State-dependent Limited Processor Sharing - Approximations and Optimal Control

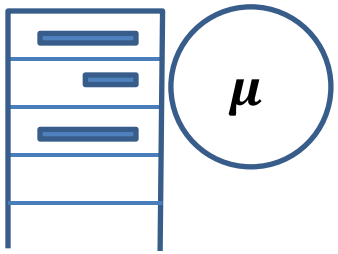
Varun Gupta

University of Chicago

Joint work with: Jiheng Zhang (HKUST)

State-dependent Limited Processor Sharing

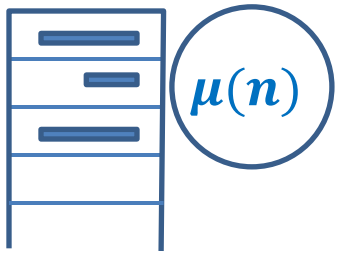
Processor Sharing

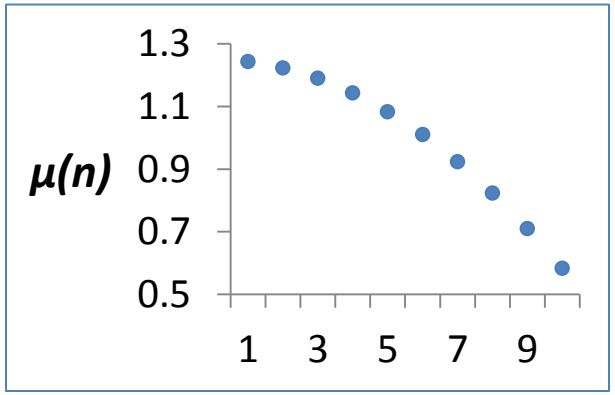


n jobs at server $\Rightarrow$ service rate μ/n per job

State-dependent Limited Processor Sharing

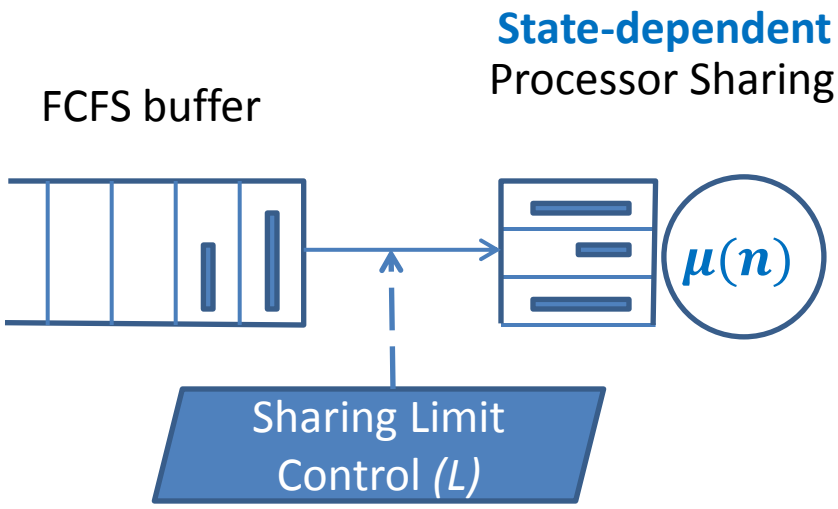
State-dependent
Processor Sharing

