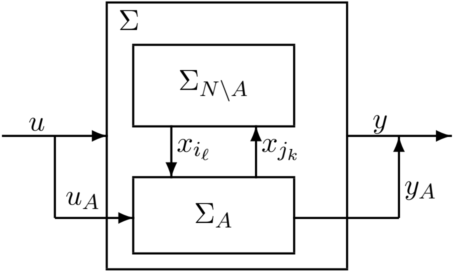


Figure 6: Partition of a BN

An Example

Denote by N the set of nodes of Σ, and let $A = \{x_{i+1}, x_{i+2}, \cdots, x_{i+\mu}\} \subset N$, $\mu > 1$ be a subset of nodes.

The dynamic equations of A, denoted by Σ_A, are

$$\begin{cases} x_{i+1}(t+1) = [(x_i(t)\bar{\nabla} x_{i+1}(t)) \wedge u(t)] \\ \qquad\qquad \vee [(x_i(t) \leftrightarrow x_{i+1}(t)) \wedge \neg u(t)], \\ x_{i+2}(t+1) = [(x_{i+1}(t)\bar{\nabla} x_{i+2}(t)) \wedge u(t)] \\ \qquad\qquad \vee [(x_{i+1}(t) \leftrightarrow x_{i+2}(t)) \wedge \neg u(t)], \\ \vdots \\ x_{i+\mu}(t+1) = [(x_{i+\mu-1}(t)\bar{\nabla} x_{i+\mu}(t)) \wedge u(t)] \\ \qquad\qquad \vee [(x_{i+\mu-1}(t) \leftrightarrow x_{i+\mu}(t)) \wedge \neg u(t)]. \end{cases}$$
$$y(t) = x_{i+\mu}(t). \tag{7}$$

where $\wedge, \vee, \neg$ are the conjunction, disjunction, and negation operators respectively.

An Example (Cont'd)

By Theorem 7, the quotient system $\Sigma_A/\sim$ is obtained as

$$y(t+1) = \delta_2[2,1,1,2,1,2,2,1]u(t)v(t)y(t), \qquad (8)$$

where $y(t) = x_{i+\mu}(t)$, $v(t) = x_i(t)$.
Obviously $\Sigma_A/\sim$ is a deterministic system.

The dimension of the subnetwork is reduced from 2^μ to 2.

An Example (Cont'd)

By Theorem 7, the quotient system $\Sigma_A/\sim$ is obtained as

$$y(t+1) = \delta_2[2,1,1,2,1,2,2,1]u(t)v(t)y(t), \tag{8}$$

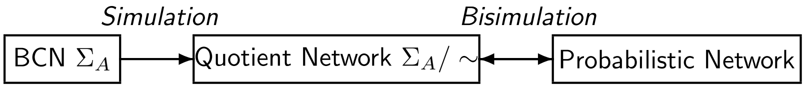
where $y(t) = x_{i+\mu}(t)$, $v(t) = x_i(t)$.
Obviously $\Sigma_A/\sim$ is a deterministic system.

The dimension of the subnetwork is reduced from 2^μ to 2.

Question:
What if the quotient system is not deterministic?

Aggregated Simulation of Boolean Networks

We execute an aggregated simulation of BCN in two steps.



Proposition 9

Let Σ be a networked system with network graph (N, E), $A \subset N$ is an aggregate-able subset. If $\Sigma_A/\sim = \Sigma_A/\approx$ is a bisimulation, then the aggregation does not affect the dynamics of the overall system.

Approximate Bisimulation of Aggregated BCN

Consider the aggregated ASSR with respect to the block A. Set

$$M_A := H_A L_A (I_{k^{m+\alpha}} \otimes H_A^T) = (m_{i,j}) \in \mathcal{M}_{\xi \times \eta}.$$

Denote $m_j := \sum_{i=1}^{\xi} m_{i,j}$, $j \in [1, \eta]$, *define a probabilistic system, denoted by* Σ_A^P, *as follows*:

$$y(t+1) = M^{i_1, i_2, \cdots, i_\eta} u(t) v(t) y(t), \quad i_j \in [1, \xi],\ j \in [1, \eta],$$

where

$$M^{i_1, i_2, \cdots, i_\eta} = \delta_\xi [i_1, i_2, \cdots, i_\eta],$$

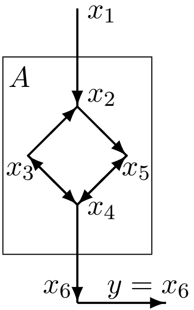
with probability

$$p_{i_1, i_2, \cdots, i_\eta} = \frac{\prod_{j=1}^{\eta} m_{i_j, j}}{\prod_{j=1}^{\eta} m_j}.$$

Σ_A^P is called the **approximate bisimulation** of the block A.

