Synchronous Dynamical Systems on Directed Acyclic Graphs (DAGs): Complexity and Algorithms

Daniel J. Rosenkrantz [1,2], Madhav V. Marathe [1], S. S. Ravi [1,2], Richard E. Stearns [1,2]

[1]University of Virginia and [2]University at Albany – State University of New York

(Poster Presented at AAAI-2021)

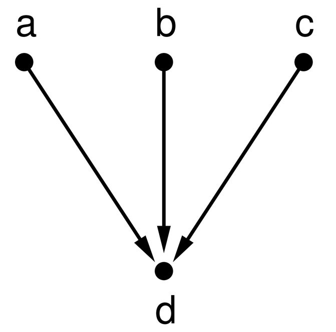
Dynamical Systems: Basics

A **Discrete Dynamical System on a DAG** $\mathcal{S}$ has:

- An underlying DAG $G(V, E)$.
- **Nodes:** Agents in the system.
- **Edges:** Permissible local interactions.
- State values for nodes from a finite domain $\mathbb{B}$. (Here, $\mathbb{B} = \{0, 1\}$.)
- A **local transition function** for each node.
- **Update mechanism: synchronous**.

Notation: DAG-SyDS (Synchronous Dynamical System on a DAG)

Local Transition Function:



Local function f_d:

- Inputs: States of a, b, c and d.
- Output: Next state of d.

- The only input to the local function f_a is the state of a.
- A similar comment applies to the local functions f_b and f_c.

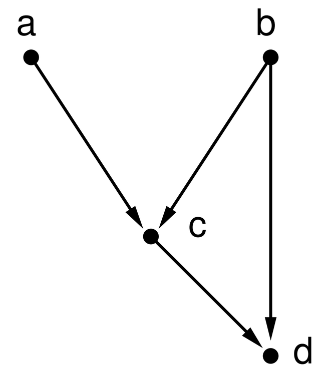
Note: The local functions are **deterministic**.

Previous Work (Brief)

- Hardness of Reachability and Convergence for undirected graphs (e.g., [Barrett et al. 2006, Rosenkrantz et al. 2018]).
- Hardness results for directed graphs [Ogihara & Uchizawa: 2017 & 2020].
- Hardness results for Hopfield neural nets (e.g., [Orponen 1994]) and Petri nets (e.g., [Esparza & Nielsen 1994]).
- Efficient algorithm for Reachability for DAG-SyDSs with **bi-threshold** functions ([Kuhlman et al. 2013]).
- Efficient algorithm for controlling a SyDS on a **directed tree** so that it reaches a desirable configuration ([Akutsu et al. 2007]).

