# eM's for Model-Based Reinforcement Learning

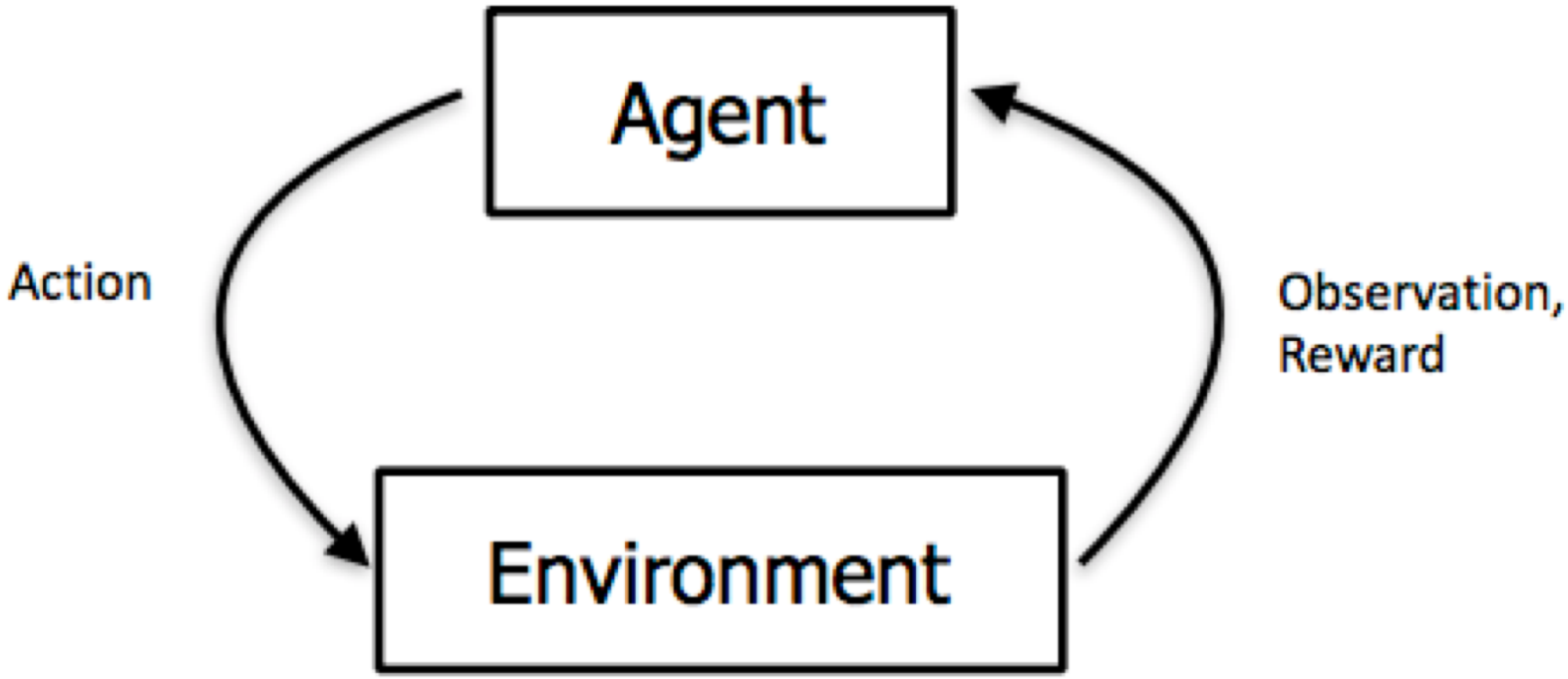
POCI Project, 2021

Sam Holton

# Motivation

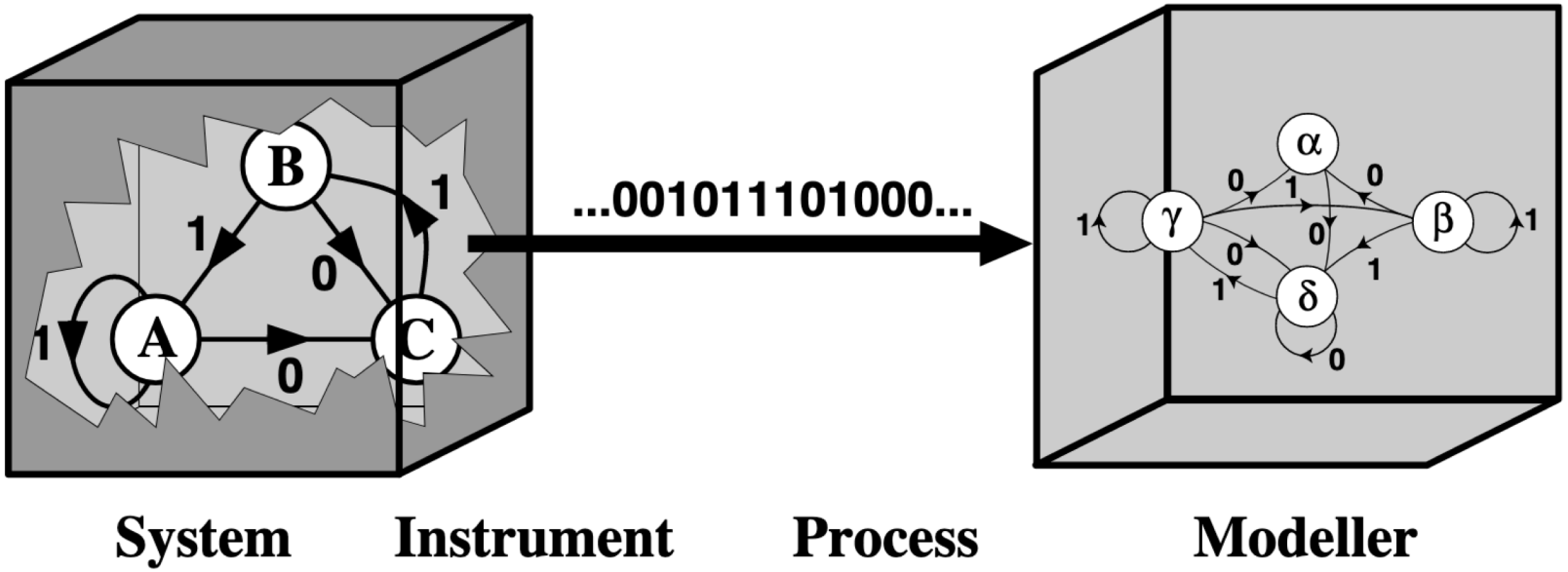
Chaos, Quantum Mechanics -> “observer” has active role

