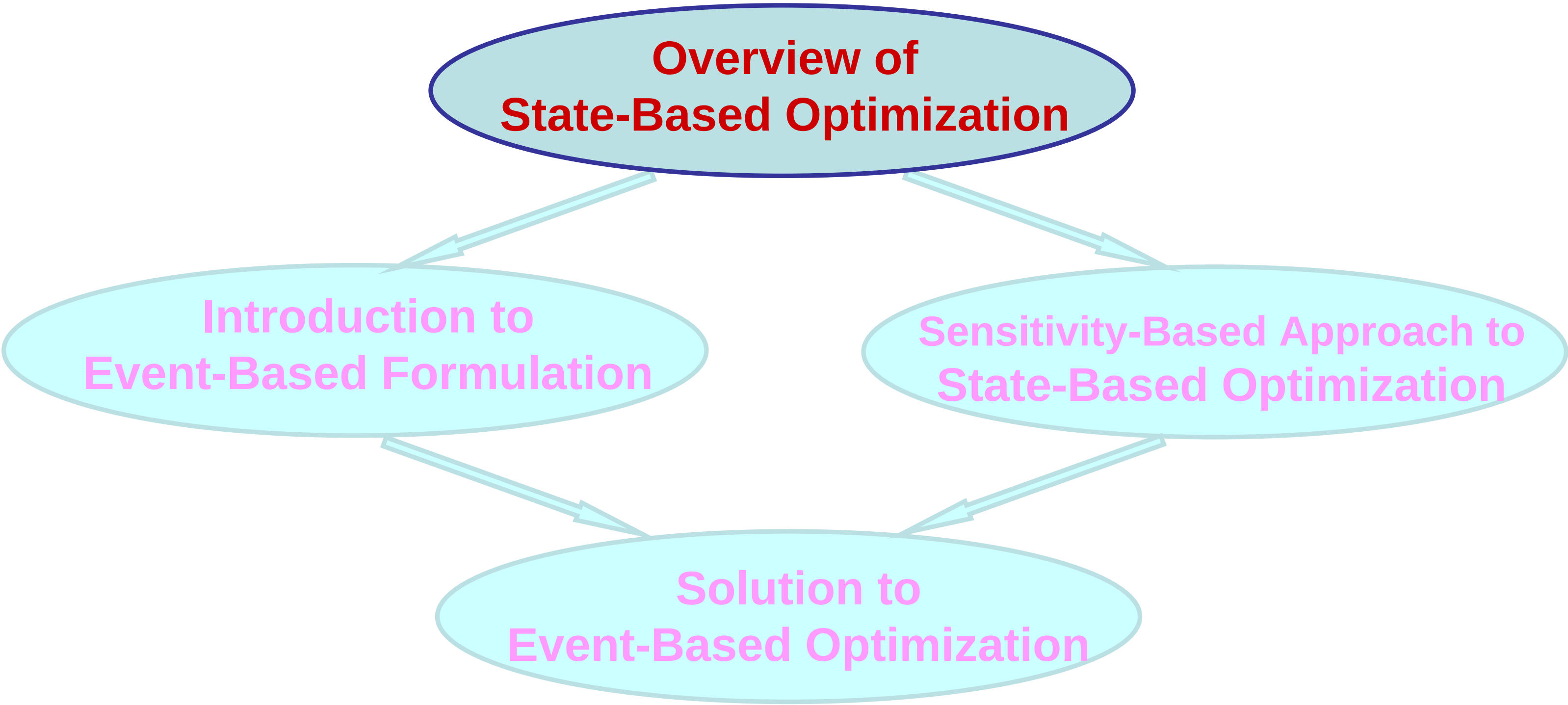
# Event-Based Optimization

## - A Strategy That Depends on the Future!

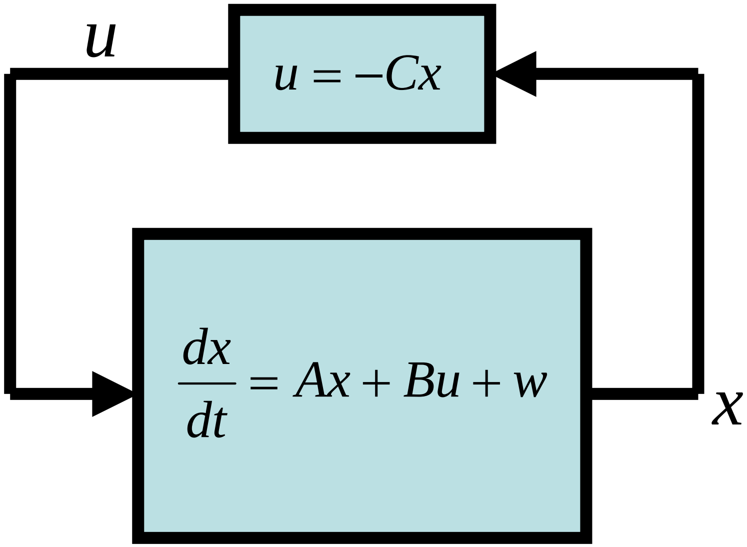
Plenary Presentation
at
2007 IEEE Multi-Conference on Systems and Control
October 2, 2007 Singapore

Xi-Ren Cao

Hong Kong University of Science and Technology


Overview of
State-Based Optimization
Introduction to
Event-Based Formulation
Sensitivity-Based Approach to
State-Based Optimization
Solution to
Event-Based Optimization

# A Typical Formulation of a Control Problem (Continuous-Time Model)




*x*: *State*

*u*: *Control variable*

*w*: *Random noise*

*Performance measure*

$$\eta = \frac{1}{T}\int_0^T E\{f[x(t), u(t)]\}dt$$

LQG problem

$$\eta = \frac{1}{T}\int_0^T E\{x^\tau Ax + u^\tau Bu\}dt$$

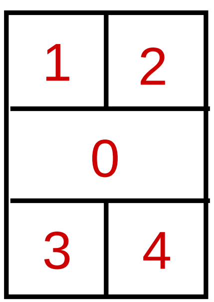
*Control u depends on state x*

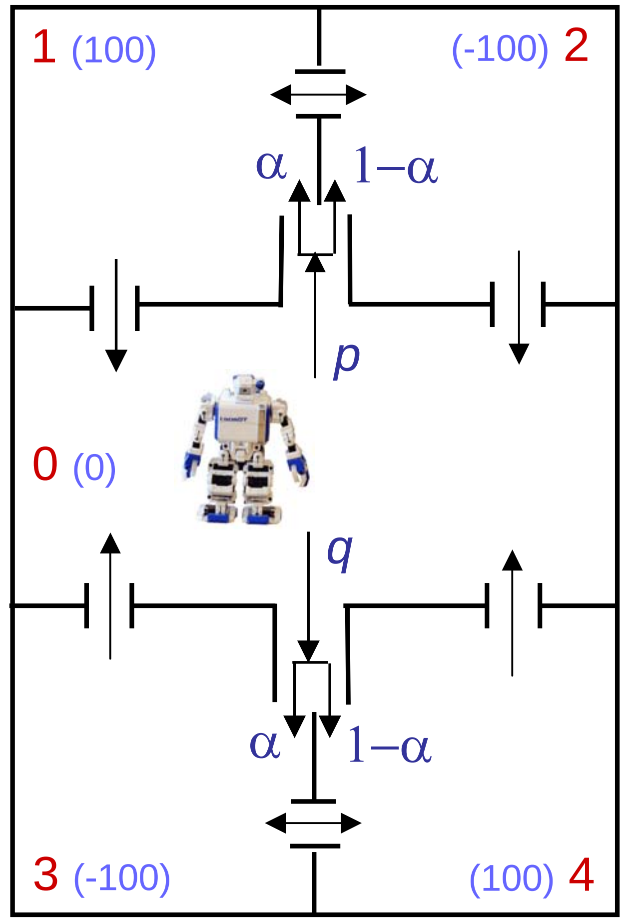
A policy u(x): x →u

State-based vs. Event-based formulation

# Discrete-time Model (I)
## - an example

A random walk of a robot




*Probabilities*

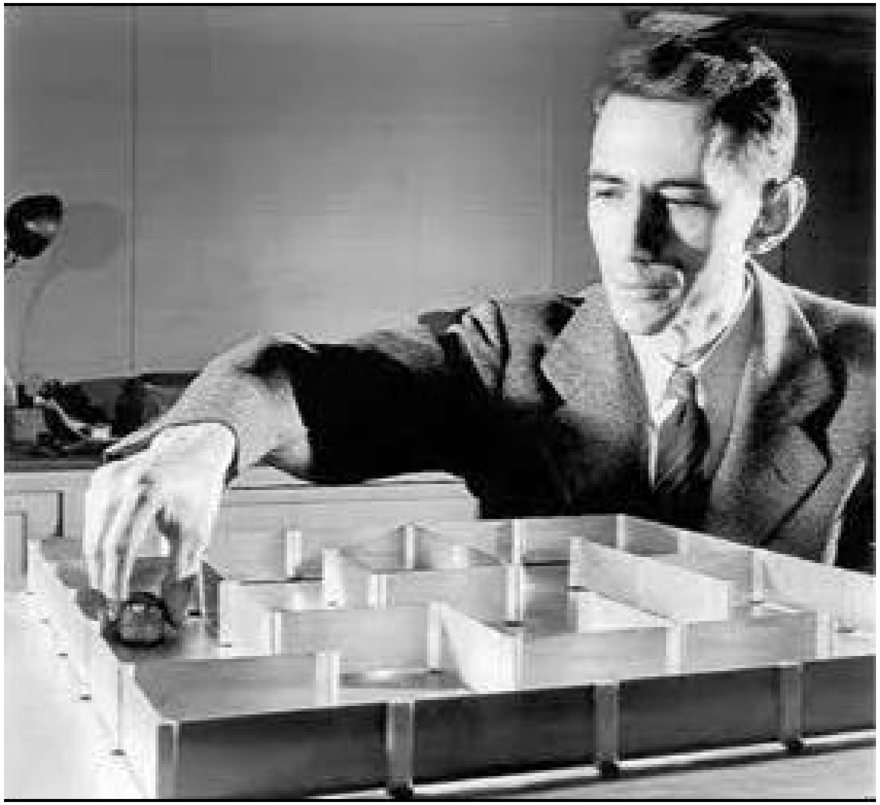
$$p + q = 1$$

*Reward function*

f(0) = 0
f(1) =f(4)=100
f(2) =f(3)= -100

*Performance measure*

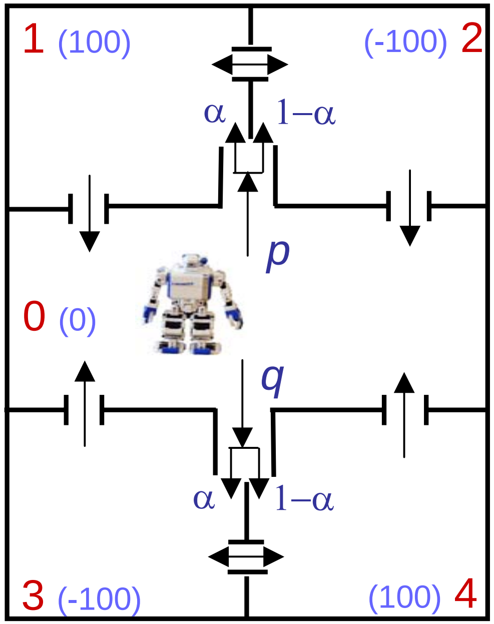
$$\eta = \lim_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} f(X_t)$$