An Example

The following system illustrates the two-step aggregated simulation of BCN. Let $A := \{x_2, x_3, x_4, x_5\}$.



$$\begin{cases} x_1(t+1) = \neg x_1(t), \\ x_2(t+1) = x_1(t) \wedge x_3(t), \\ x_3(t+1) = x_3(t) \vee x_4(t), \\ x_4(t+1) = x_3(t) \rightarrow x_5(t), \\ x_5(t+1) = x_2(t) \bar{\vee} x_4(t), \\ x_6(t+1) = x_4(t) \leftrightarrow x_6(t) \end{cases}$$

$$y(t) = x_6(t)$$

Figure 7: An Aggregated BN

ASSR of A: $\begin{cases} z(t+1) = L_A v(t) z(t) = \delta_{16}[2, \cdots, 14] v(t) \ltimes_{i=1}^{4} z_i(t) \\ y(t) = H z(t) = \delta_2[1, 1, \cdots, 2, 2] \ltimes_{i=1}^{4} z_i(t), \end{cases}$

where $z_1 = x_2$, $z_2 = x_3$, $z_3 = x_4$, $z_4 = x_5$, $v = x_1$, $y = x_4$.

An Example (Cont'd)

Compressing A into one single node, we derive the quotient system $\Sigma_A/\sim$ with the following ASSR:

$$\begin{cases} w(t+1) = Lv(t)w(t), \\ y(t) = w(t), \end{cases}$$

where $L = H \times_{\mathcal{B}} L_A \times_{\mathcal{B}} (I_2 \otimes H^T) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$.

Next, we proceed to construct the approximate identification of the block.

An Example (Cont'd)

Recall the system in Figure 4. The weighted structure matrix of $\Sigma_A/\sim$ can be calculated as follows:

$$\begin{aligned} M_A &= H L_A (I_2 \otimes H^T) \\ &= \begin{bmatrix} 6 & 6 & 6 & 6 \\ 2 & 2 & 2 & 2 \end{bmatrix}. \end{aligned} \tag{9}$$

Then the simulation-aggregation is using the following probabilistic network Σ_A^P to replace A:

$$z(t+1) = L_A^P v(t) z(t), \tag{10}$$

where

$$L_A^P = \begin{bmatrix} 2/3 & 2/3 & 2/3 & 2/3 \\ 1/3 & 1/3 & 1/3 & 1/3 \end{bmatrix}.$$

Outlook: Aggregation-Simulation of Large-Scale BCNs

Using the aggregation-(bi-)simulation method, we may decompose a BCN into probabilistic blocks and analyse them separately (topological structures, control properties, etc.), which is a trade-off between computational load and precision of approximation.

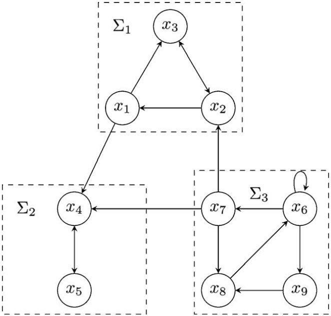


Figure 8: Aggregation of a BN [9]

Outlook: Aggregation-Simulation of Large-Scale BCNs

Possible approaches for designing aggregate-able blocks:

- Balancing method
- Pinning control
- Invariant spaces and minimal bisimulation

Reduction and Realization

In the reduction (aggregation) problem, the inner dynamics is known; in the realization problem, the only available data is the input-output relation.

Reduction and Realization

In the reduction (aggregation) problem, the inner dynamics is known; in the realization problem, the only available data is the input-output relation.

Question:
How many nodes do we need to reconstruct an input-output transition rule?