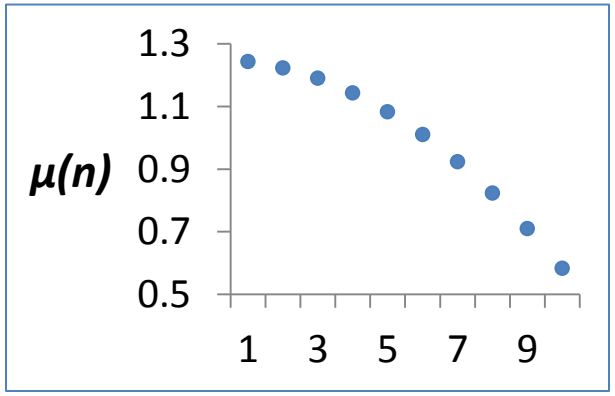




n jobs at server $\Rightarrow$ service rate ***μ(n)/n*** per job

State-dependent Limited Processor Sharing



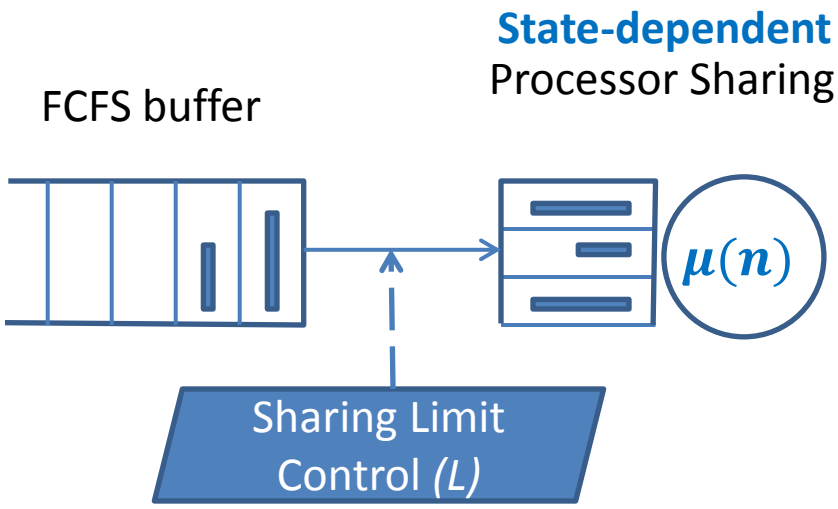


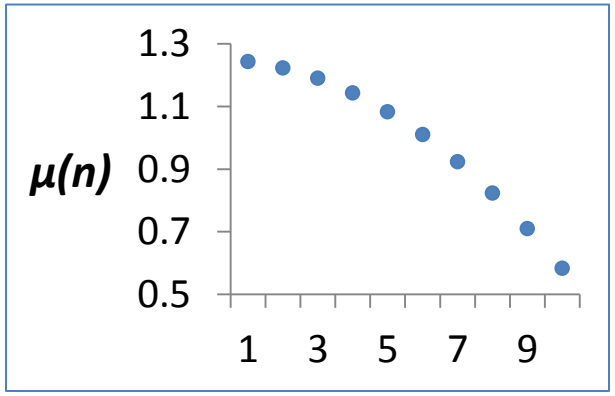
Motivation

- Congestion based friction – e.g., server thread-pool management
- Service systems with human agents

GOAL: Design of good control policies

State-dependent Limited Processor Sharing





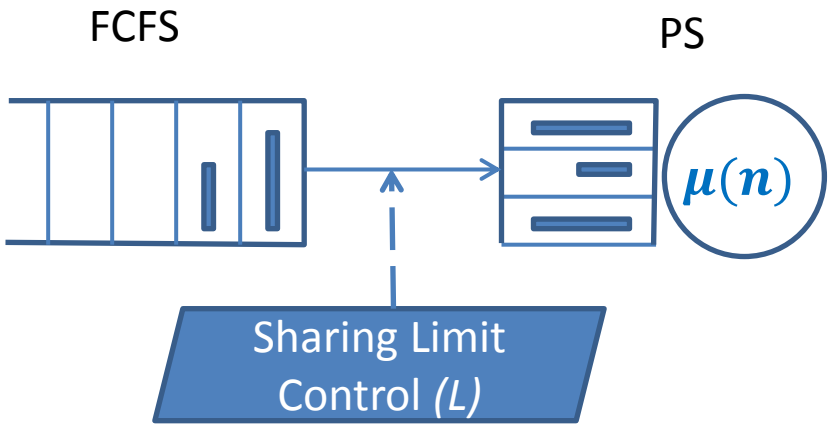
Straw-man: Set to maximum efficiency point (L^*)