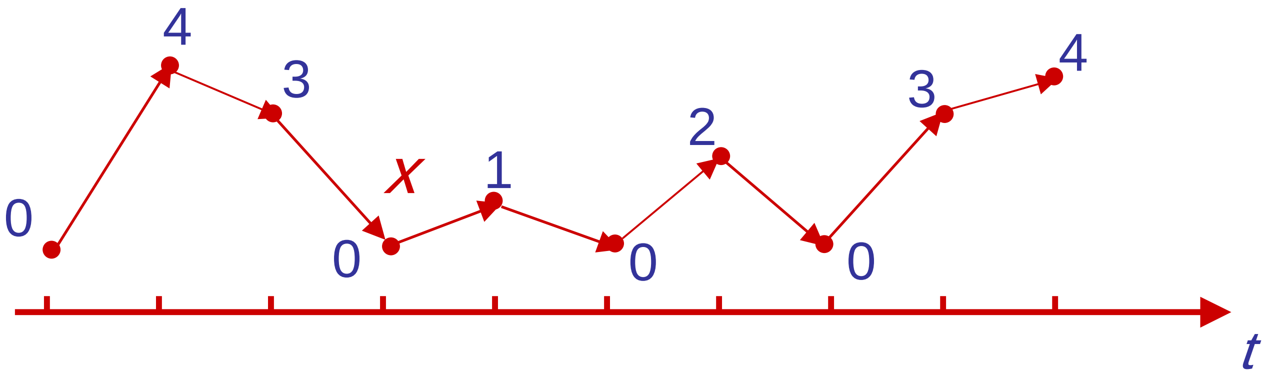
Shannon Mouse (Theseus)

# Discrete-time Model (II)
## - the dynamics

A random walk
of a robot

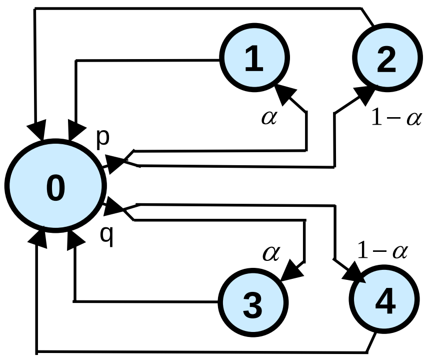



*A Sample Path (system dynamics):*



# Discrete-time Model (III)
## - the Markov model

Random Walker




**System dynamics:**

-X = {$X_n$, n=1,2,…}, $X_n$ in S = {1,2,…,M}

- Transition Prob. Matrix $P=[p(i,j)]_{i,j=1,..,M}$

**System performance:**

– Reward function: $f=(f(1),\ldots,f(M))^T$

– Performance measure:

$$\eta = \lim_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} f(X_t) = \pi f = \sum_{i\in S} \pi(i) f(i)$$

**Steady-state probability:**

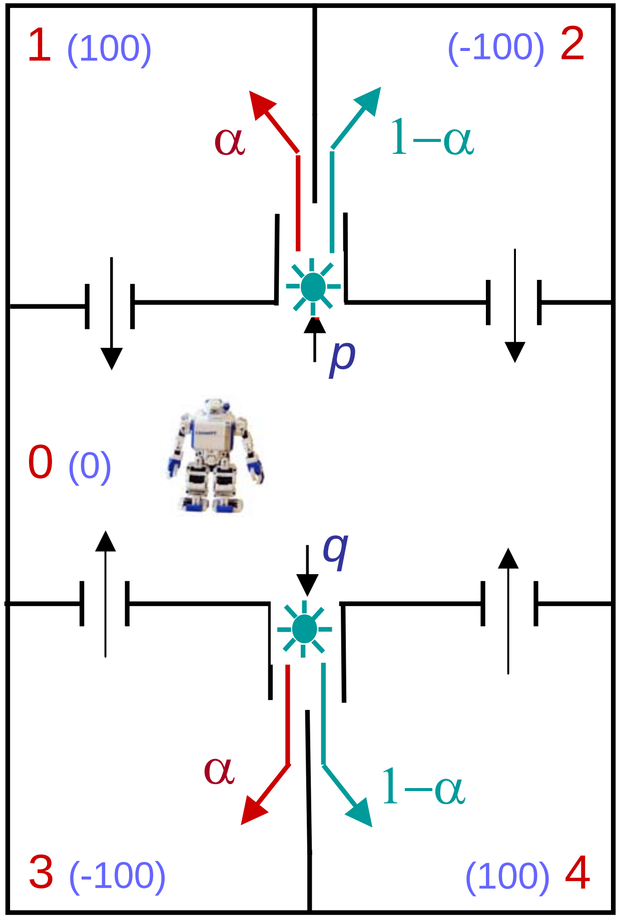
– Steady-state probability:

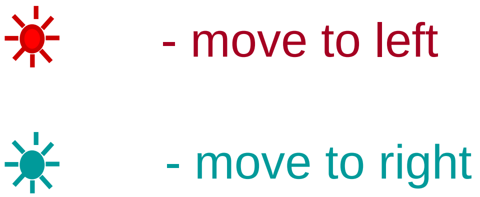
$\pi=(\pi(1), \pi(2),..,\pi(M))$.

$\pi(I-P)=0, \quad \pi e=1$

I:identity matrix, $e=(1,\ldots,1)^T$

# Control of Transition Probabilities



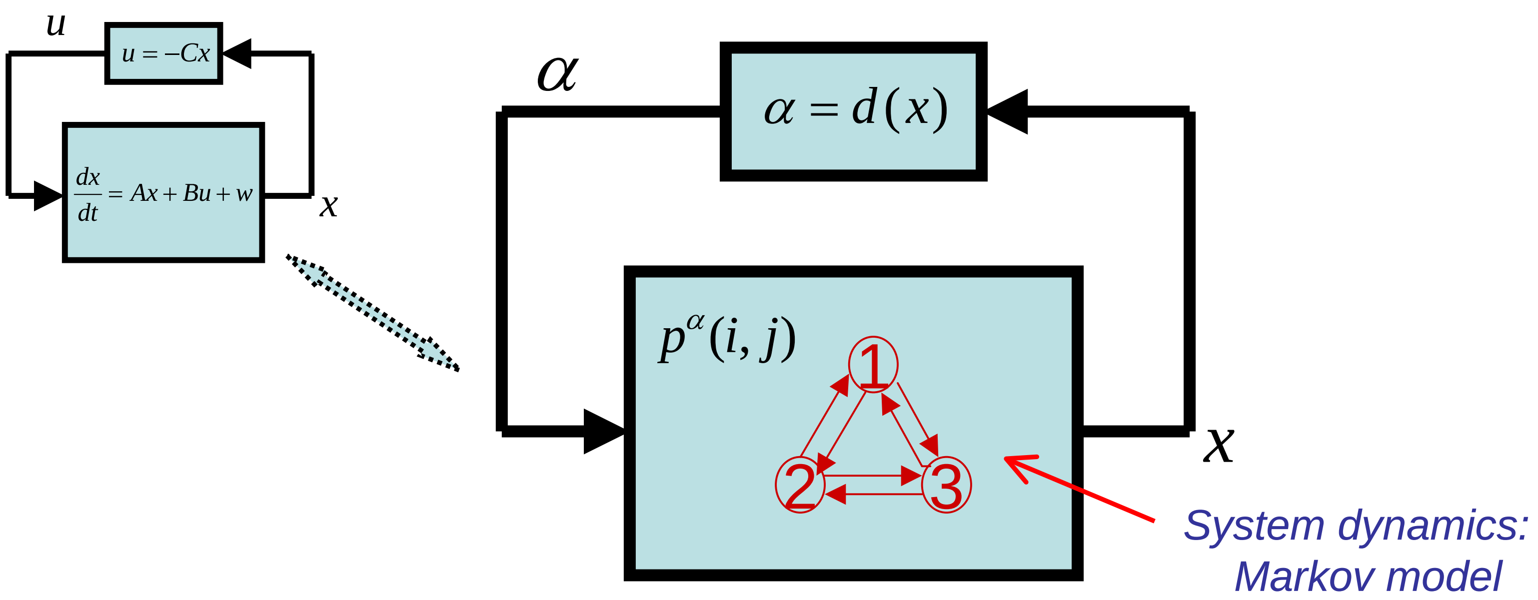




Turn on red with prob. $\alpha$

Turn on green with prob. $1-\alpha$

# Discrete-time Model (IV)
## - Markov decision processes (MDPs)

### - the Control Model




$\alpha$: *Action controls transition probabilities*

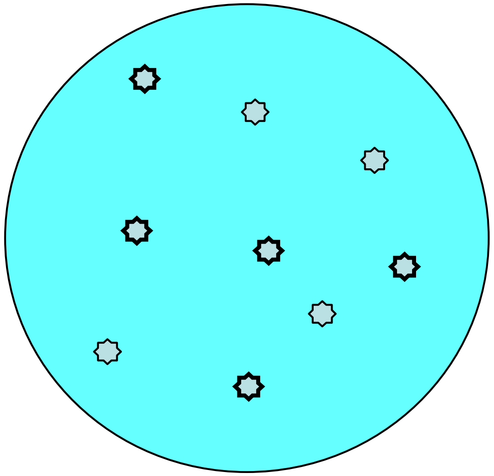
$p^{\alpha}(i,j)$: *governs the system dynamics*

$\alpha=d(x)$: *policy (state based)*

*Performance depend on policies,* $\pi^d$, $\eta^d$, *etc*

$$\eta^d = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} f(X_t^d)$$

*Goal of Optimization:*
*Find a policy d that maximizes* $\eta^d$ *in policy space*