Reduction and Realization

In the reduction (aggregation) problem, the inner dynamics is known; in the realization problem, the only available data is the input-output relation.

Question:
How many nodes do we need to reconstruct an input-output transition rule?

Proposition 10

To construct an identification of a network with p output nodes, one needs a network of at most $2p$ state nodes.

The proof follows from solving the equation
$H \times_{\mathcal{B}} L_A \times_{\mathcal{B}} (I_2 \otimes H^T) = \mathbf{1}_{k^p \times k^{m+p}}$.

Identification Algorithm

Following Proposition 10, we propose an identification algorithm to reconstruct structure matrices from given input-output data.

Given a series $S := ((u_0, y_0), \cdots, (u_T, y_T), \cdots)$, where $\{u_t\}_{t \geqslant 0} \subset \Delta_{k^m}$, $\{y_t\}_{t \geqslant 0} \subset \Delta_{k^p}$.

- **Step 0**. For $\forall q, s \in [1, k^p]$ and $\forall r \in [1, k^m]$, set $\ell_s^r := 1$, $\alpha_{qs}^r := 0$. Set $L := \mathbf{0}_{k^m} \otimes I_{k^{2p}}$ and denote by $L = [L_1, \cdots, L_{k^m}]$, where $L_i \in \mathcal{L}_{k^{2p} \times k^{2p}}$ is the i-th block of L, $i \in [1, k^m]$.
- **Step t>0**. Consider the case that $(u_{t-1}, y_{t-1}) = (\delta_{k^m}^{i_0}, \delta_{k^m}^{j_0})$ and $(u_t, y_t) = (\delta_{k^m}^{i_1}, \delta_{k^m}^{j_1})$.
  If $\alpha_{j_0 j_1}^{i_0} \neq 0$ or $\ell_{j_0}^{i_0} = k^p$, go to **Step $t+1$**.
  Else, for $\forall j \in [(j_0 - 1)k^p + \ell_{j_0}^{i_0}, j_0 k^p]$, set $\mathrm{Col}_j L_{i_0} := \delta_{k^{2p}}^{j_1 k^p}$ and $\ell_{j_0}^{i_0} := \ell_{j_0}^{i_0} + 1$, $\alpha_{j_0 j_1}^{i_0} := \alpha_{j_0 j_1}^{i_0} + 1$, go to **Step $t+1$**. (If the sequence is of finite length T, stop at **Step T**.)

Realization of the Network

The identification (realization) algorithm is described in the following figure.

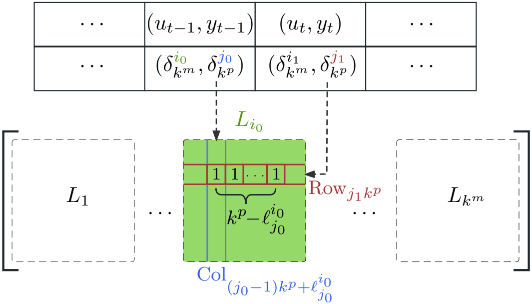


Figure 9: Illustration of the identification algorithm

An Example

Consider the following series of input-output data of length $T = 20$, with one Boolean input and two Boolean outputs.

$$(\delta_2^1, \delta_4^4), (\delta_2^1, \delta_4^2), (\delta_2^2, \delta_4^3), (\delta_2^2, \delta_4^2), (\delta_2^2, \delta_4^4), (\delta_2^2, \delta_4^2), (\delta_2^2, \delta_4^1),$$
$$(\delta_2^1, \delta_4^3), (\delta_2^1, \delta_4^4), (\delta_2^1, \delta_4^1), (\delta_2^2, \delta_4^1), (\delta_2^2, \delta_4^3), (\delta_2^2, \delta_4^1), (\delta_2^1, \delta_4^2),$$
$$(\delta_2^1, \delta_4^3), (\delta_2^1, \delta_4^4), (\delta_2^2, \delta_4^4), (\delta_2^1, \delta_4^2), (\delta_2^1, \delta_4^1), (\delta_2^2, \delta_4^2), \cdots$$