Control the system -> RL

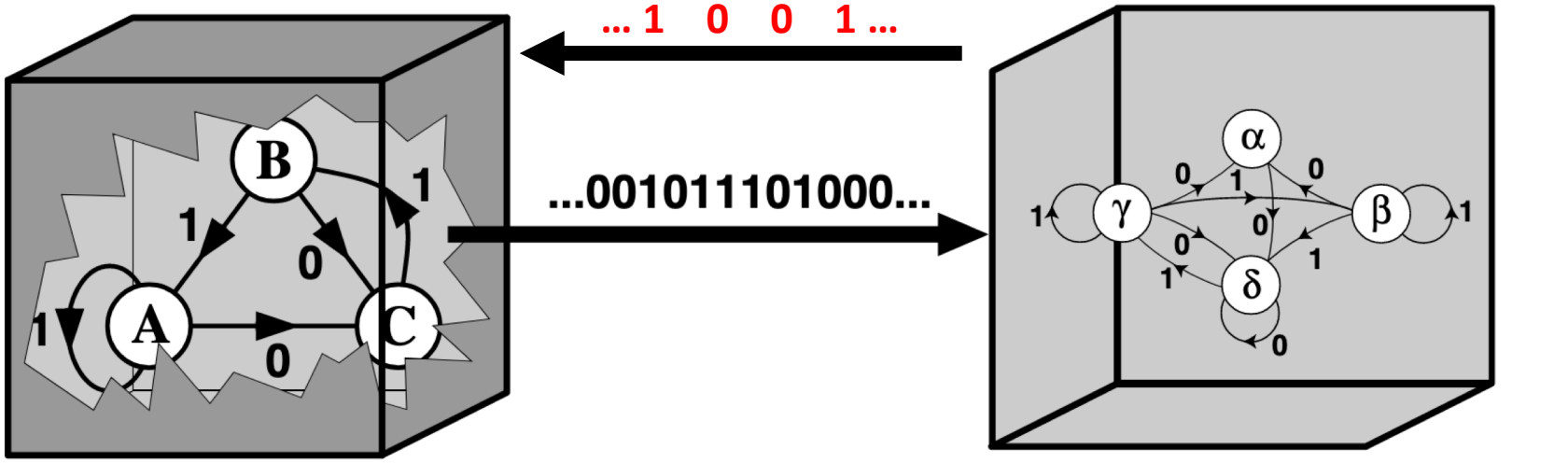



https://itnext.io/reinforcement-learning-with-q-tables-5f11168862c8

# Modelling the RL problem

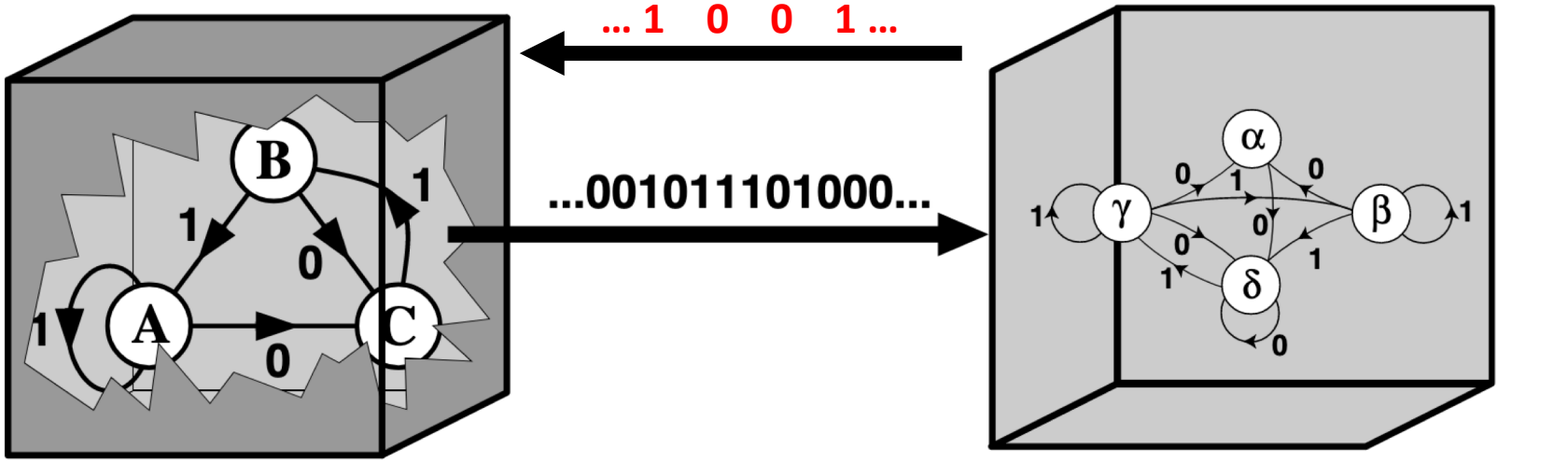


# Modelling the RL problem




System Instrument Process ~~Modeller~~ Agent

# Modelling the RL problem

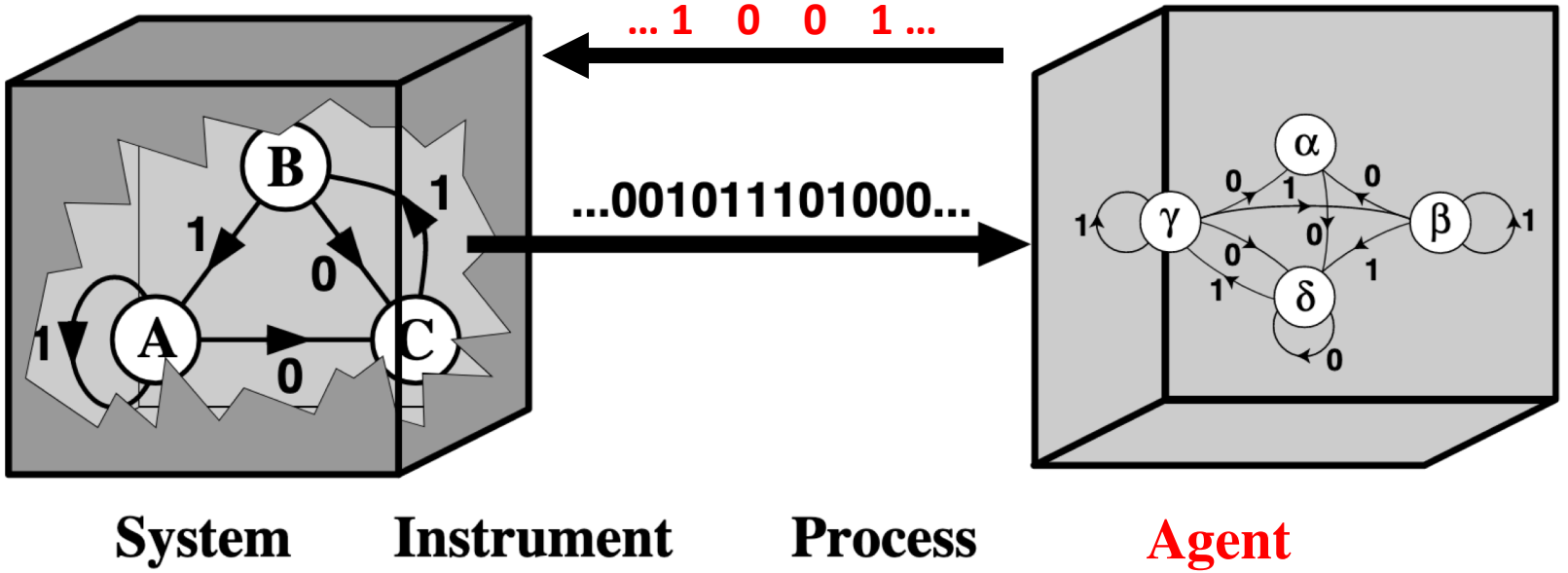




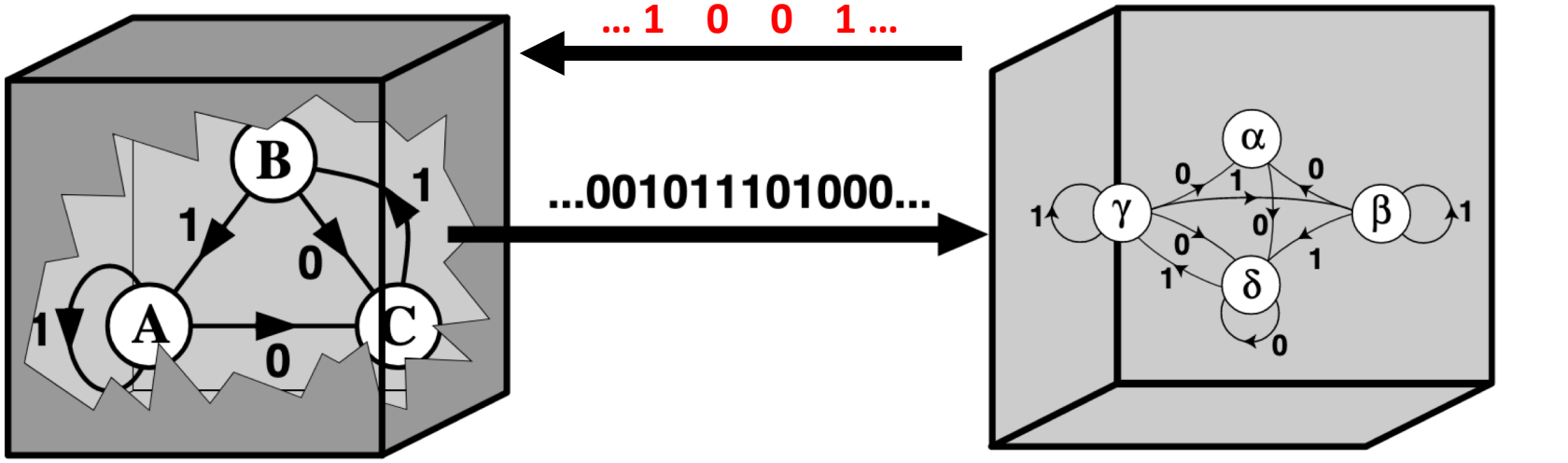

System History: … 001011101000 …

System + Agent History: … 0011011010100001 …

# Modelling the RL problem



# Modelling the RL problem






System History: … 001011101000 …