- Li
- The policy space is too large

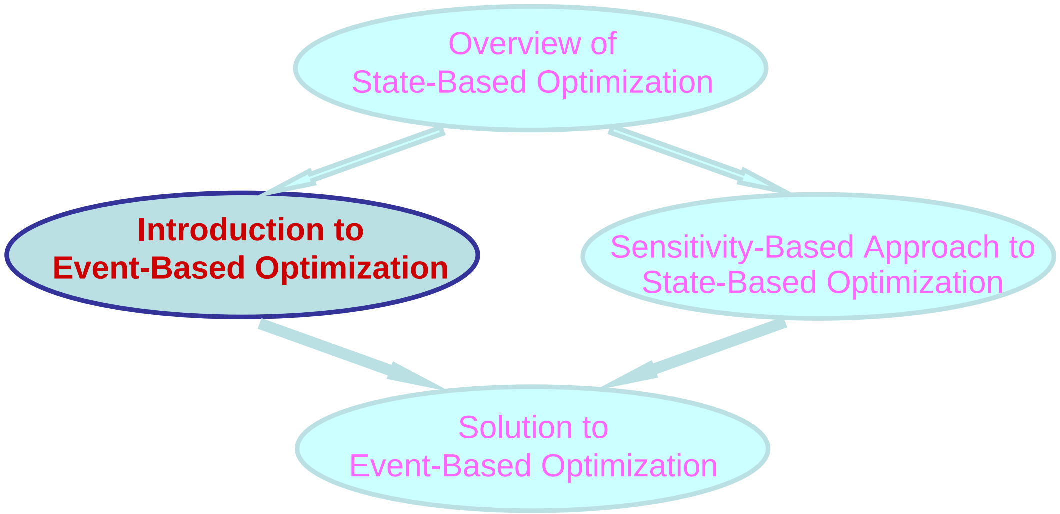
  M = 100 states, N=2 actions,

  $N^M = 2^{100} = 10^{30}$ policies

  (10GHZ ➔ $3 * 10^{12}$ years to count!)
- Special structures not utilized
- Different approaches:
  - Policy iteration (PI), value iteration
  - Perturbation analysis (PA)
  - Reinforcement learning
  - Stochastic control theory
  - ……

0. Review: Optimization Problems (state-based policies)

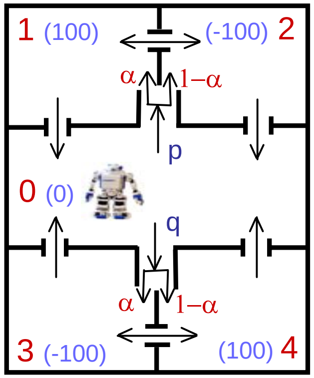
1. Event-Based Optimization
   - Limitation of the state-based formulation
   - Events and event-based policies
   - Event-Based Optimization

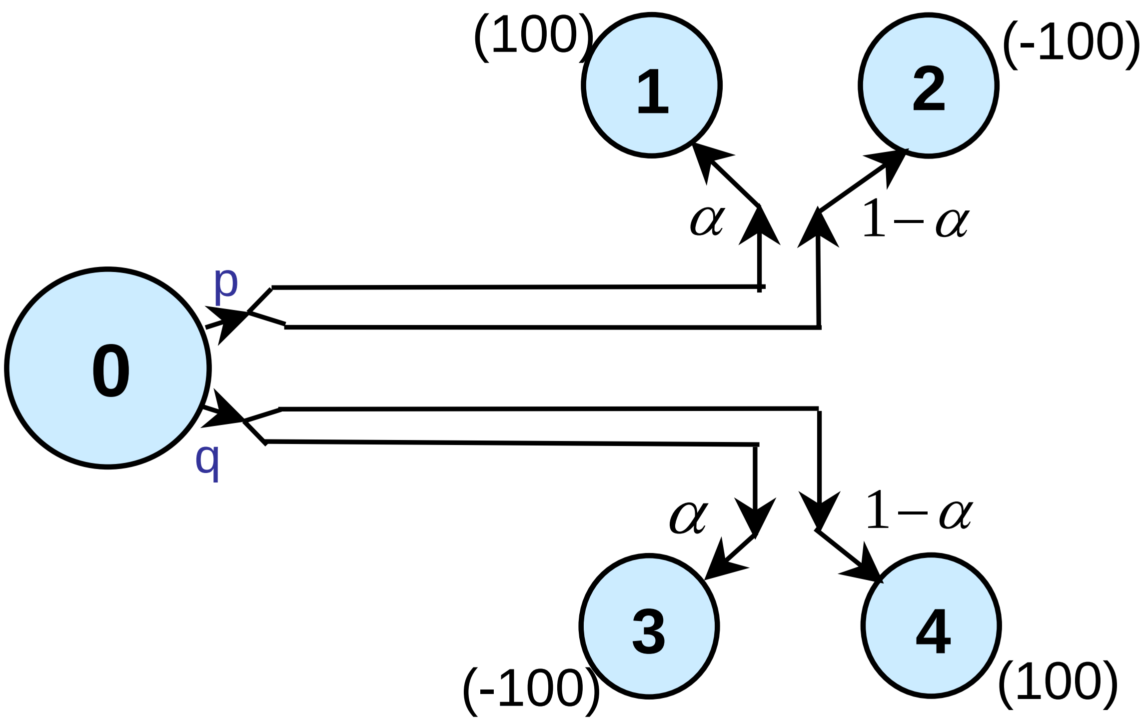


# Limitation of State-Based Formulation (II)

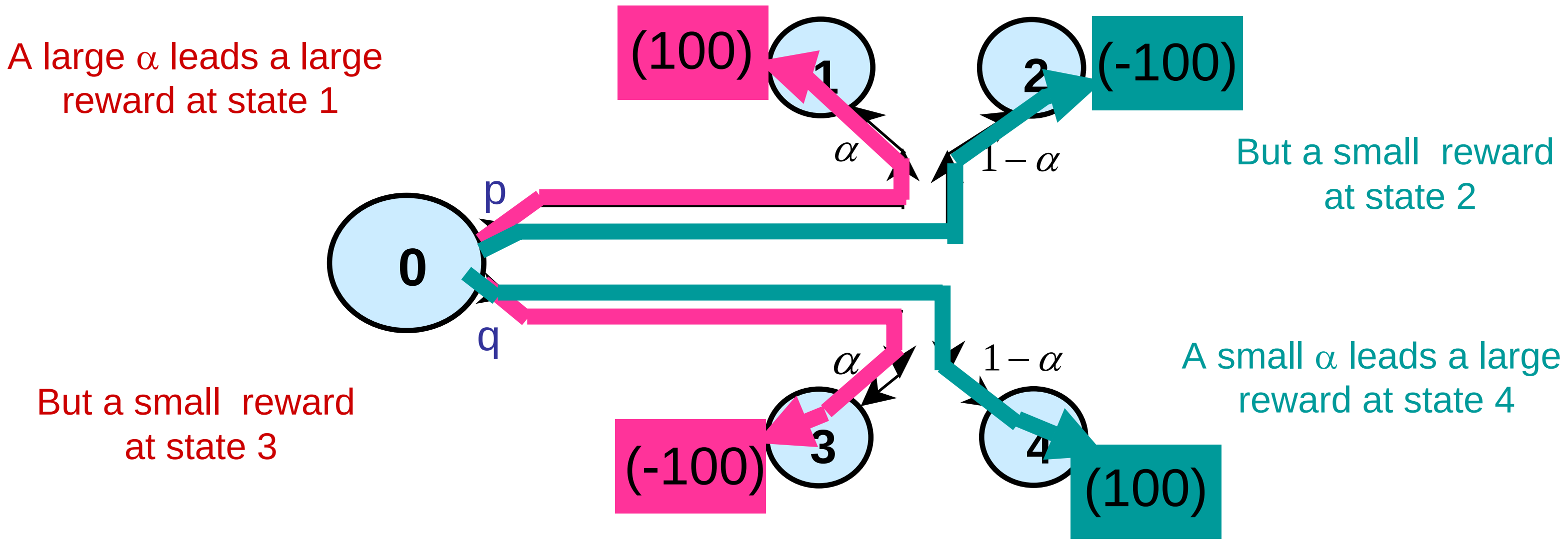
## Example: Random walk of a robot

Choose $\alpha$ to maximize the average performance



# Limitation of State-Based Formulation (III)

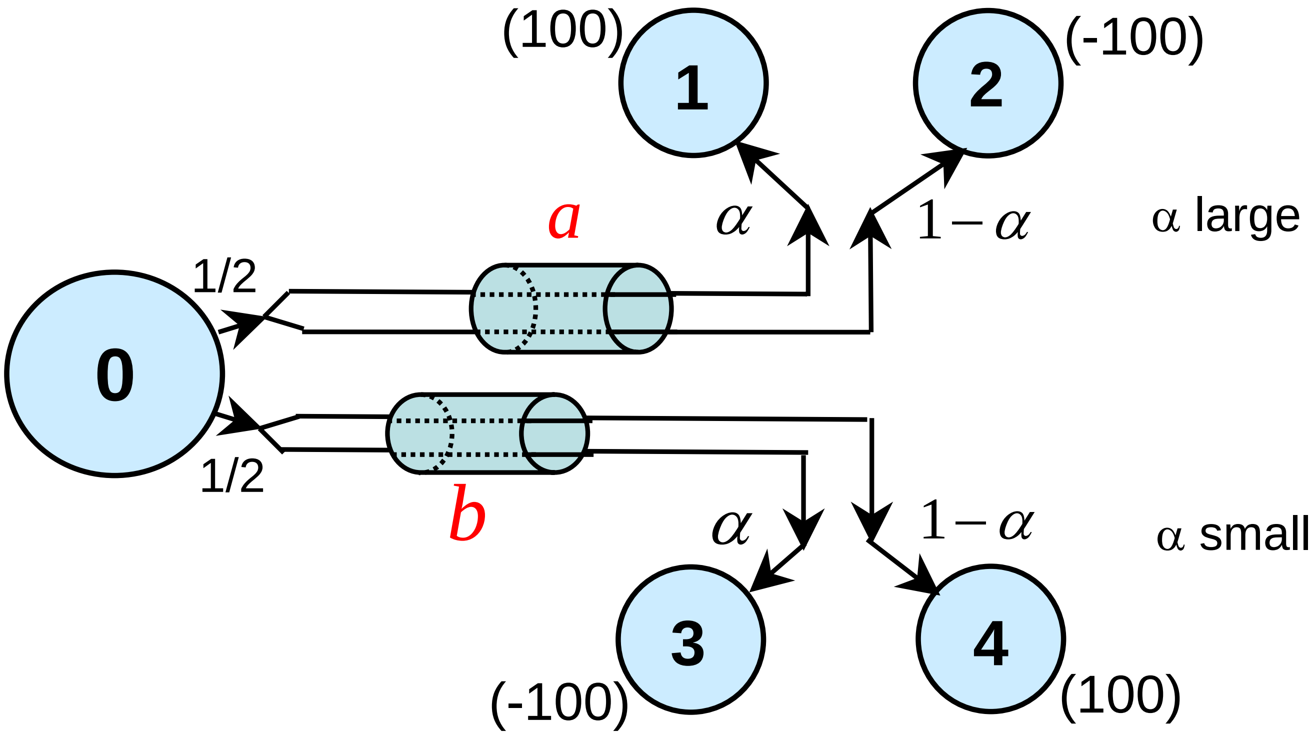



*Transition probabilities:*

| | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 0 | $p\alpha$ | $p(1-\alpha)$ | $q\alpha$ | $q(1-\alpha)$ |