Applying the identification algorithm, we construct a Boolean network of one input, four states, and two outputs, with ASSR as (2) where the structure matrices are

$$\begin{aligned} L = \delta_{16} & [4, 8, 8, 8, 12, 4, 4, 4, 16, 16, 16, 16, 8, 4, 16, \\ & 16, 12, 8, 8, 8, 16, 4, 4, 4, 8, 4, 4, 4, 8, 8, 8, 8], \\ H &= I_4 \otimes \mathbf{1}_4. \end{aligned}$$

An Example

The transition matrix of the output nodes according to the given data is

$$\tilde{L} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}. \tag{11}$$

One can check that $H \times_{\mathcal{B}} L \times_{\mathcal{B}} (I_2 \otimes H^{\mathrm{T}}) = \tilde{L}$, that is to say, the transition system defined by $\tilde{L}$ is indeed generated from the network defined by L, H under observational equivalence.

An Example

The transition matrix of the output nodes according to the given data is

$$\tilde{L} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}. \tag{11}$$

One can check that $H \times_{\mathcal{B}} L \times_{\mathcal{B}} (I_2 \otimes H^{\mathrm{T}}) = \tilde{L}$, that is to say, the transition system defined by $\tilde{L}$ is indeed generated from the network defined by L, H under observational equivalence.

Question:
How to make the identification algorithm more precise?

Probabilistic Identification Algorithm

Given a sequence

$$S := ((u_0, y_0), \cdots, (u_T, y_T), \cdots),$$

where $\{u_t\}_{t \geqslant 0} \subset \Delta_{k^m}$, $\{y_t\}_{t \geqslant 0} \subset \Delta_{k^p}$. Choose an integer $d \geqslant p$.

- **Step 0**. Set $N^i_{j\ell} = 1$, $i \in [1, k^m]$, $j, \ell \in [1, k^p]$. Set $L := \mathbf{0}_{k^m} \otimes I_{k^{d+p}}$, and denote $L = [L_1, \cdots, L_{k^m}]$, where $L_i \in \mathcal{L}_{k^{d+p} \times k^{d+p}}$ is the i-th block of L, $i = 1, \cdots, k^m$.
- **Step t>0**. Assume $(u_{t-1}, y_{t-1}) = (\delta^{i_0}_{k^m}, \delta^{j_0}_{k^p})$, and $(u_t, y_t) = (\delta^{i_1}_{k^m}, \delta^{j_1}_{k^p})$.
  Set $N^{i_0}_{j_0 j_1} := N^{i_0}_{j_0 j_1} + 1$, $S^i_j := \left\{\ell \in [1, k^p] \mid N^i_{j\ell} > 1\right\}$.
  Assume that $S^{i_0}_{j_0} = \{\ell_1, \cdots, \ell_q\}$, $q \in [1, k^p]$.
  For $\forall i \in [1, k^m]$, $\forall j \in [1, k^p]$, $\forall \ell \in S^i_j$, set

$$\beta^i_{j\ell} := \varphi\left(k^d \frac{N^i_{j\ell}}{\sum_{s \in S^i_j} N^i_{js}}\right),$$

  where $\varphi(\cdot)$ is the round down function.

Identification Algorithm

- Let $r_j := (j-1)k^d$, $j \in [1, k^p]$. Set

$$\begin{aligned}
&\mathrm{Col}_{s_1} L_{i_0} := \delta_{k^{d+p}}^{\ell_1 k^d}, \\
&\forall s_1 \in \left[r_{j_0}+1, r_{j_0}+\beta_{j_0 \ell_1}^{i_0}\right], \\
&\mathrm{Col}_{s_2} L_{i_0} := \delta_{k^{d+p}}^{\ell_2 k^d}, \\
&\forall s_2 \in \left[r_{j_0}+\beta_{j_0 \ell_1}^{i_0}+1, r_{j_0}+\beta_{j_0 \ell_1}^{i_0}+\beta_{j_0 \ell_2}^{i_0}\right], \\
&\vdots \\
&\mathrm{Col}_{s_q} L_{i_0} := \delta_{k^{d+p}}^{\ell_q k^d}, \\
&\forall s_q \in \left[r_{j_0}+\sum_{t=0}^{q-1} \beta_{j_0 \ell_t}^{i_0}+1, j_0 k^d\right],
\end{aligned}$$

then go to **Step $t+1$**.

