A Summary of Some Topics: Learning Automata

B. John Oommen

Chancellor's Professor

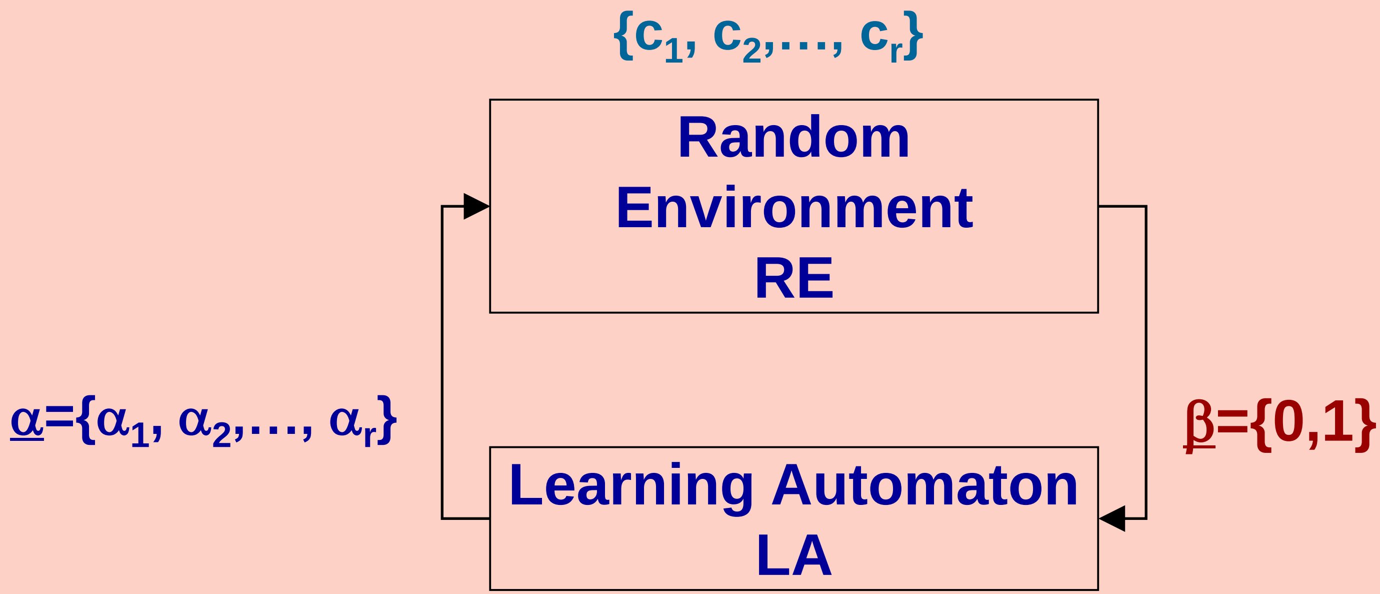
Carleton University, Canada

Life Fellow : IEEE ; Fellow : IAPR

Learning Automata

- ***Learning Problem:***
  - Acquisition and utilization of relevant knowledge
  - Improve the performance of a system.
- ***Learning Automaton:*** Model of computer learning used to solve the learning problem
  - **Models** - Biological learning systems
  - **Goal** - Determine the optimal action from a set
  - **Optimal Action** - Has *Minimum Penalty Probability*
  - **Learns** - Process responses from random *Environment*

Learning Automata - Learning Loop



- $\underline{\alpha}=\{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ - *r* actions
- $\{c_1, c_2, \ldots, c_r\}$ - action *penalty probabilities*
- $\underline{\beta}=\{0,1\}$ - response from the *Environment*:

Reward and *Penalty*

Learning Automata - Learning Loop

LA chooses one from set of actions $\{\alpha_1,..,\alpha_r\}$ offered by *Environment* **RE**

Chosen action $\alpha(t)$ is Input to the **RE**

RE *rewards* or *penalizes* LA
Based on penalty probabilities

RE's response is Input to **LA**
Then chooses next action

Norms of Behavior

- ***Expedient***: Automaton better than *pure-chance* machine:

$$\lim_{t\to\infty} E[M(t)] < M_0$$

- ***Absolutely Expedient***: $E[M(t+1) \mid \mathbf{P}(t)] < M(t)$

- ***Optimal***: $\lim_{t\to\infty} p_b(t) \to 1$ with probability 1.