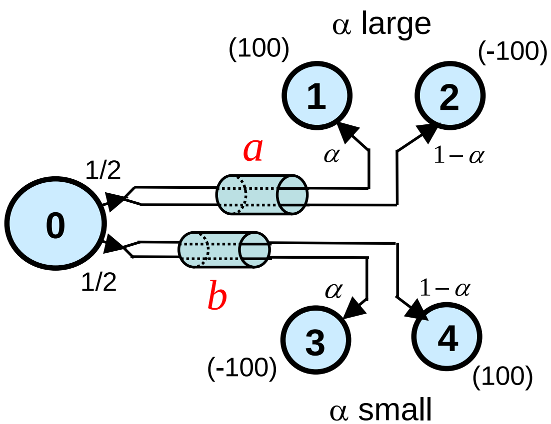
- At state 0,
  - ➔ if moves top, α needs to be as large as possible
  - ➔ if moves down, α needs to be as small as possible
- Let p = q = 1/2,
  - ➔ Average perf in next step = 0, no matter what α you choose (best you can do with a state-based model)

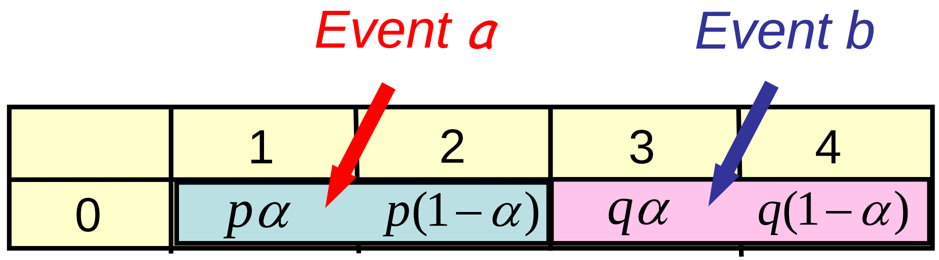
# We can do better!




- Group two up transitions together as an event "a" and two down transitions as event "b".
- When "a" happens, choose the largest $\alpha$,
  When "b" happens, choose the smallest $\alpha$.
- Average performance $= 100$, if $\alpha=1$.

# Events and Event-Based Policies



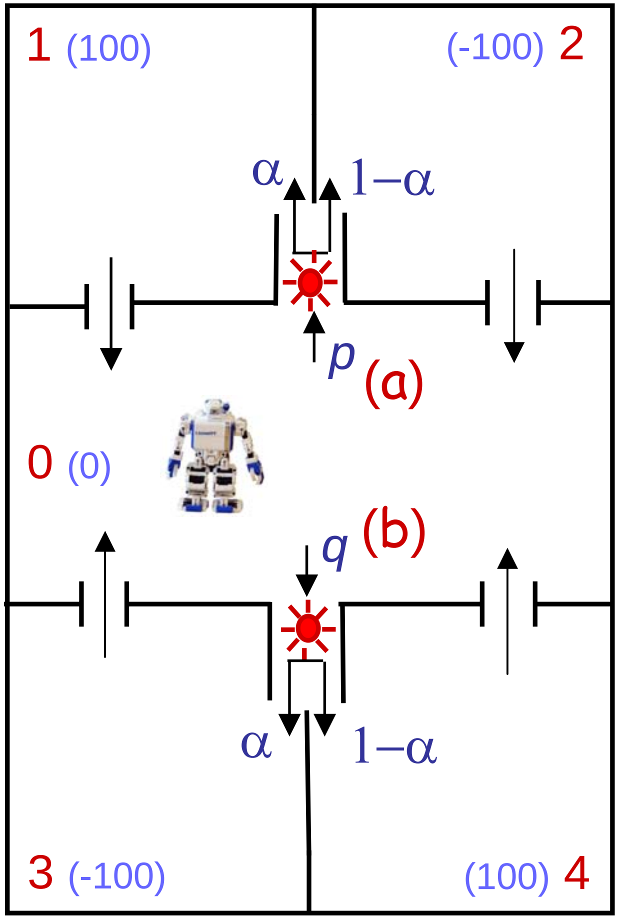




- An event is defined as a set of state transitions
- Event-based optimization:
  - May lead to a better performance than the state-based formulation
  - MDP model may not fit:
    - Only controls a part of transitions
    - An event may consist of transitions from many states
  - May reflect and utilize special structures
- Questions:
  - Why it may be better?
  - How general is the formulation?
  - How to solve event-based optimization problems?

# Notations:

- A single transition <i,j>,
  i,j in S ={1,2, …, M}
- An event: a set of transitions,
  $2^M$ sets
  a = {<0,1>, <0,2>}
  b = {<0,3>, <0,4>}

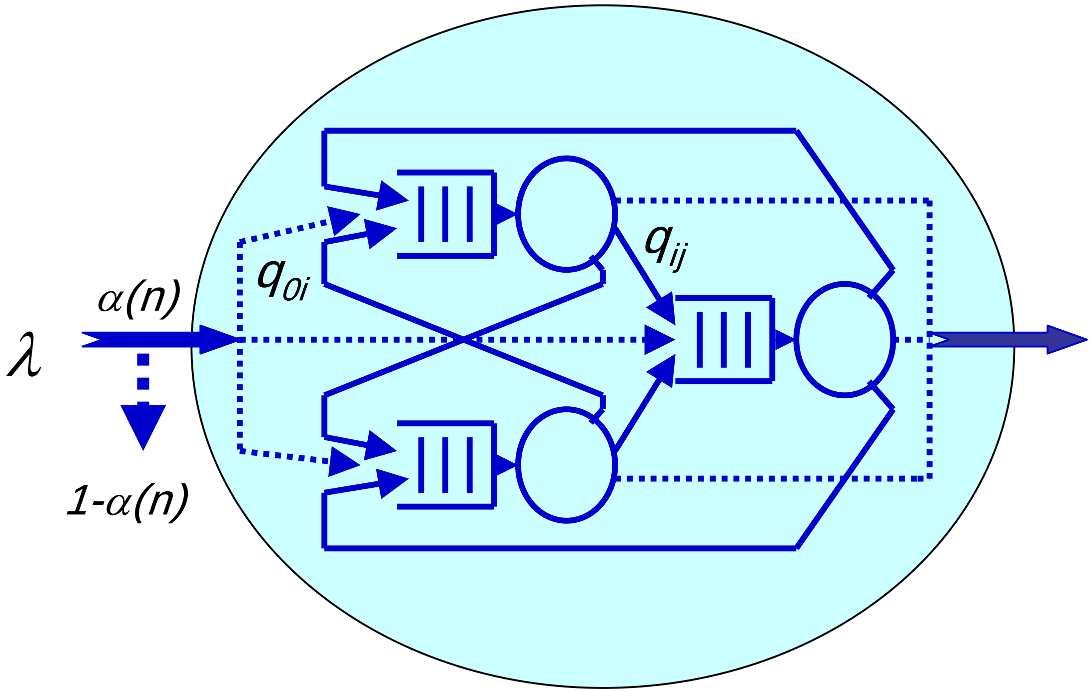
Physical interpretation




# Why it is better?

An event contains information about the future!
(compared with the state-based policies)