System + Agent History: … 0011011010100001 …

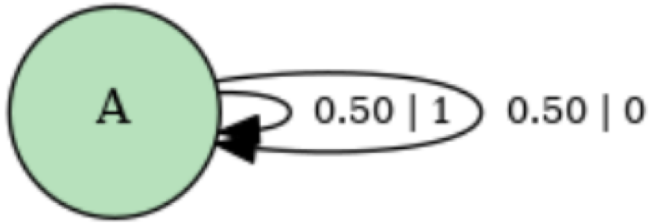
1. Initial random policy
2. Observe output -> Infer eM
3. Choose new policy

# Simple Example: Rain or Shine

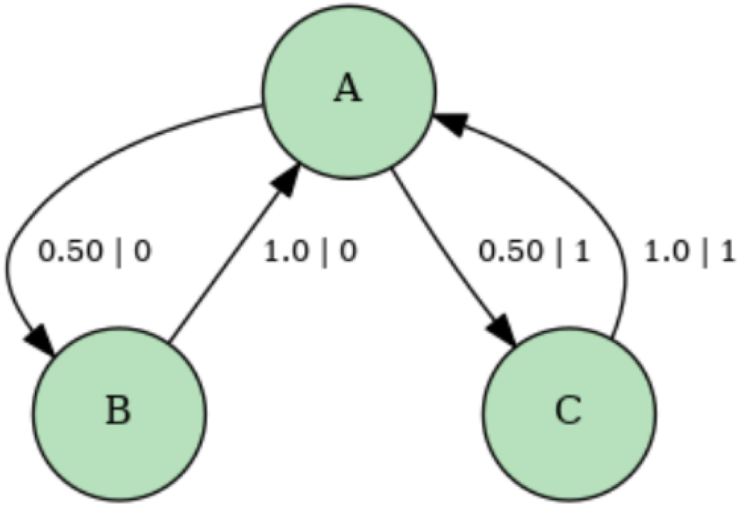
- Random weather
- Agent chooses 2nd bit e.g. 01 11 11 00
- Want sunglasses when sunny (11) and raincoat when rainy (00)

| Word | Result |
|---|---|
| 00 | Raincoat when raining -> ☺ |
| 01 | Sunglasses when raining -> ☹ |
| 10 | Raincoat when sunny -> ☹ |
| 11 | Sunglasses when sunny -> ☺ |