  Note: There are no optimal learning automata.

- ***ε-Optimal***: A LA is said to be *ε-optimal* if

For any $\varepsilon>0$ and $\delta>0$, there exists $t_0>\infty$ and $\lambda_0>0$ such that

$$\Pr\left[\,|p_b(t)-1| < \varepsilon\,\right] > 1-\delta$$

Norms of Behavior (II)

Absolute Expediency → **ε-Optimality**

In all stationary random environments

Categories of Learning Automata

- **Deterministic** – Transition/Output Matrices deterministic

- **Stochastic** - Transition or output matrices are stochastic

  - **Fixed Structure Stochastic Automata (FSSA):** Transition and output matrices are *time invariant*

  - **Variable Structure Stochastic Automata (VSSA):** Transition or output matrices *change with time*

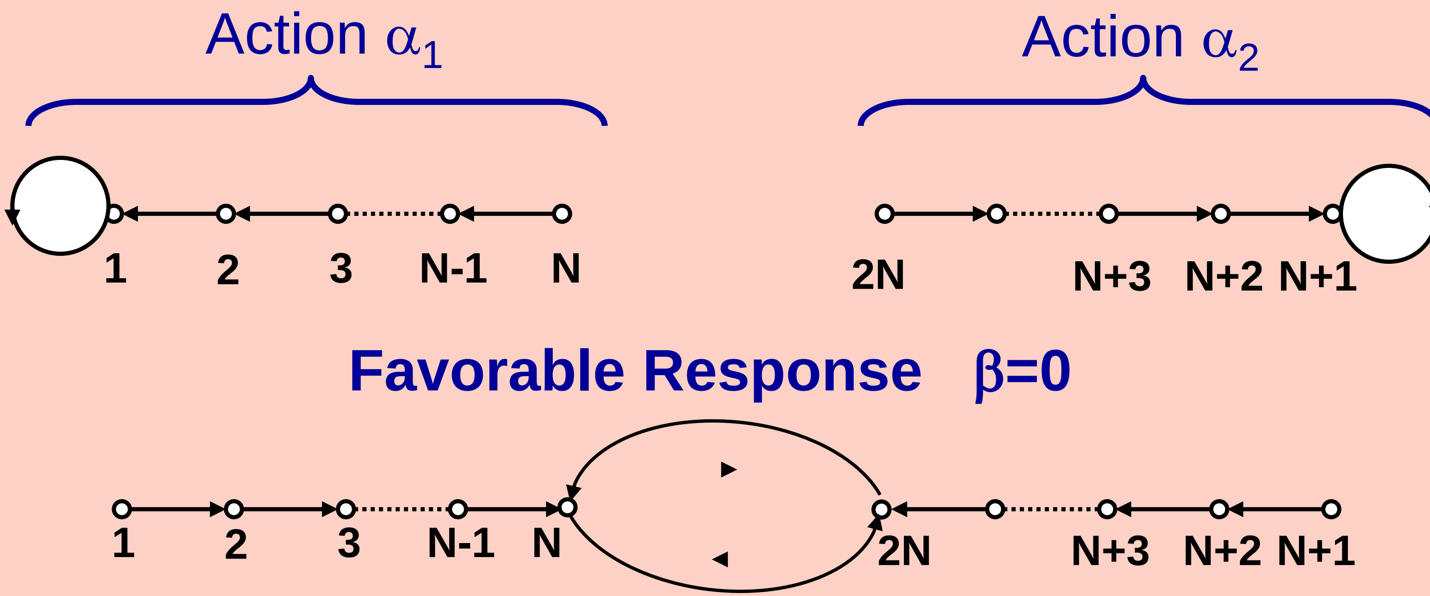
Deterministic Automata

Tsetlin Automaton

Krinsky Automaton

Are deterministic automata with 2N states and 2 actions.