# How general is the formulation?




Admission control

*n: population*

*No. of customers in network*

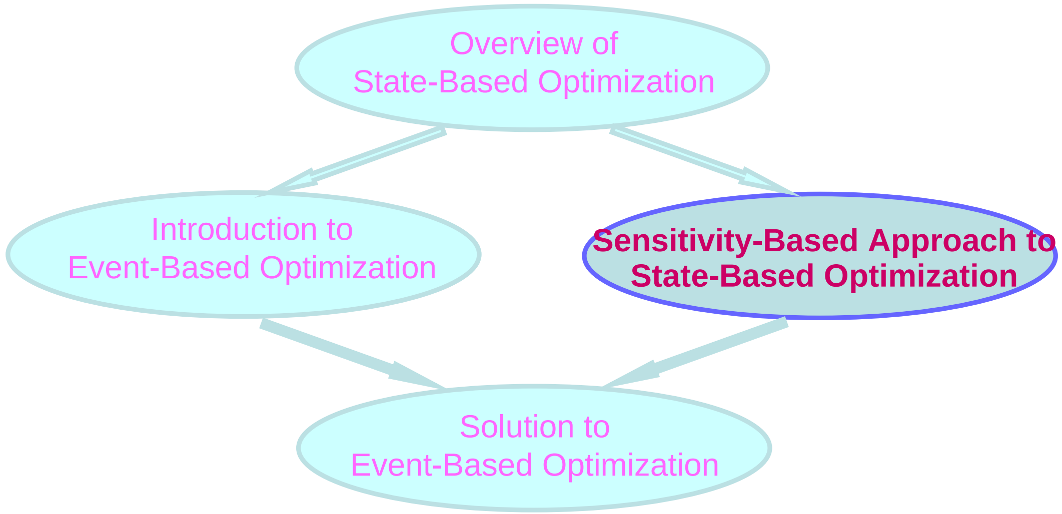
*$n_i$: No. of customers at server i*

*$\boldsymbol{n}=(n_1,\ldots,n_M)$: state*

*N: network capacity*

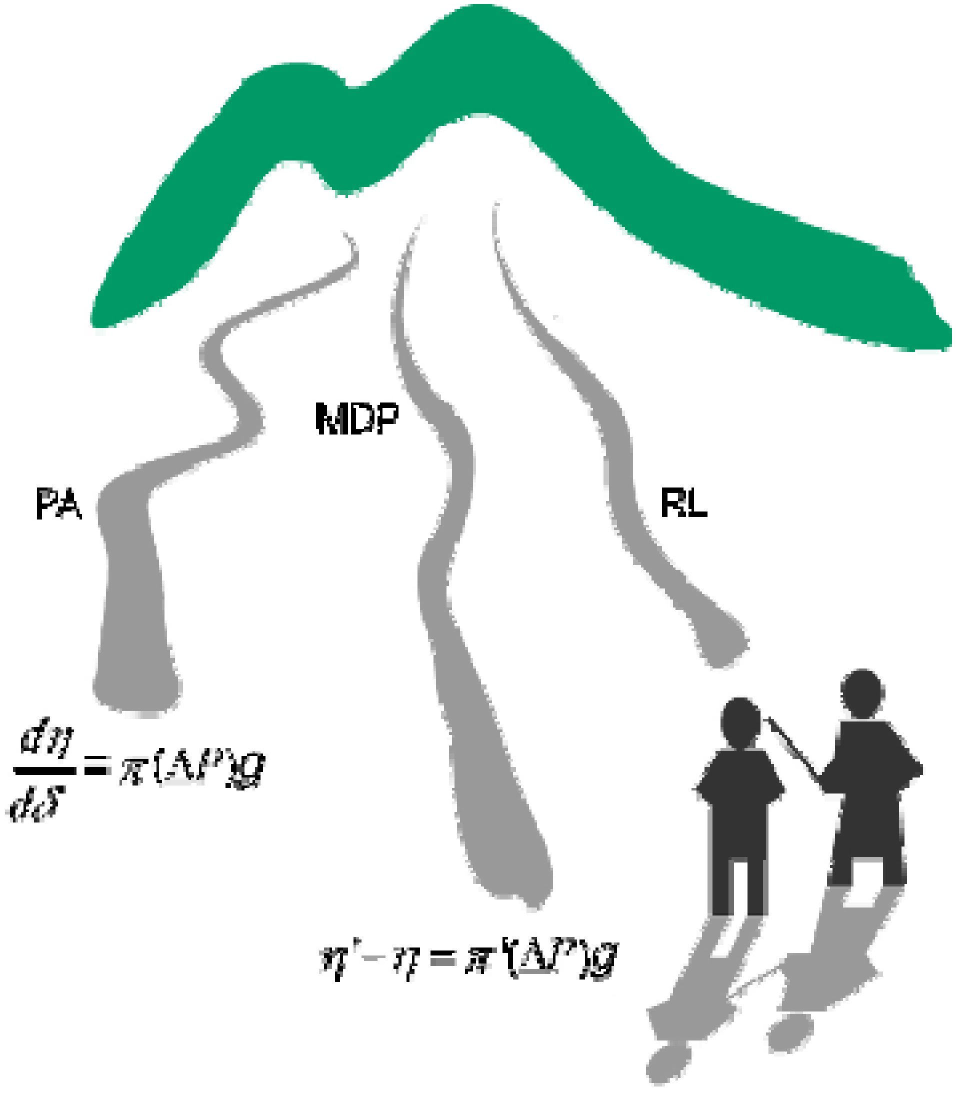
- Event: a customer arrival finding population n
- Action: accept or reject

  Only applies when an event occurs
- MDP does not apply: Same action is applied for different state with the same population



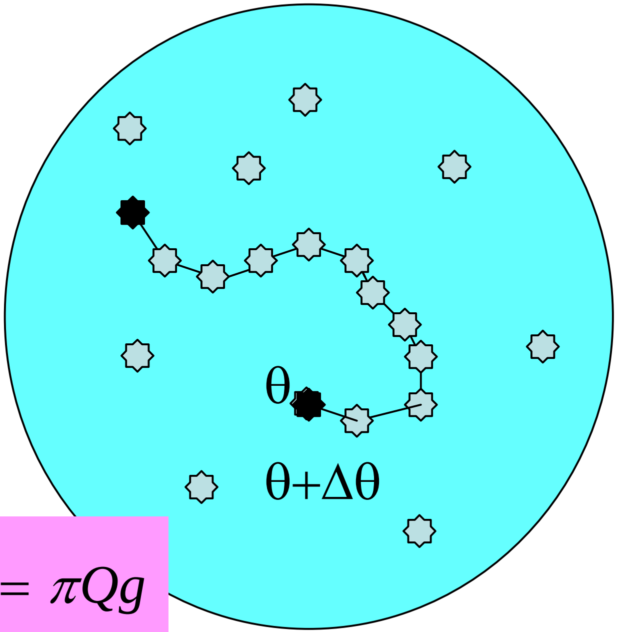

2. Sensitivity-Based Approach to Optimization
    - A unified framework for optimization
    - Extensions to event-based optimization
3. Summary

## An overview of the paths to the top of a hill

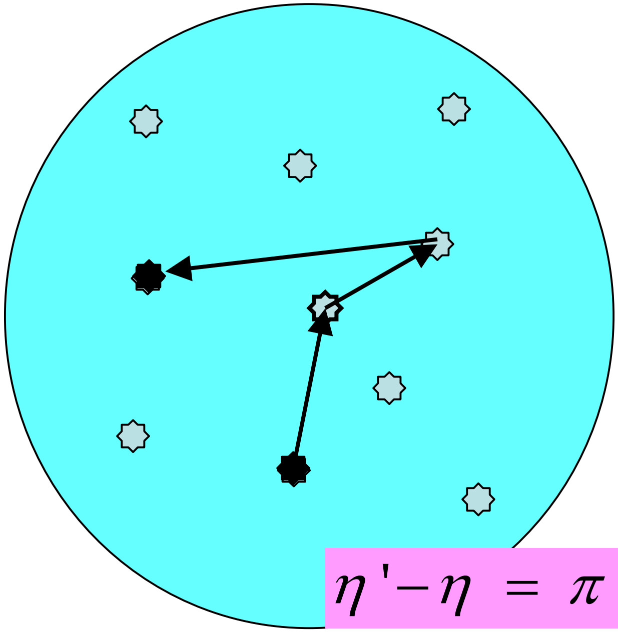


# A Sensitivity-Based View of Optimization

- Continuous Parameters (perturbation analysis)
- Discrete Policy Space (policy iteration)




$$\frac{d\eta}{d\delta} = \pi Q g$$



$$\eta' - \eta = \pi' Q g$$

$\eta$: *performance*
$\pi$: *steady-state prob*
*g: perf. potentials*
*Q=P'-P*

# Poisson Equation

*g(i) = potential contribution of state i (potential, or bias)*
*= contribution of the current state f(i)-$\eta$*
*+ expected long term contribution after a transition*

$$g(i) = f(i) - \eta + \sum_{j=1}^{M} p(i,j) g(j)$$

*In matrix (Poisson equation):*

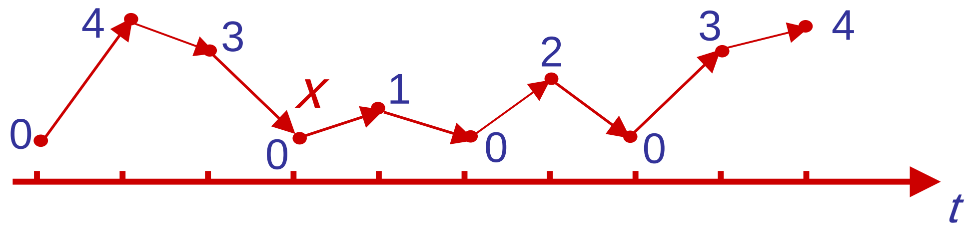
$$(I - P)g + \eta e = f$$

*Potential is relative: if g(i) is solution, i=1,..., M, so is g(i)+c, c: constant*

Physical interpretation:

$$g(i) = E\{\sum_{l=0}^{\infty}[f(X_l) - \eta] \mid X_0 = i\}$$

$g(4) \approx$ *average of* $\sum f(X_l)$ *starting from* $X_0 = 4$



# Two Sensitivity Formulas

For two Markov chains *P*, $\eta$, $\pi$ and *P'*, $\eta'$, $\pi'$, let *Q=P'-P*

*Performance difference:* $$\eta' - \eta = \pi' Q g = \pi'(P' - P) g$$

*One line simple derivation:* $\times \pi'$: $(I - P) g + \eta e = f$

*Performance derivative:* *P is a function of* $\theta$: *P*($\theta$)

$$\frac{d\eta(\theta)}{d\theta} = \pi \frac{dP(\theta)}{d\theta} g \quad = \quad \frac{d}{d\theta}[\pi P(\theta) g]$$

Derivative =average change in expected potential at next step

Perturbation analysis: choose the direction with the largest average change in expected potential at next step

# Policy Iteration

$$\eta' - \eta = \pi' Q g = \pi' (P' - P) g$$

1. *$\eta' > \eta$ if $P'g > Pg$ (Fact: $\pi' > 0$ )*

2. *Policy iteration:*
   *At any state find a policy P' with $P'g > Pg$*

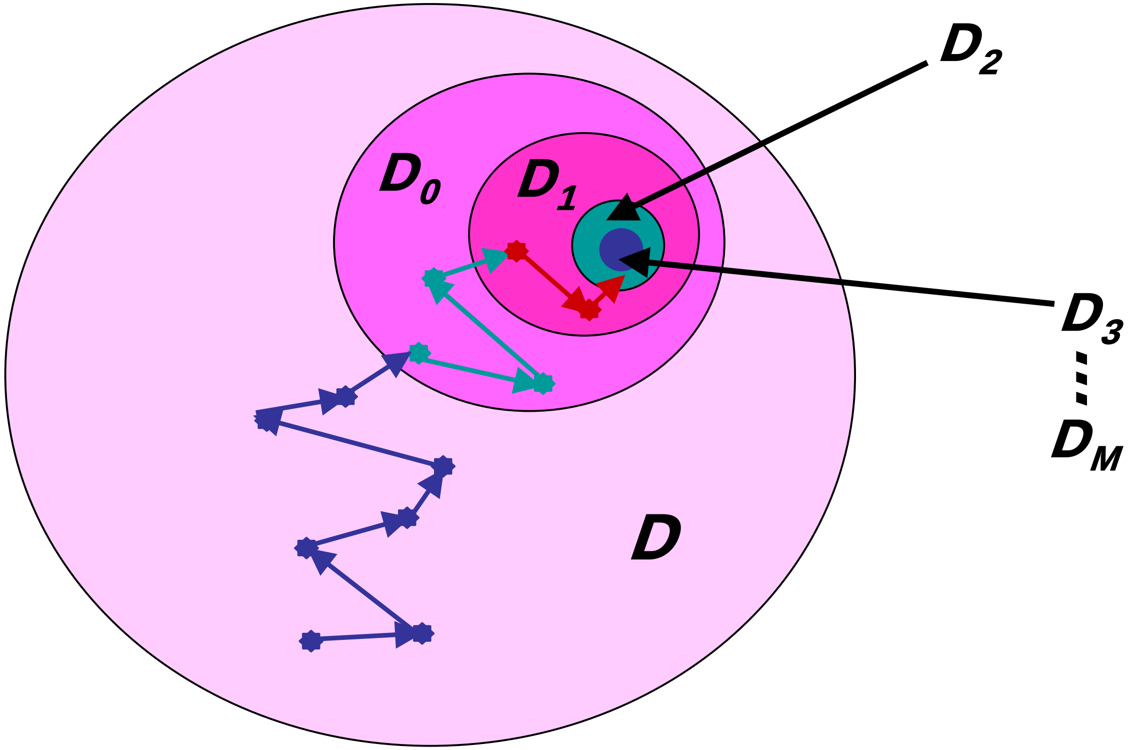
Policy iteration: Choose the action with largest changes in expected potential at next step

3. *Reinforcement learning*
   *(Stochastic approximation algorithms)*

# Mutli-Chain MDPs

## Perf./ Bias/ Blackwell Optimization

*With perf. difference formulas, we can derive a simple, intuitive approach without discounting*