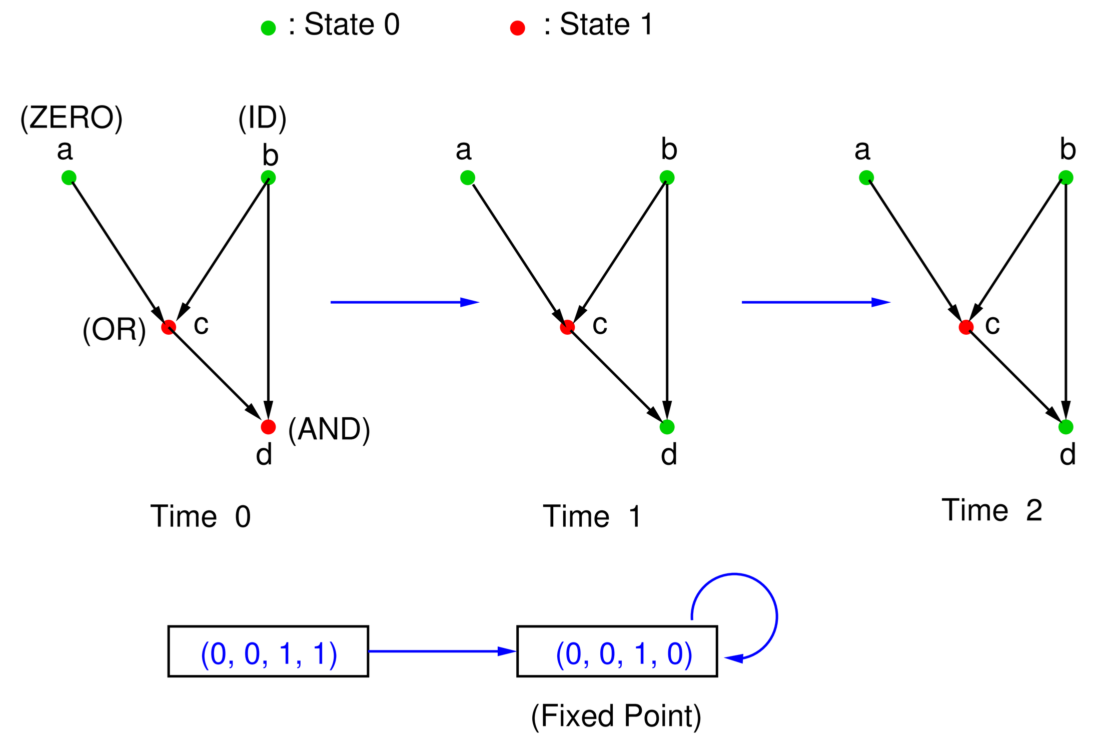
Examples

DAG-SyDS Example:



- $\mathbb{B} = \{0, 1\}$
- f_a : Zero function
- f_b : Identity function
- f_c : OR function
- f_d : AND function

Time Evolution of the Above DAG-SyDS:



Chistikov et al. (AAAI-2020)

- SyDSs on directed graphs in the context of diffusion of opinions.
- Each local function is the **majority** function: an agent changes her {0,1} opinion only when a majority of her neighbors have a different opinion.
- The Convergence problem is efficiently solvable for DAG-SyDSs regardless of local functions.
- Convergence and Convergence Guarantee problems are **PSPACE**-complete for SyDSs on directed graphs where each local function is the majority function.

Acknowledgements

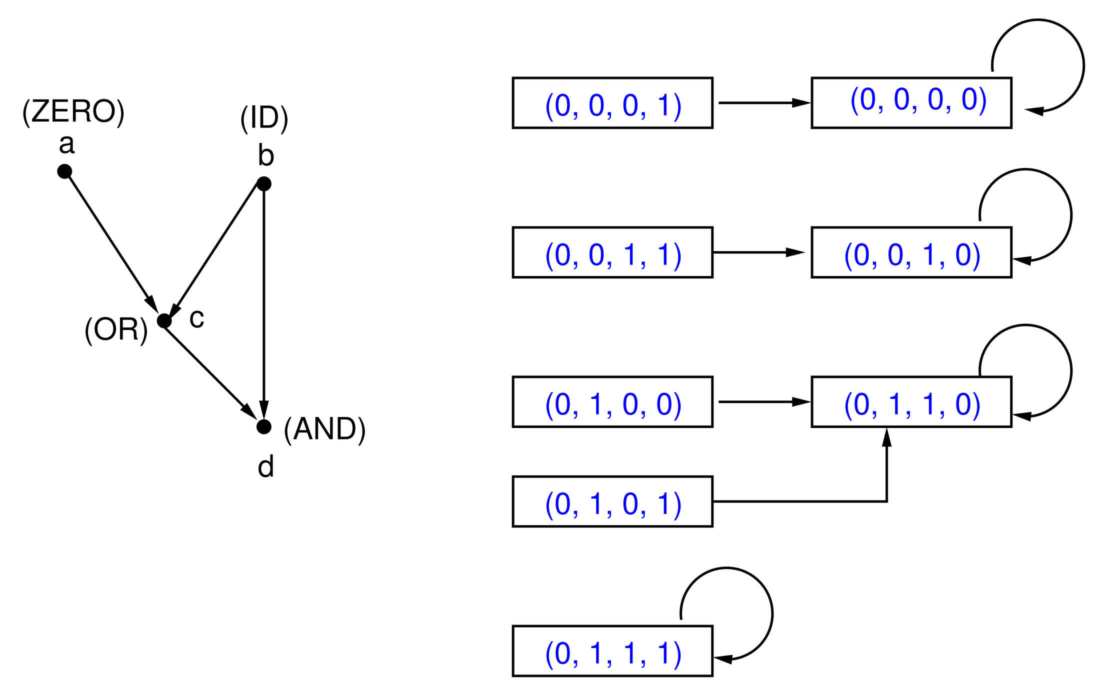
We thank the referees of AAAI-2021 for providing helpful feedback. This work was partially supported by NSF Grants IIS-1633028 (BIG DATA), CMMI-1745207 (EAGER), OAC-1916805 (CINES), CCF-1918656 (Expeditions) and IIS-1908530.