Random Policy



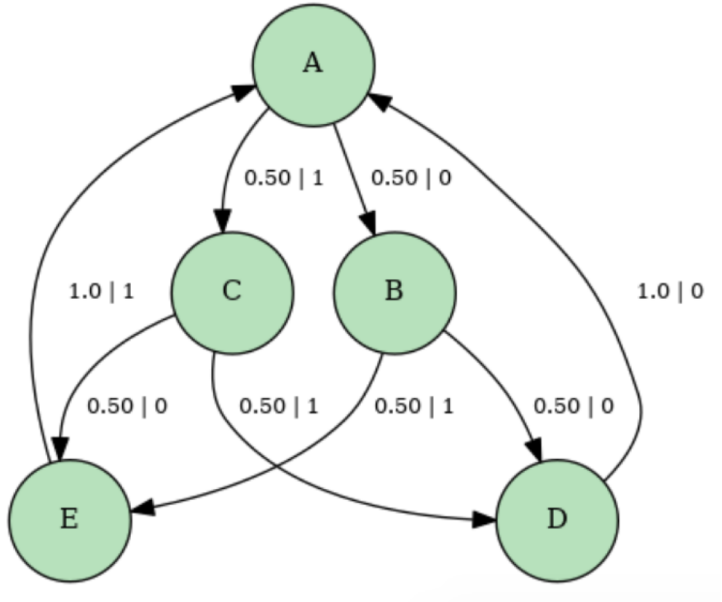

Optimal Policy



# Controlling RRXOR

Agent chooses
2nd bit in RRXOR

e.g. 011

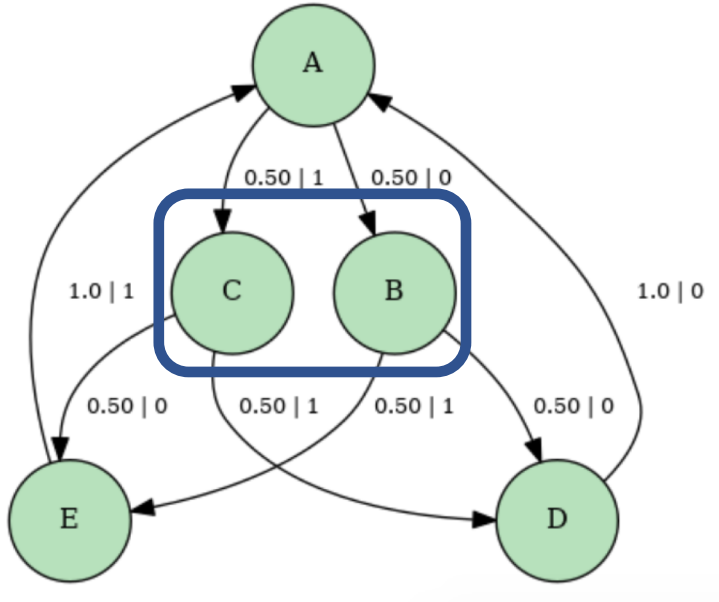



Random
Policy

# Controlling RRXOR

Agent chooses
2nd bit in RRXOR

e.g. 011

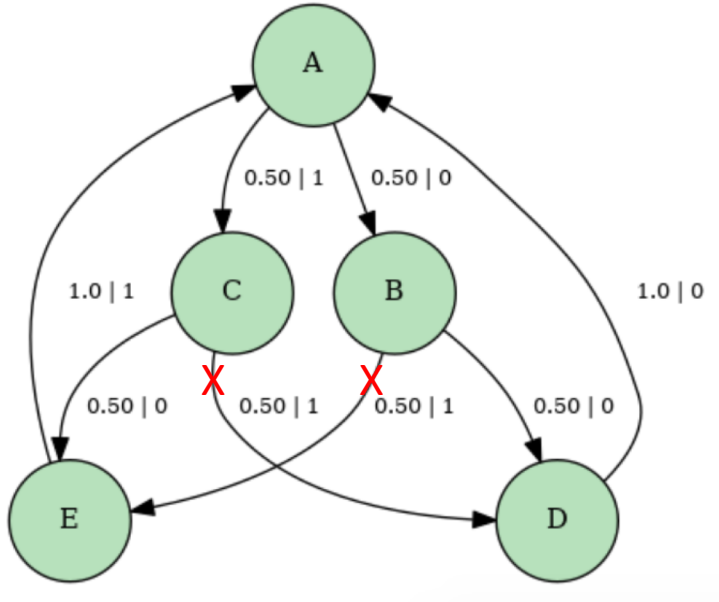



Random
Policy

# Controlling RRXOR

Agent chooses
2nd bit in RRXOR

e.g. 011

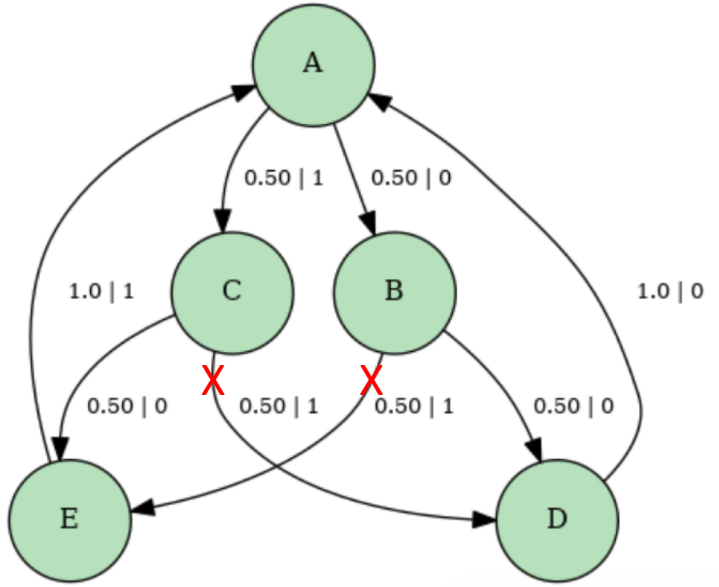



Random
Policy