***D:*** *Policy space*

***D$_0$:*** *Perf. optimal policies*

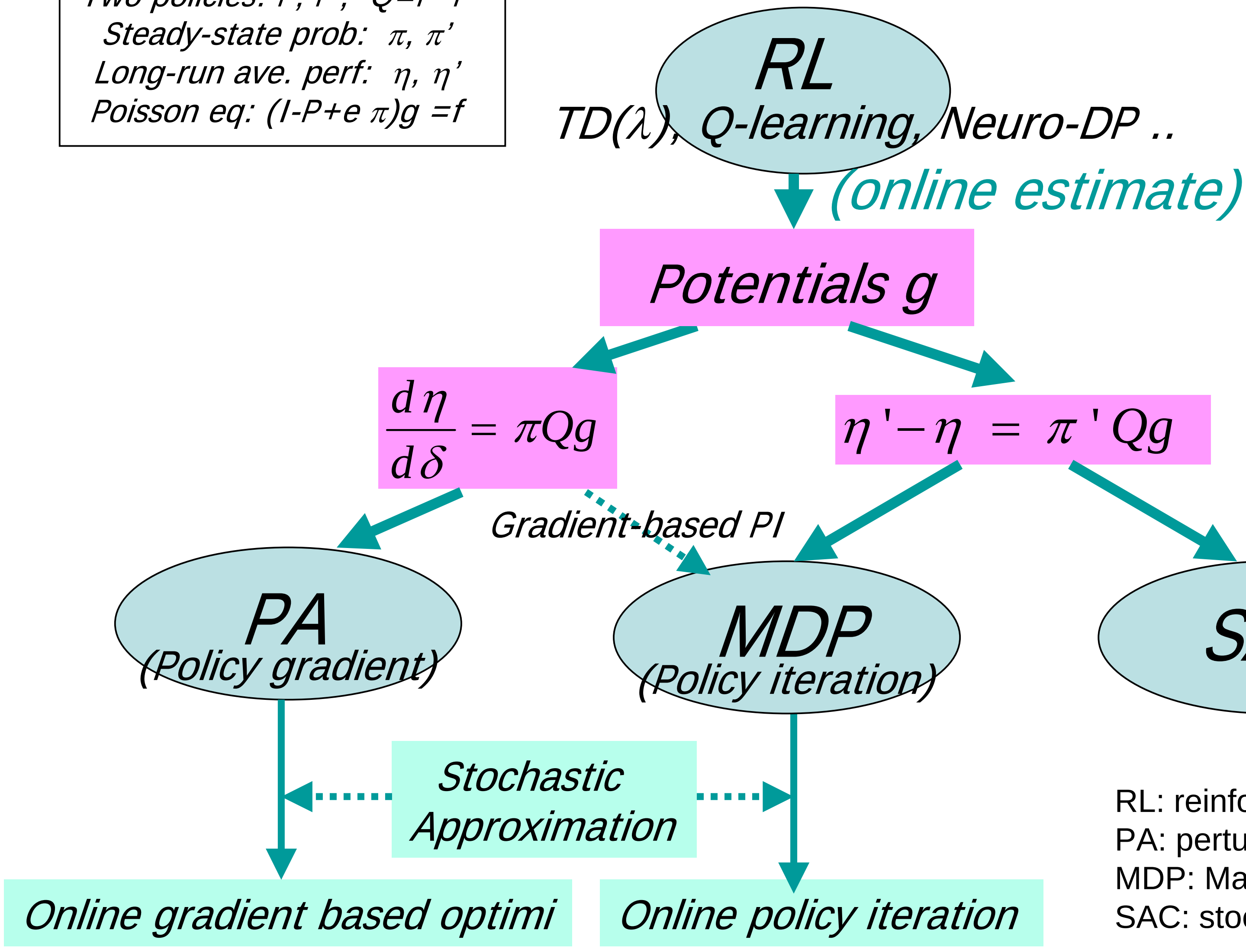
***D$_1$:*** *(1$^{st}$) Bias optimal policies*

***D$_2$:*** *2$^{nd}$ Bias optimal policies*

**……** ***D$_M$:*** *Blackwell optimal policies*

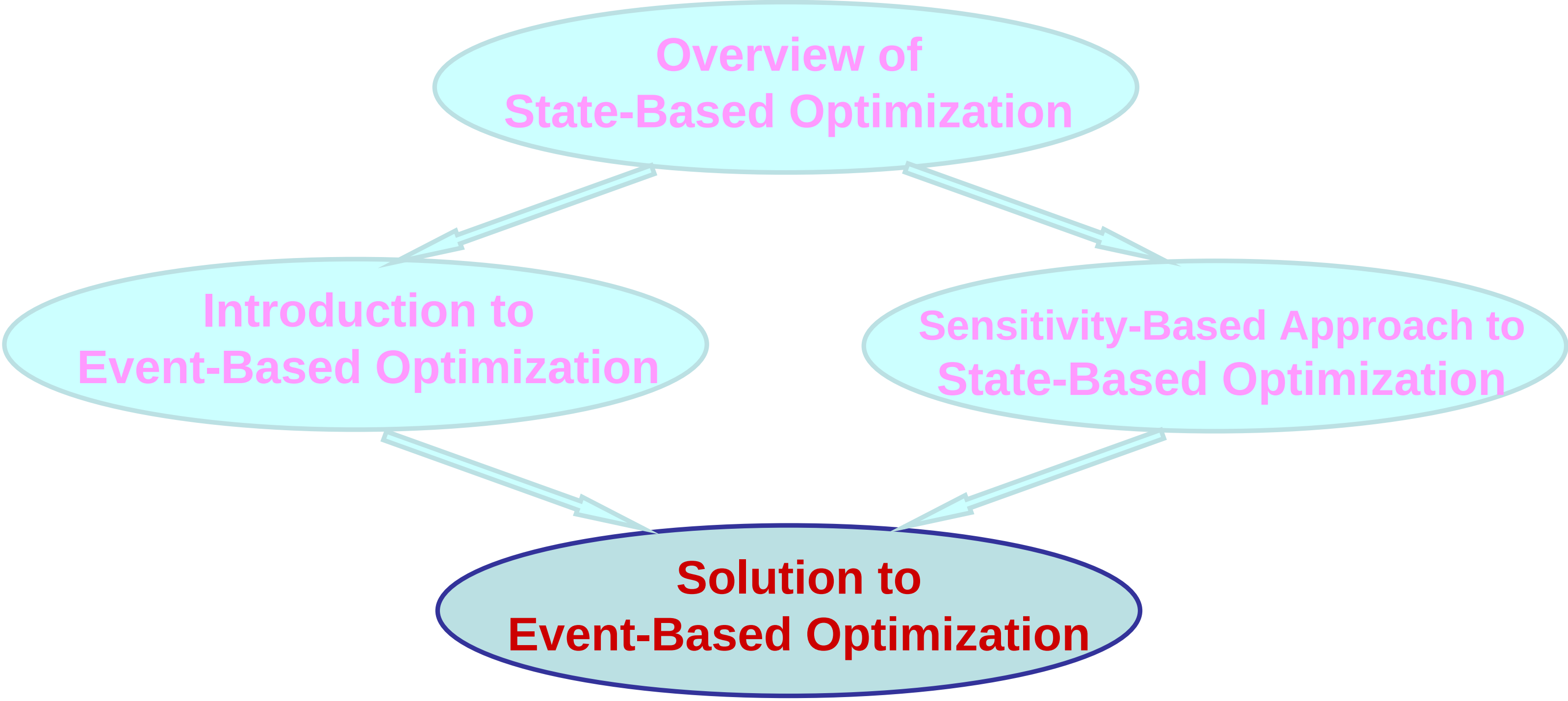
Bias measures transient behavior

*Two policies: P, P', Q=P'-P*
*Steady-state prob: $\pi$, $\pi'$*
*Long-run ave. perf: $\eta$, $\eta'$*
*Poisson eq: $(I-P+e\,\pi)g = f$*




RL: reinforcement learning
PA: perturbation analysis
MDP: Markov decision proc.
SAC: stochastic adaptive cont.

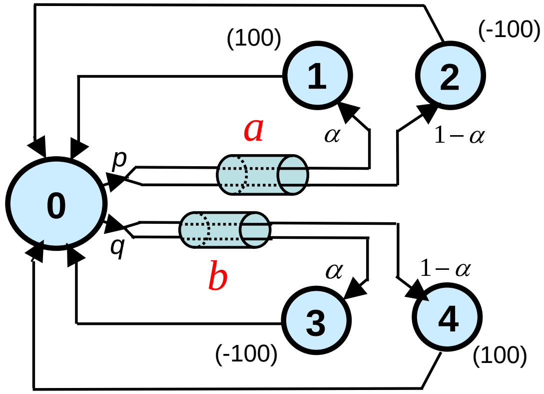
# A Map of the L&O World




Extension of the sensitivity-based approach
to event-based optimization

- Two sensitivity formulas
  - Performance derivatives
  - Performance differences
- PA & PI
  - PA: Choose the direction with largest average change in expected potential at next step
  - PI: Choose the action with largest changes in expected potential at next step
- Potentials are aggregated according to event structure

# Solution to Random Walker Problem




*Two policies:*

$$\alpha_a = d(a), \quad \alpha_b = d(b)$$

$$\alpha'_a = d'(a), \quad \alpha'_b = d'(b)$$

1. *Performance diff:*

$$\eta' - \eta = \pi'(a)[(\alpha_a' - \alpha_a)g(a)] + \pi'(b)[(\alpha_b' - \alpha_b)g(b)]$$

$$g(a) = g(1) - g(2) \quad g(b) = g(3) - g(4)$$

π'(a), π'(b): perturbed steady-state prob. of events a and b

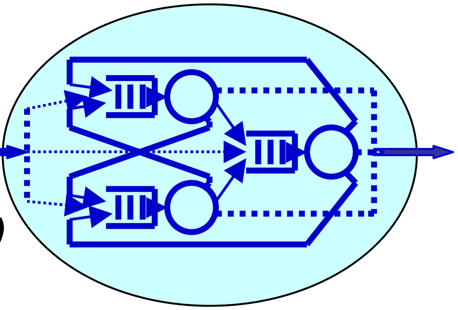
Choose the action with the largest changes
In expected potential at next step
*g(a), g(b)* aggregated

2. *Performance deriv:*

*Continuous with θ:* $\alpha_a(\theta)$, $\alpha_b(\theta)$

$$\frac{d\eta_\theta}{d\theta} = \pi_\theta(a)\frac{d\alpha_a(\theta)}{d\theta}[g_\theta(1) - g_\theta(2)] + \pi_\theta(b)\frac{d\alpha_b(\theta)}{d\theta}[g_\theta(3) - g_\theta(4)]$$