- If the series is of finite length T, stop at **Step T**.

Probabilistic Realization of Finite-Valued Networks

The probabilistic identification (realization) algorithm is described in the following figure.

$\cdots$	(u_{t-1}, y_{t-1})	(u_t, y_t)	$\cdots$
$\cdots$	$(\delta_{k^m}^{i_0}, \delta_{k^p}^{j_0})$	$(\delta_{k^m}^{i_1}, \delta_{k^p}^{j_1})$	$\cdots$

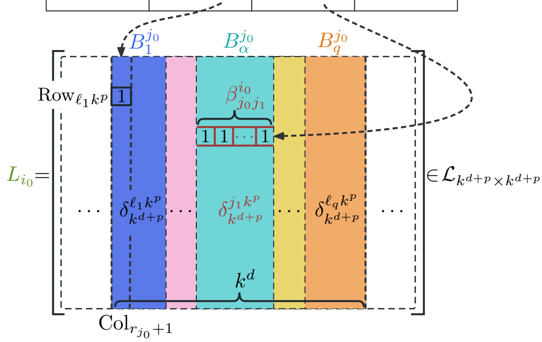


Figure 10: Illustration of the identification algorithm

An Example

Consider the input-output sequence in the previous example. Applying the identification algorithm, one will solve the structure matrices of the 4-state, 1-input, 2-output network as

$$\begin{aligned} L = \delta_{16}\ &[4,4,8,8,12,12,4,4,16,16,16,16,8,4,16,16,\\ &12,12,8,8,16,16,4,4,8,8,4,4,8,8,8,8];\\ H &= I_4 \otimes \mathbf{1}_4. \end{aligned}$$

Calculating the approximate bisimulation of this system yields

$$\tilde{L}' = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & \frac{1}{4} & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{4} & \frac{1}{2} & 0 & \frac{1}{2} & 1 \\ 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 1 & \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \end{bmatrix}.$$

One can see that this transition matrix coincides with the frequency of different transitions appearing in the given sequence S; with the same accessibility property as the matrix (11).

Limit of the Approximation

Let S be an input-output sequence of length t from a k-valued network Σ_0 with p outputs and m inputs. Denote by Σ the approximate simulation of Σ_0. Denote by Σ_S the network constructed from S following the probabilistic identification algorithm, of $d+p$ inner state variables, and let Σ_t^d be the approximate simulation of Σ_S.

Theorem 11

For $u \in U$, $i, j \in X$, denote by $n_i^u(t)$ the frequency of input-output pair (u, i) in S, denote by $p_{i,j}^u(d,t)$ the probability of transition in Σ_t^d from output i to output j under input u, and $p_{i,j}^u$ the probability of the same transition in Σ. If $\lim_{t\to\infty} n_i^u(t) = \infty$, then $p_{i,j}^u(d_1,t) \leqslant p_{i,j}^u(d_2,t)$ for all $d_1 > d_2$, and

$$\lim_{d\to\infty, t\to\infty} p_{i,j}^u(d,t) = p_{i,j}^u.$$

Meaning: the reconstructed network converges to the approximate bisimulation of the original system.

Conclusion

Main contribution of our work:

1. Model reduction of large-scale networks via observational equivalence;
2. Identification and realization of the networks with minimal node sets.

Meanwhile, the bisimulation approach has application in switched systems and continuous-time multi-agent systems.

Perspective: Switched Systems

Consider a hybrid linear system

$$\xi(t+1) = A_{y(t)}\xi(t) + B_{y(t)}\eta(t), \tag{12}$$

where $\xi(t) \in \mathbb{R}^n$ is the state, $\eta(t) \in \mathbb{R}^m$ is the control, the switching signal $y(t)$ is generated by logical control system (2).

Assume $\mathbb{R}^n = \mathrm{Im}(A_1) \oplus \cdots \oplus \mathrm{Im}(A_p)$, $\mathrm{rank}(B_i) = \mathrm{rank}(A_i)$, $\mathrm{Im}(B_i) = \mathrm{Im}(A_i)$, $i = 1, \cdots, p$. We consider the reachability of two given points $x, y \in \mathbb{R}^n$ with respect to the above system.