Depends on arrival rate and variance of job size distribution

- Low L → FCFS dominates → good for low variance
- High L → PS dominates → good for high variance

State-dependent Limited Processor Sharing

GOAL : Design good control policies



Two classes

1. Static control policy
   - Sub-goal: Approximation for a given control *L*

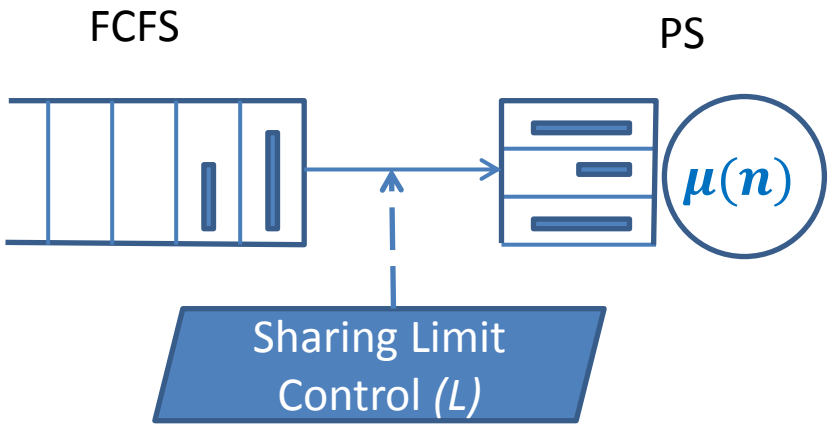
2. Dynamic control policy
   - Sub-goal: Numerical algorithm to solve a dynamic control problem

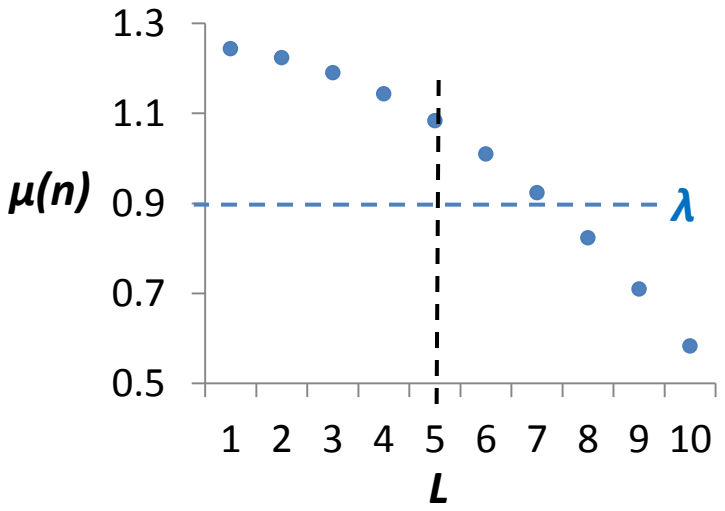
Setting: Analysis under diffusion scaling

1. Diffusion Approximation for Static policies

Approximation for static sharing limit

- General i.i.d. interarrivals
  - λ = arrival rate
  - c_a = coefficient of variation
- General i.i.d. service requirements
  - c_s = coefficient of variation
- L = static sharing limit





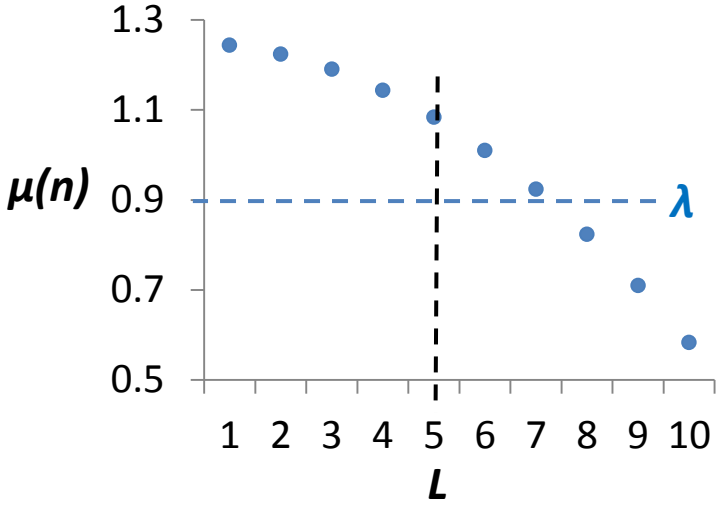
Q: A meaningful asymptotic regime?