# Solution to Admission Control Problem




*Two policies: b(n) and b'(n)*

1. *Performance diff:*

$$\eta' - \eta = \sum_{n=0}^{N-1} \{ p'(n) [\alpha'(n) - \alpha(n)] d(n) \}$$

*p(n): prob. of arrival finding n cust.*

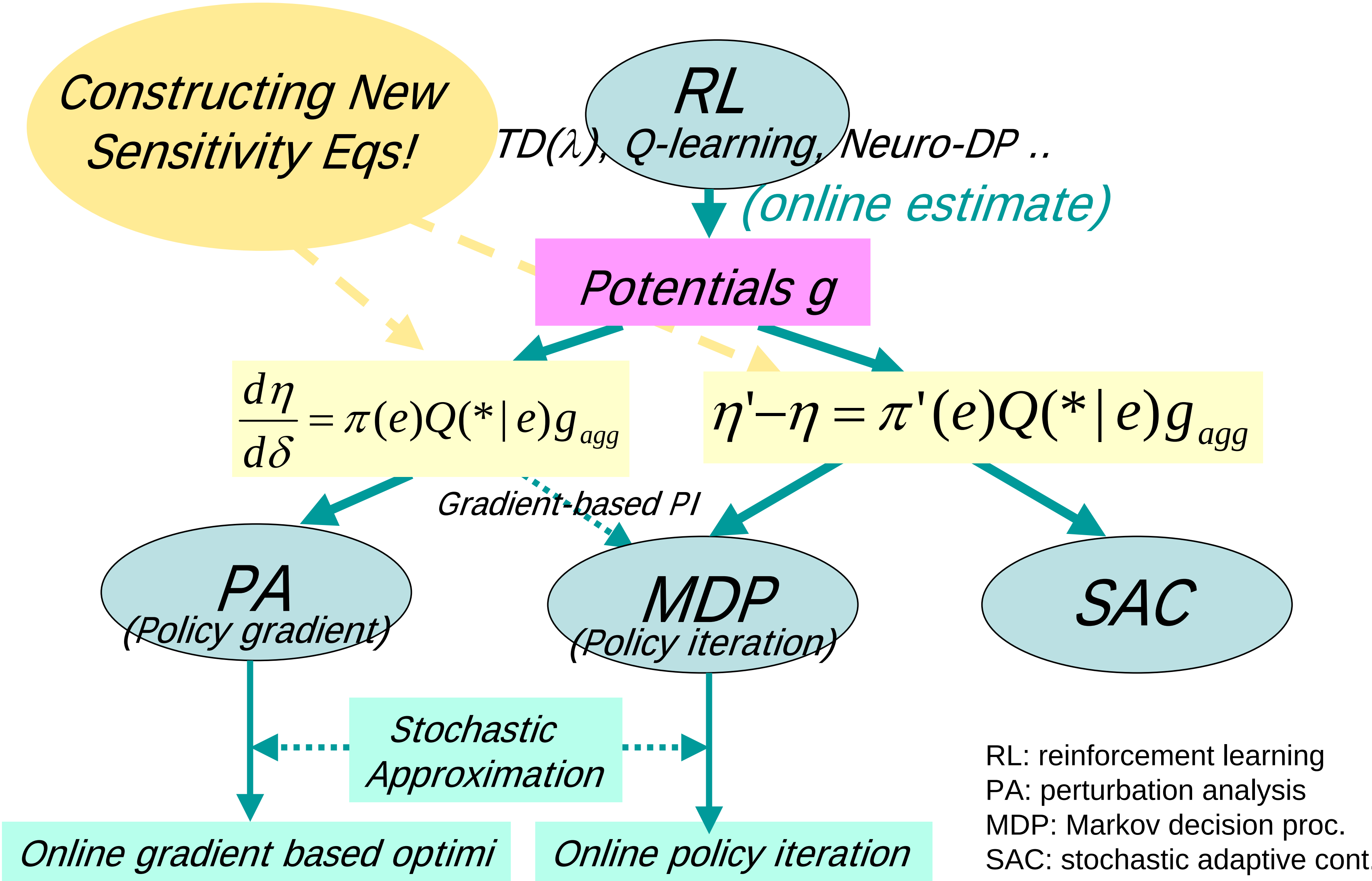
*Potential aggregation:*

$$d(n) = \frac{1}{p(n)} \{ \sum_{i=1}^{M} q_{0i} [ \sum_{\sum n_i = n} p(\bar{n}) g(\bar{n}_{+i}) ] - \sum_{\sum n_i = n} p(\bar{n}) g(\bar{n}) \}$$

Choose the action with the largest changes
In expected potential at next step
*d(n):* aggregated potential

2. *Performance deriv:*

$$\frac{d\eta}{d\delta} = \sum_{n=0}^{N-1} \{ p(n) [\alpha'(n) - \alpha(n)] d(n) \}$$




*Sensitivity-Based Approaches to Event-Based Optimization*

# *Summary*

# Advantages of the Event-Based Approach

1. *May have better performance*
2. *# of aggregated potentials d(n): N*

   *may be linear in system*
3. *Actions at different states are correlated*

   *standard MDPs do not apply*
4. *Special features captured by events*

   *action depends on future information*
5. *Opens up a new direction*

   *to many engineering problems*

   *POMDPs: observation y as event*

   *hierarchical control: mode change as event*

   *network of networks: transitions among subnets as events*

   *Lebesgue Sampling*

# Sensitivity-Based View of Optimization

1. *A map of the learning and optimization world:*

   *Different approaches can be obtained from two sensitivity equations*

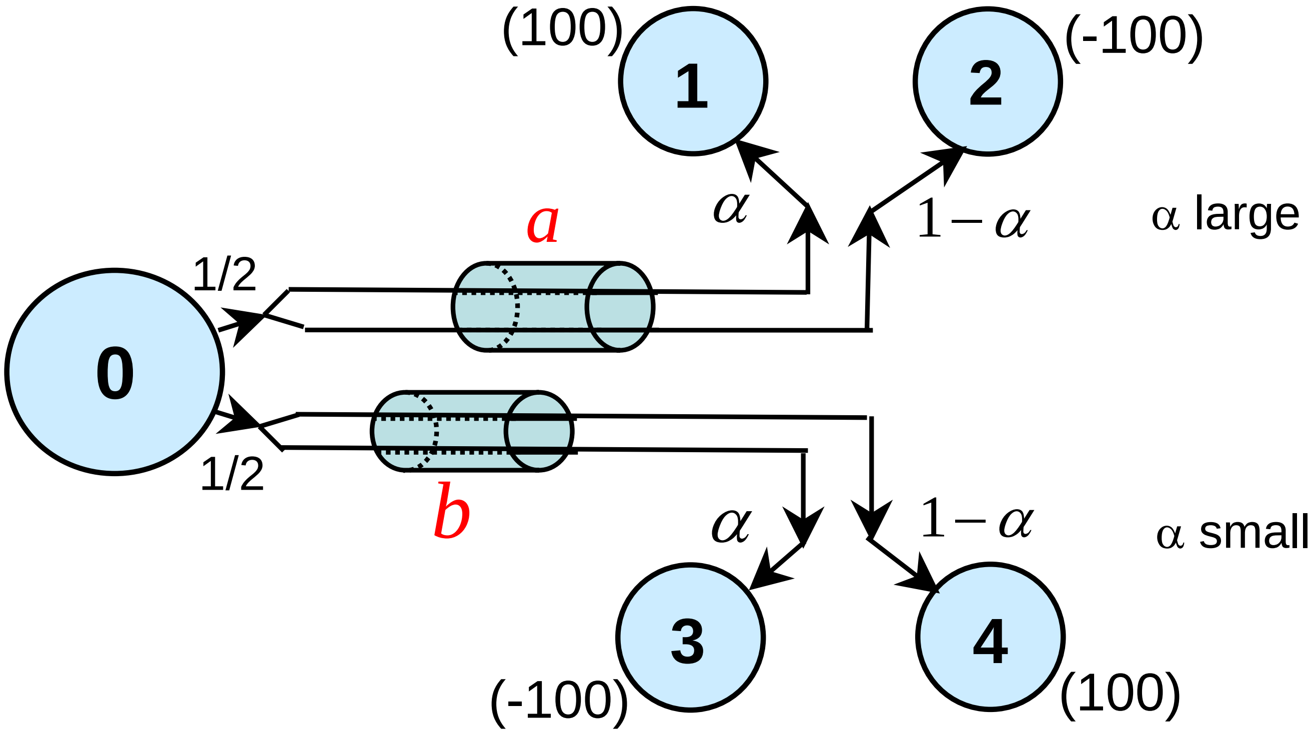
2. *Extension to event-based optimization*

   *Policy iteration, perturbation analysis*
   *reinforcement learning, time aggregation*
   *stochastic approximation, Lebesgue sampling*
   *……*

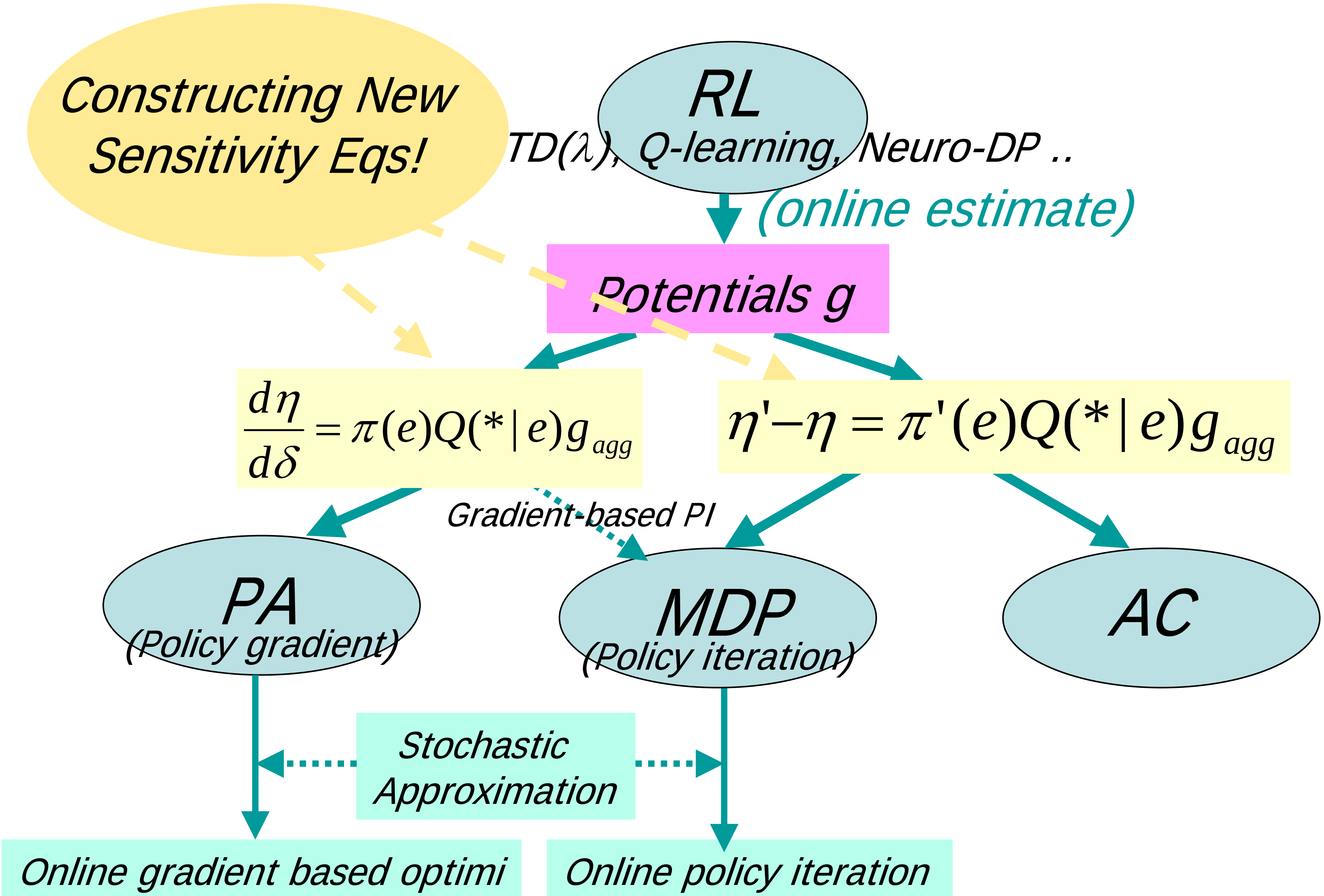
3. *Simpler and complete derivation for MDPs*