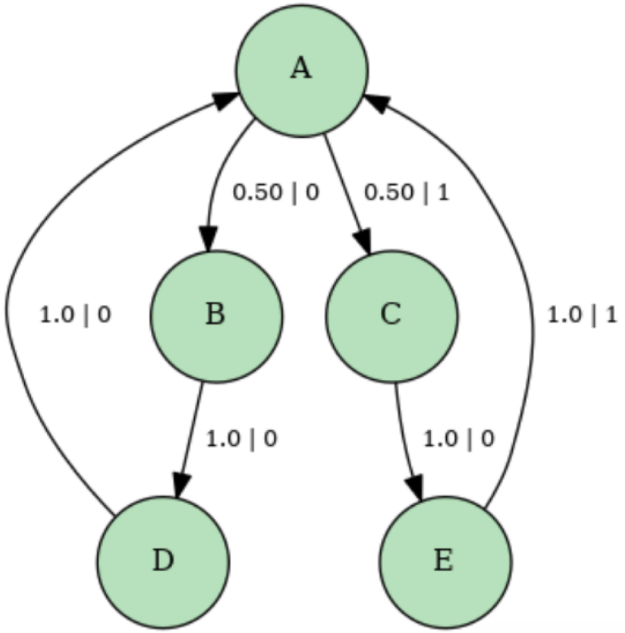
# Controlling RRXOR

Agent chooses
2nd bit in RRXOR

e.g. 011




Random Policy

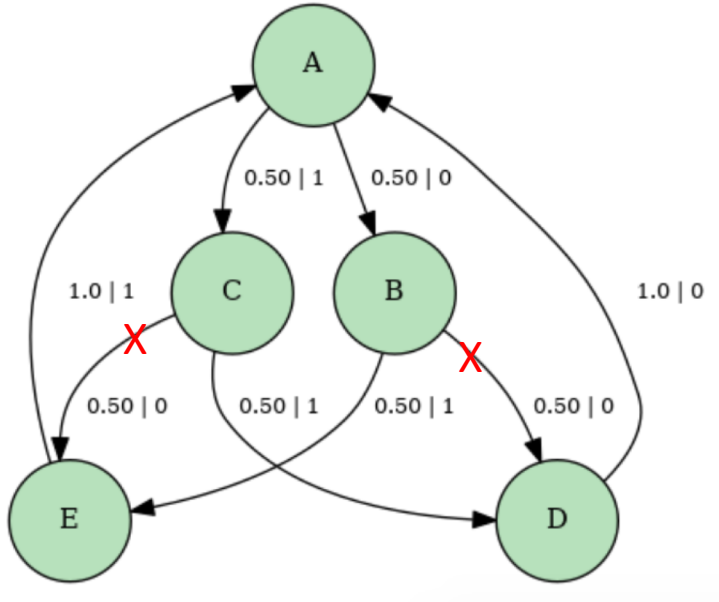



All-0's Policy

# Controlling RRXOR

Agent chooses
2nd bit in RRXOR

e.g. 011

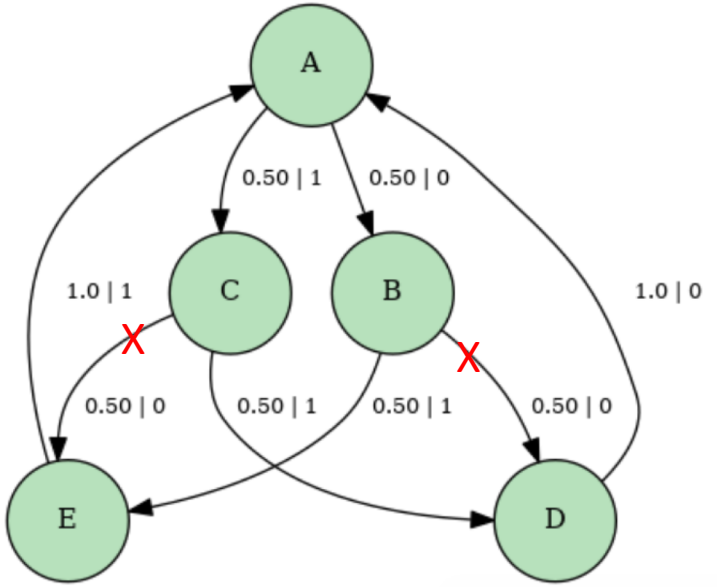



Random
Policy

# Controlling RRXOR

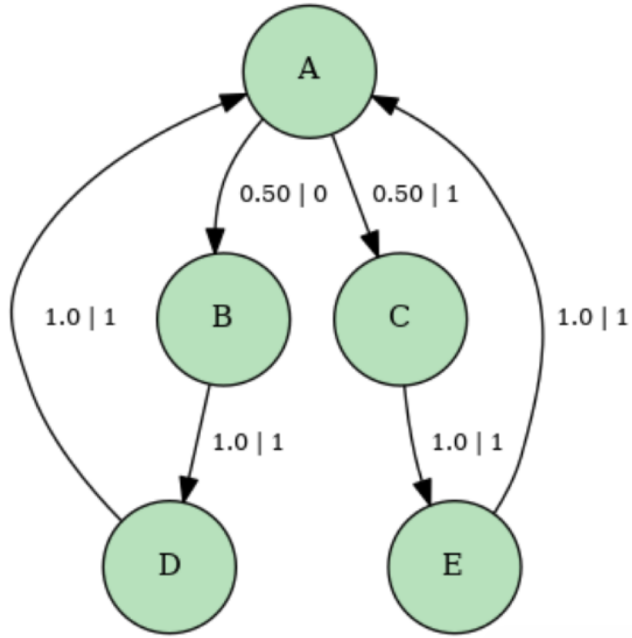
Agent chooses
2nd bit in RRXOR

e.g. 011




Random
Policy