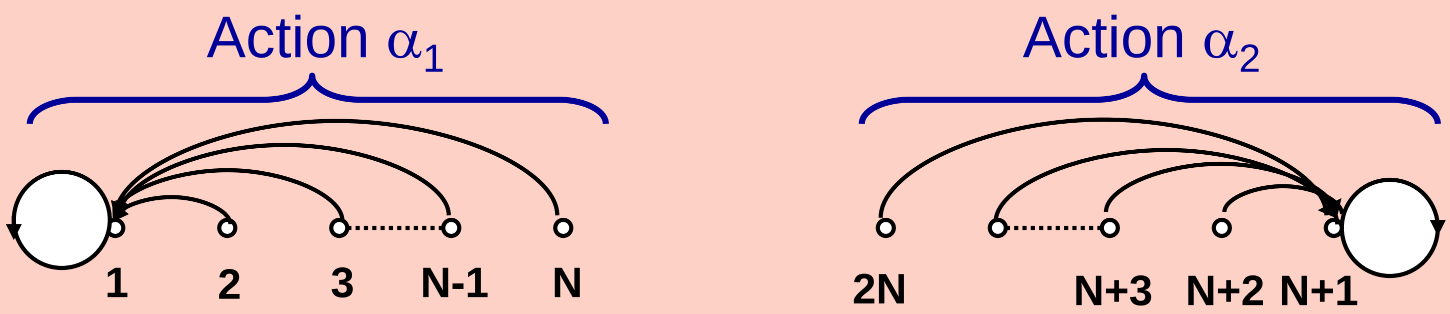
Tsetlin Automaton



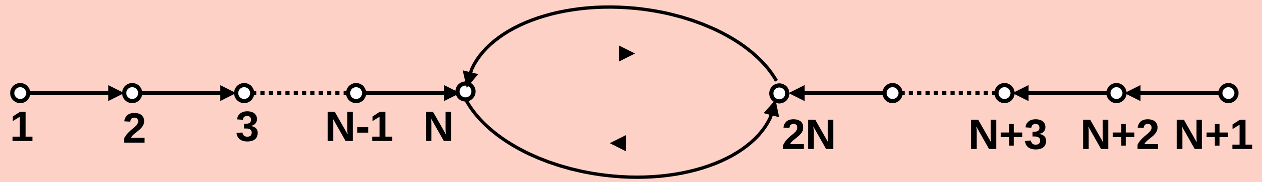
Unfavorable Response β=1

- ε-Optimal when $\min\{c_1,c_2\} \leq 0.5$
- Ergodic: (Type of Markov Chain; Don't worry about it)