   *Multi-chains, different perf. criteria*
   *Average performance with no discounting*
   *N-bias optimality – Blackwell optimality*

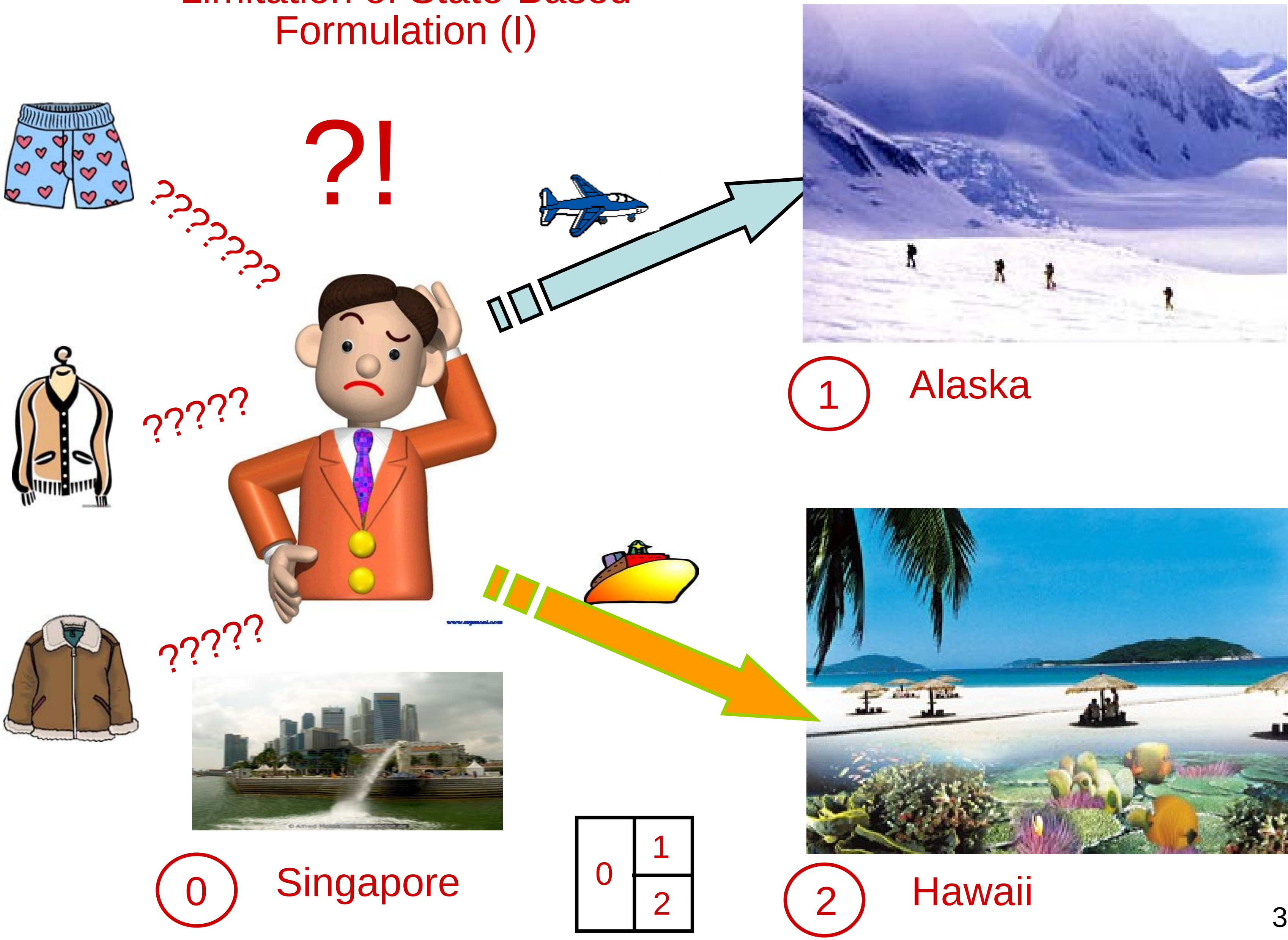
# Pictures to Remember (I)

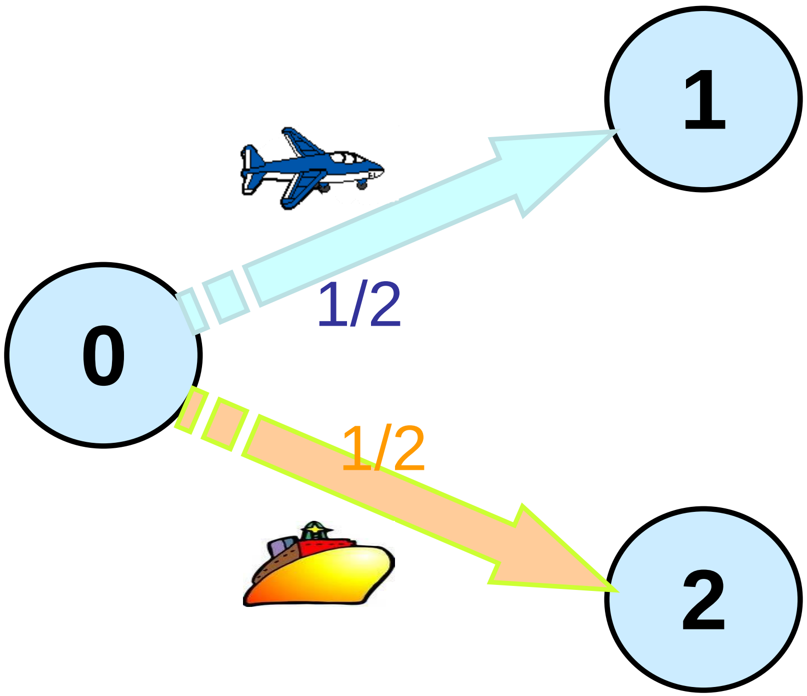


# Pictures to Remember (II)



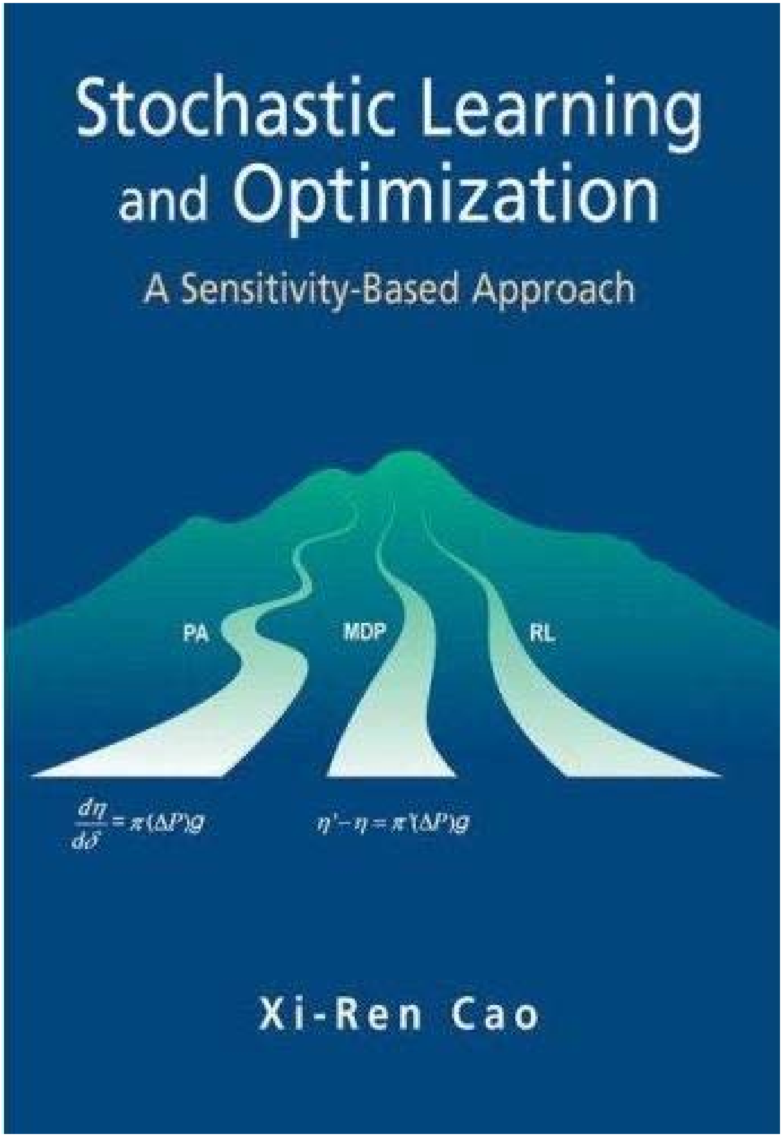
# Limitation of State-Based Formulation (I)




1
1/2
0
1/2
2

## Limitation of State-Based Formulation (I)


# *Thank You!*




*Xi-Ren Cao:*

*Stochastic Learning and Optimization - A Sensitivity Based Approach*

*9 Chapters, 566 pages*
*119 Figures, 27 Tables,*
*212 homework problems*

*Springer*
*October 2007*