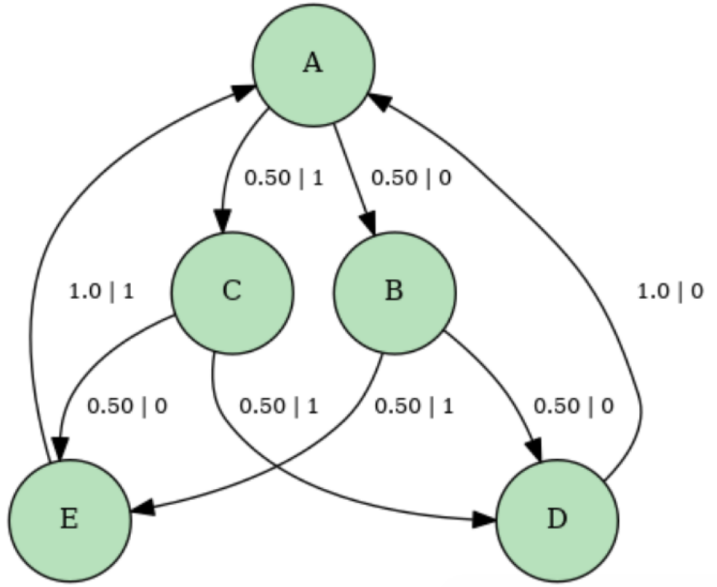
All-1's
Policy



# Controlling RRXOR

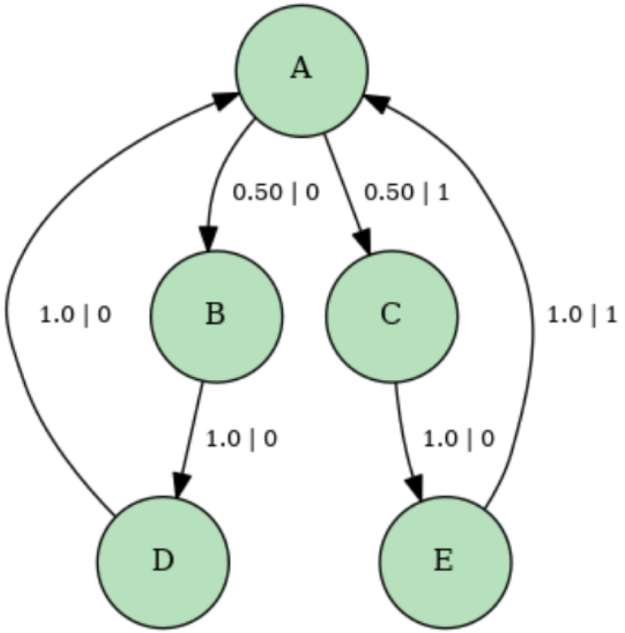
Agent chooses
2nd bit in RRXOR

e.g. 011

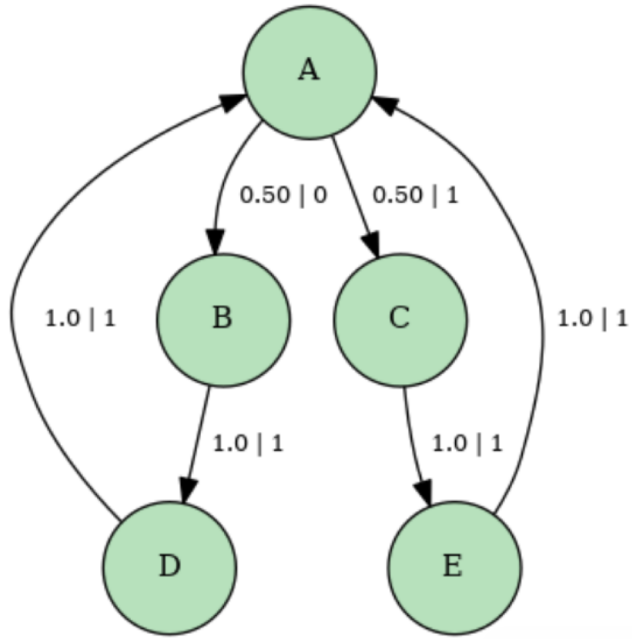



Random Policy

All-0's Policy



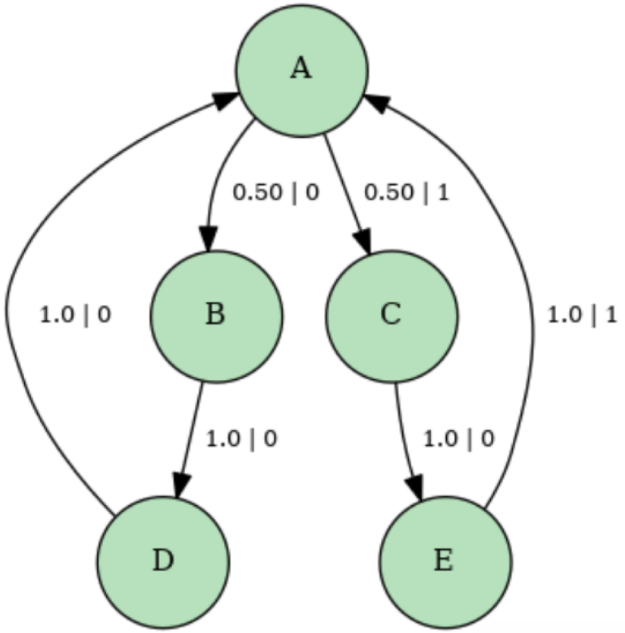

All-1's Policy



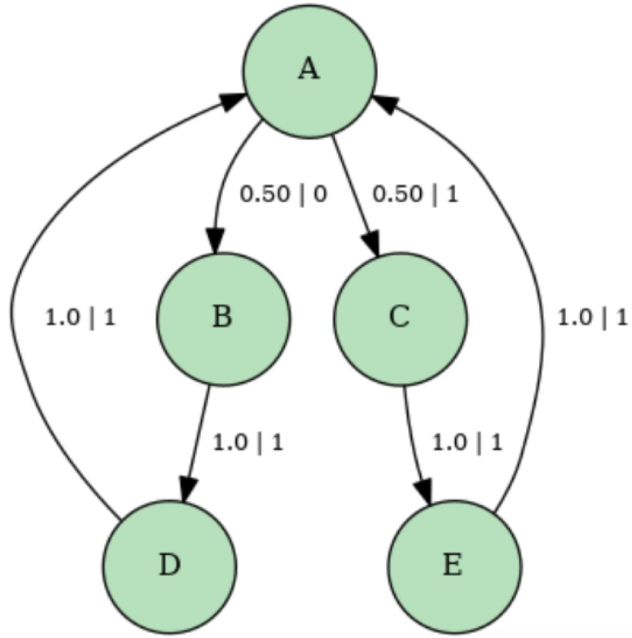
# Picking a policy

- Choose output at controlled causal states
- Policy -> eM -> word distribution -> utility