Definitions

- **Configuration** at time t: Vector specifying the state of each node at time t.
- **Successor** of a configuration $\mathcal{C}$: The configuration that **immediately follows** $\mathcal{C}$ in time evolution.
- **Fixed Point**: A configuration $\mathcal{C}$ whose successor is $\mathcal{C}$ itself.
- The **phase space** of a discrete dynamical system $\mathcal{S}$ is a **directed graph** $\mathcal{P}$.
  - Each node of $\mathcal{P}$ represents a configuration of $\mathcal{S}$.
  - Each directed edge (x, y) indicates that y is the successor of x.

Note: The size of $\mathcal{P}$ is **exponential** in the size of $\mathcal{S}$.

Example – Phase Space of DAG-SyDS $\mathcal{S}$:



Note: Only a part of the phase space is shown.

Our Contributions – I

- **Reachability** is **PSPACE**-complete for **symmetric** DAG-SyDSs.

  Proof Idea: Reduction from Quantified 3SAT.
- **Convergence Guarantee** is **Co-NP**-complete for DAG-SyDSs with at most **three** levels.

  Proof Idea: Reduction from 3SAT.
- New structural properties of the phase spaces of DAG-SyDSs (e.g., bounds on lengths of cycles and transients in phase space).

Contact Author

S. S. Ravi (`ssravi0@gmail.com`)

Problems Considered

Motivation – Diffusion in Networks:

- **Contagion** processes model many social phenomena (e.g., propagation of information, influence, diseases, trends, etc.).
- Example: Diffusion of opinions in networks (e.g., [Auletta et al. 2018, Chistikov et al. 2020, Botan et al. 2019, Bredereck & Elkind 2017]).
- Usual modeling assumptions:
  - Agents in the system have states that vary with time.
  - The next state of an agent depends on her current state and those of her neighbors (i.e., agent interactions are **local**).

Analysis Problems:

Reachability:

Given: SyDS $\mathcal{S}$ and configurations $\mathcal{C}_1$ and $\mathcal{C}_2$.
Question: Does $\mathcal{S}$ starting from $\mathcal{C}_1$ reach $\mathcal{C}_2$?

Convergence:

Given: SyDS $\mathcal{S}$ and configuration $\mathcal{C}_1$.
Question: Does $\mathcal{S}$ starting from $\mathcal{C}_1$ reach a fixed point?

Convergence Guarantee:

Given: SyDS $\mathcal{S}$.
Question: Does $\mathcal{S}$ reach a fixed point from **every** starting configuration?

Note: The above questions concern the phase space $\mathcal{P}(\mathcal{S})$ of $\mathcal{S}$.

Our Contributions – II

- **Reachability** is efficiently solvable for **monotone** DAG-SyDSs.

  Proof Idea: Show that each cycle is a fixed point; use the result that the transient length is $\leq$ the number of levels in the DAG.

Future Work:

- DAG-SyDSs with other local functions (e.g., weighted threshold functions).
- Stochastic DAG-SyDSs.