Krinsky Automaton



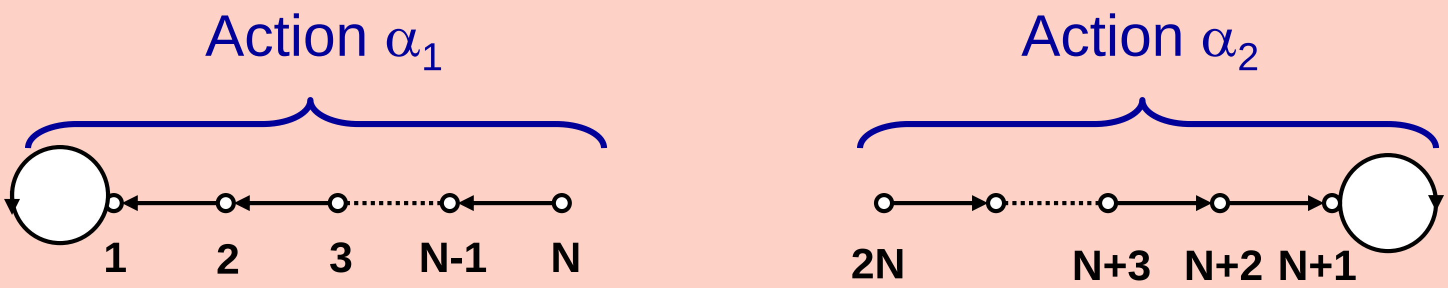
Favorable Response β=0



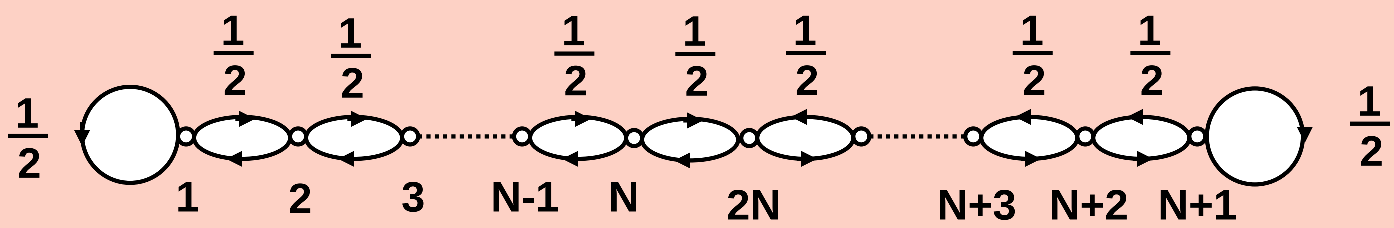
Unfavorable Response β=1

- ε-Optimal in all stationary random environments
- Ergodic (Type of Markov Chain; Don't worry about it)

Krylov Automaton



Favorable Response $\beta=0$



Unfavorable Response $\beta=1$

- FSSA automaton with 2N states and 2 actions
- ε-Optimal in all stationary random environments

Variable Structure Stochastic Automata

- State transition probabilities or action selecting probabilities are updated with time

- Defined in terms of Action Probability Updating Schemes

- Operates on the action probability vector $\mathbf{P}(t)=\begin{pmatrix} p_1(t) \\ p_2(t) \\ \ldots \\ p_r(t) \end{pmatrix}$

- Updates the action probability vector $\mathbf{P}(t+1)$ based on
  - $\mathbf{P}(t)$ - Previous value of the action probability vector
  - $\beta(t)$ - Response of the Environment

Categories of VSSA

Classification
based on the
learning paradigm

- ***Reward-Penalty schemes***
- ***Reward-Inaction schemes***
- ***Inaction-Penalty schemes***

Classification
based on the properties of
probability space [0,1]

- ***Continuous schemes***
- ***Discrete schemes***

Categories of VSSA (II)

- **Ergodic Schemes**
  - Learning Automaton does not lock in any action
  - Limiting distribution: independent of initial distribution
  - Used for Non-Stationary Random Environments
  - Example: Linear Reward-Penalty (L_{RP}) scheme

- **Absorbing Schemes**
  - Learning Automaton gets locked into its final action
  - Limiting distribution: dependent of initial distribution
  - Used for Stationary Random Environments
  - Example: Linear Reward-Inaction (L_{RI}) scheme

Exp. of Continuous Scheme: L_{RI} (II)

- **Action Probability Updating Scheme:**

$p_1(t+1)=p_1(t)+\lambda(1-p_1(t))$ - if α_1 is rewarded or α_2 is penalized

$p_1(t+1)= (1-\lambda)p_1(t)$ - if α_1 is penalized or α_2 is rewarded

$p_2(t+1)= 1-p_1(t)$

Exp. of Continuous Scheme: L_{RI}

- **Example:**

$$P(t) = \begin{pmatrix} 0.4 \\ 0.3 \\ 0.1 \\ 0.2 \end{pmatrix} \xrightarrow[\lambda = 0.2]{\alpha_2 \text{ chosen and rewarded}} P(t+1) = \begin{pmatrix} 0.32 \downarrow \\ 0.44 \uparrow \\ 0.08 \downarrow \\ 0.16 \downarrow \end{pmatrix}$$

- p_2 increased
- p_1, p_3, p_4 decreased

If α_1 is the best action:

$$\Longrightarrow \begin{pmatrix} p_1(\infty) \\ p_2(\infty) \\ p_3(\infty) \\ p_4(\infty) \end{pmatrix} \longrightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Thanks!!!

Thank You Very Much!!