All-0's Policy

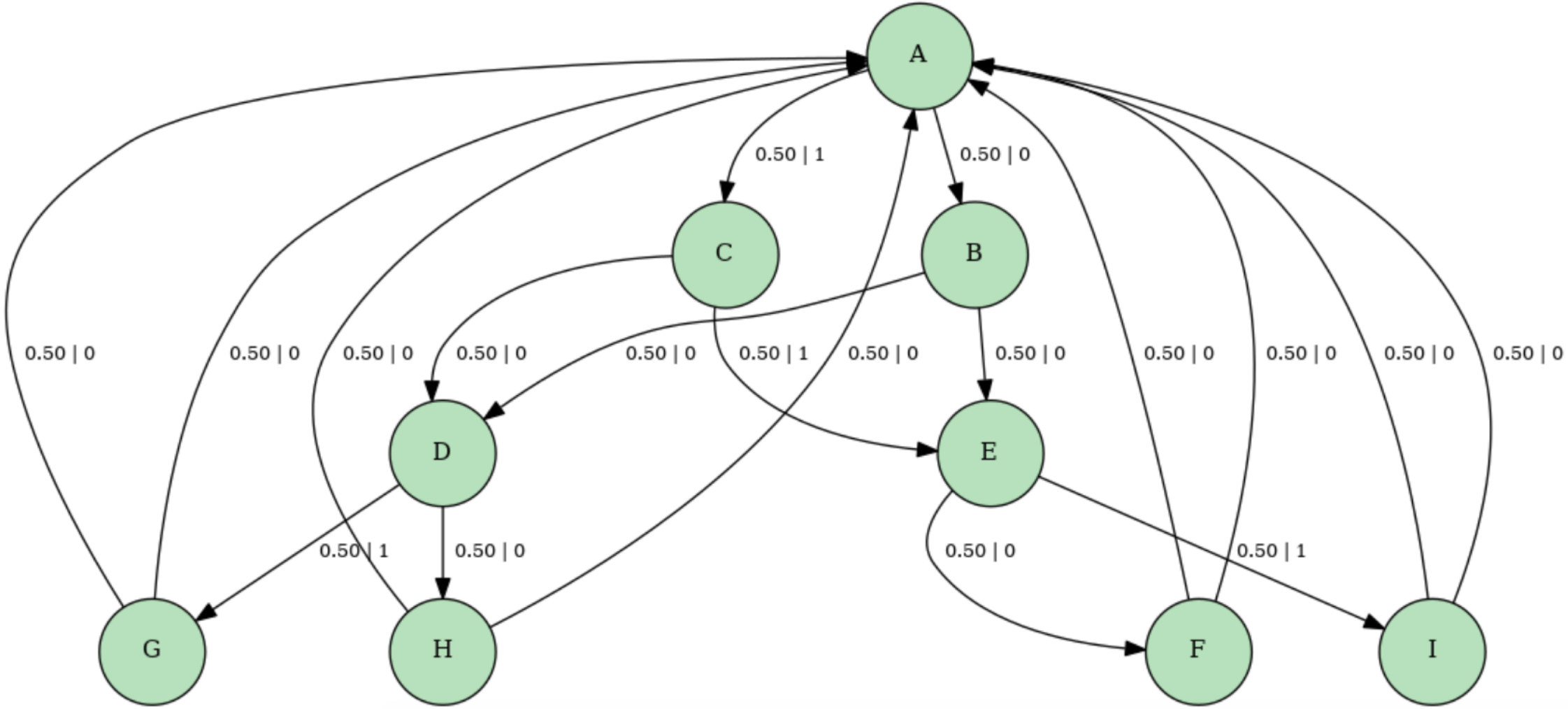



All-1's Policy



# Larger System

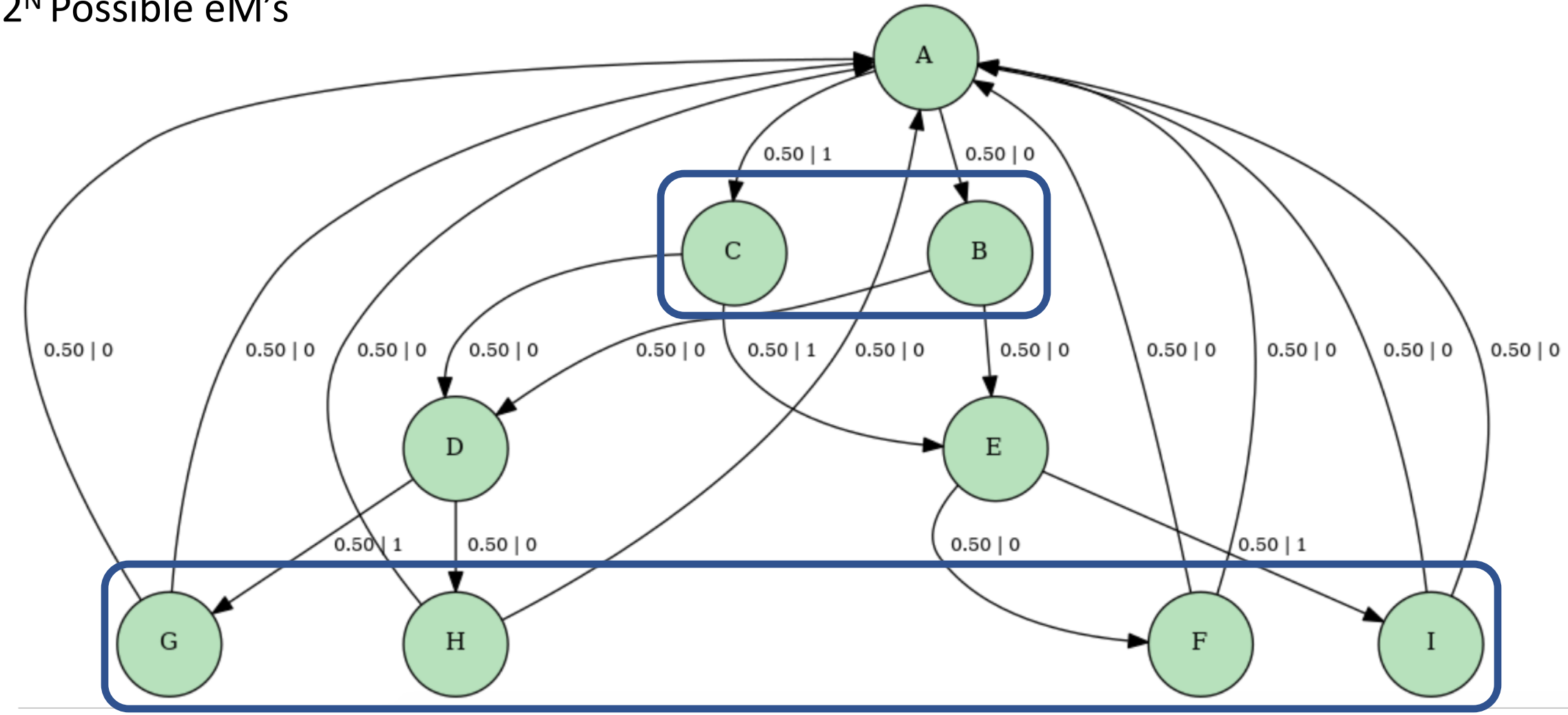
Random-Controlled-Random-Controlled



# Larger System

Random-Controlled-Random-Controlled

**Problem:** $2^N$ Possible eM's

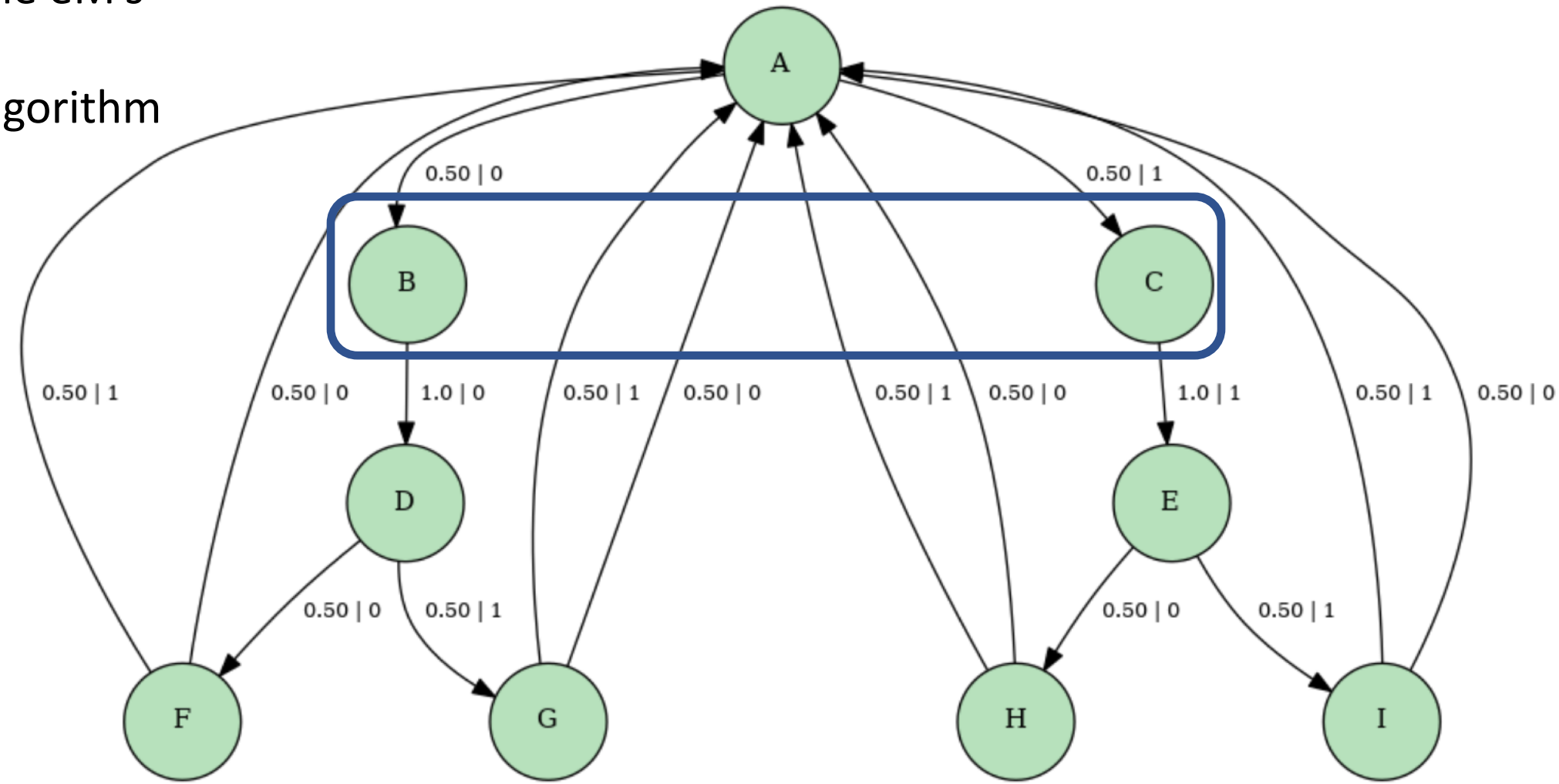


# Larger System

Random-Controlled-Random-Controlled