Proposition 12

$\forall x, y \in \mathbb{R}^n$. Assume $x \in \mathrm{Im}(A_i)$, $y \in \mathrm{Im}(A_j)$, then $\exists T > 0$, a set of switching and controls $\{u(0), \cdots, u(T)\}$ driving a trajectory of (2) from x to y, if and only if, $\tilde{L}_{i,j} \neq 0$.

Bisimulation View of Continuous and Discrete Transitions

Continuous-time nonlinear systems

$$\begin{cases} \dot{x}(t) = f(x(t)) + \sum_{i=1}^{m} u^i(t) g_i(x(t)) \\ y(t) = h(x(t)) \end{cases}$$

$$\Downarrow$$

$$\dot{\tilde{y}}(t) = M\tilde{y}(t) + Fu(t)$$

Finite transition systems

$$\begin{cases} x(t+1) = Lx(t)u(t) \\ y(t) = Hx(t) \end{cases}$$

$$\Downarrow$$

$$\tilde{y}(t+1) = M\tilde{y}(t)u(t)$$

$(x, \tilde{y})$ - Bisimulation;
$\mathrm{Span}\{H\}$ - Invariant subspace;
$\tilde{y}(t)$ - Transition in quotients.

Perspective: Bisimulation of Continuous-Time Systems

Quotient representation gives rise to the observer realization of the control systems.

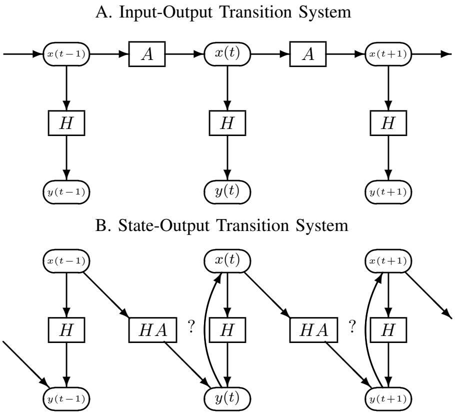


Figure 11: S-System with I-O vs SO-System

Further Studies

We propose the following topics for future work:

1. Analysis of switched systems via transition representation;
2. Ensemble control of large-scale networks via (bi-)simulation;
3. Reduction of finite-valued networks by minimal bisimulation.

References

- Z. Ji, X. Zhang, and D. Cheng, Aggregated (bi-)simulation and identification of finite-valued networks, *IEEE Transactions on Automatic Control*, (2025). https://ieeexplore.ieee.org/document/10771692
- D. Cheng, X. Zhang, and Z. Ji, Transition system representation of Boolean control networks, *Unmanned Systems*, 12 (2023), no. 2, 239-247.
- C. Belta, B. Yordanov, and E. A. Gol, *Formal methods for discrete-time dynamical systems*. Cham: Springer International Publishing, 2017.
- P. Chao, W. Li, X. Liang, Y. Shuai, F. Sun, and Y. Ge, A comprehensive review on dynamic equivalent modeling of large photovoltaic power plants. Solar Energy, 210 (2020), 87-100.
- X. Zhang, M. Meng, and Z. Ji, Analysis of discrete-time switched linear systems under logic dynamic switching, *IEEE Transactions on Neural Networks and Learning Systems*, (2024). https://ieeexplore.ieee.org/document/10744584
- D. H. Johnson, Origins of the equivalent circuit concept: the current-source equivalent. Proceedings of the IEEE, 91 (2003), no. 5, 817-821.
- A. C. J. Luo, Dynamical system synchronization. New York: Springer, 2013.
- D. Sangiorgi, Introduction to bisimulation and coinduction. Cambridge University Press, 2011.
- Y. Zhao, J. Kim, M. Filippone, Aggregation algorithm towards large-scale Boolean network analysis, IEEE Transactions on Automatic Control, 58 (2013), no. 8, 1976-1985.
- S. Zhu, J. Cao, L. Lin, J. Lam, and S. I. Azuma, Towards stabilizable large-scale Boolean networks by controlling the minimal set of nodes, *IEEE Transactions on Automatic Control*, 69 (2024), no. 1, 174-188.

Thanks for your attention!