**Problem:** $2^N$ Possible eM's

**Solution:** greedy algorithm

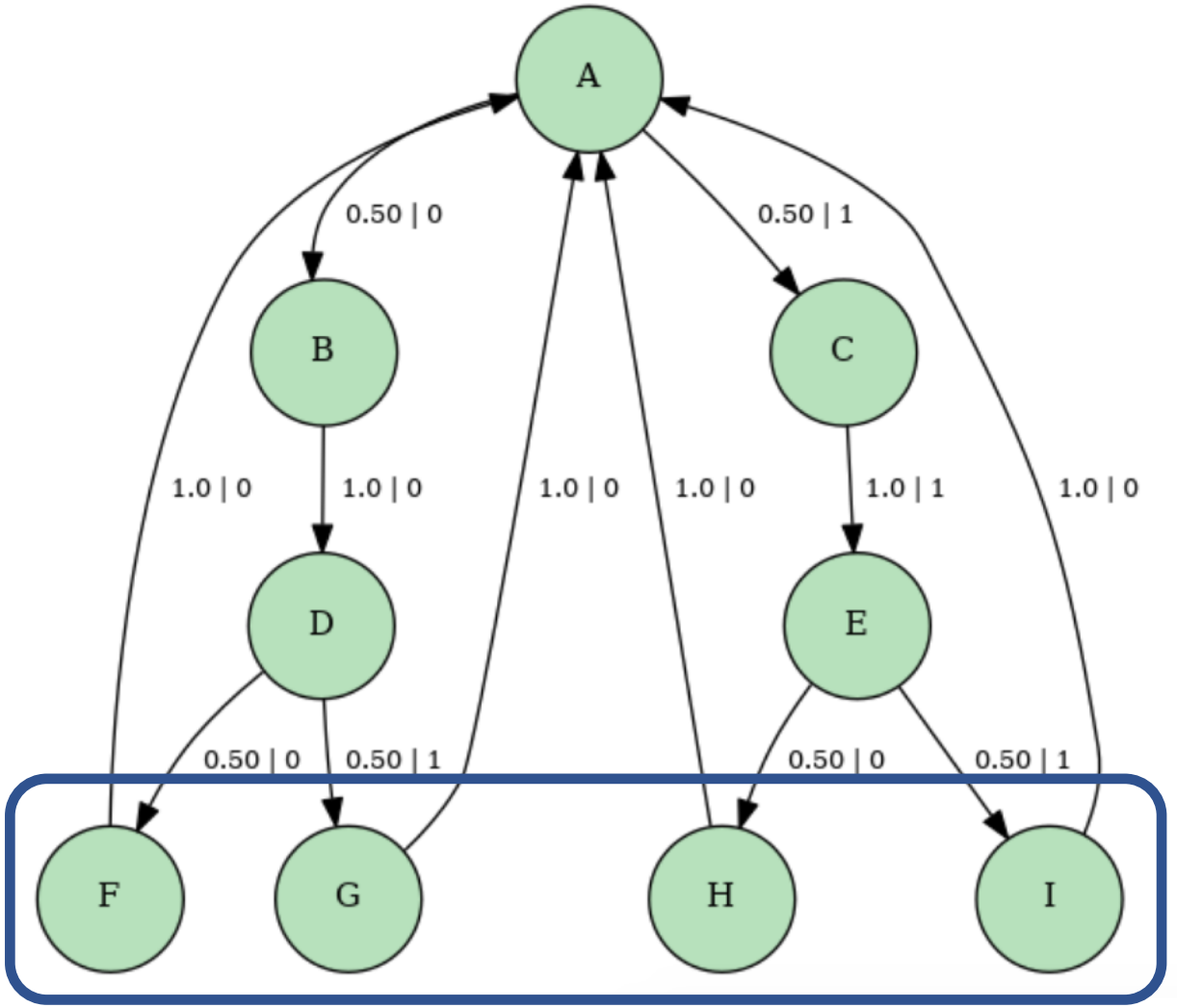


# Larger System

Random-Controlled-Random-Controlled

**Problem:** $2^N$ Possible eM's

**Solution:** greedy algorithm



# Future Directions

- Bigger eM’s, full Bayesian approach, explore-exploit etc.
- Relationship between controller/system info measures -> Good Regulator Theorem?
- Interacting agents