That is, construct a sequence of LPS systems that faithfully approximates the original system

Q: A meaningful asymptotic regime?

Proposal #1: Conventional heavy traffic

$$\lambda^{(r)} \nearrow \mu(L)$$



Does not capture original system!

Q: A meaningful asymptotic regime?

Proposal #2:

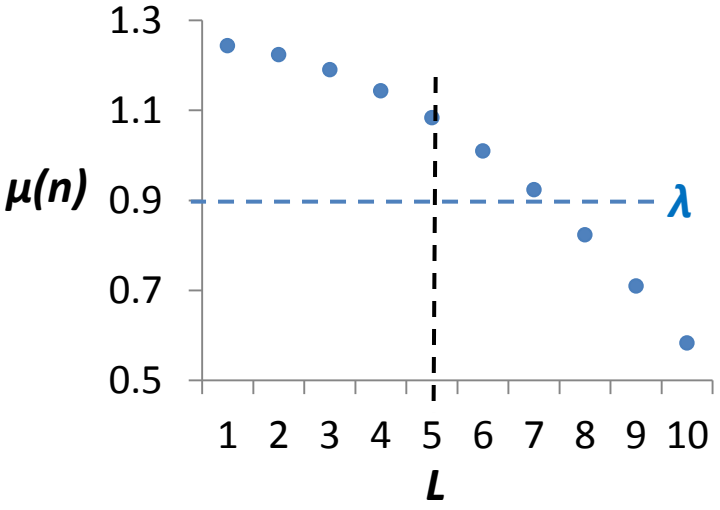
- Scale sharing limit

$$L^{(r)} = Lr$$

- "Stretch" service rate curve

$$\mu^{(r)}(rx) \to \hat{\mu}(x)$$

where $\hat{\mu}(x)$ is a continuous extension of $\mu(n)$



Also not faithful enough! In this example, system gets stuck at 0.

An axiomatic approach for state-dependent systems

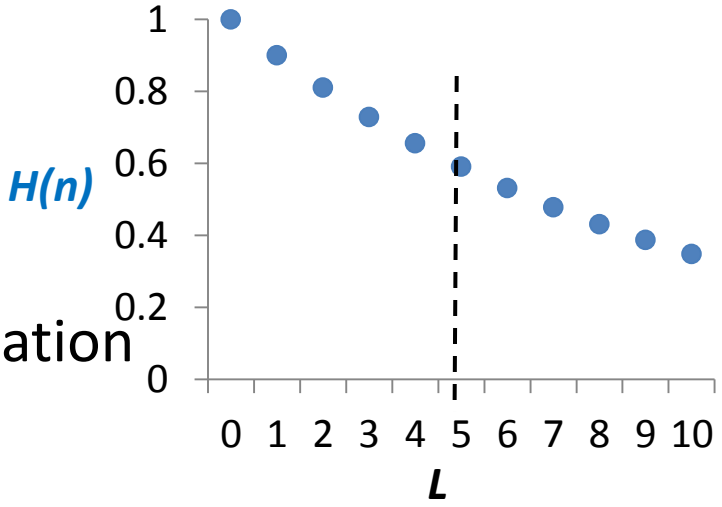
IDEA: Define what we mean by “faithful”
And reverse engineer the asymptotic scaling

Proposal

- Feed the original system M/M/ input
- Let

$$H(n) = Prob(N \geq n)$$

and $\widehat{H}$ a continuous differentiable interpolation



An axiomatic approach for state-dependent systems

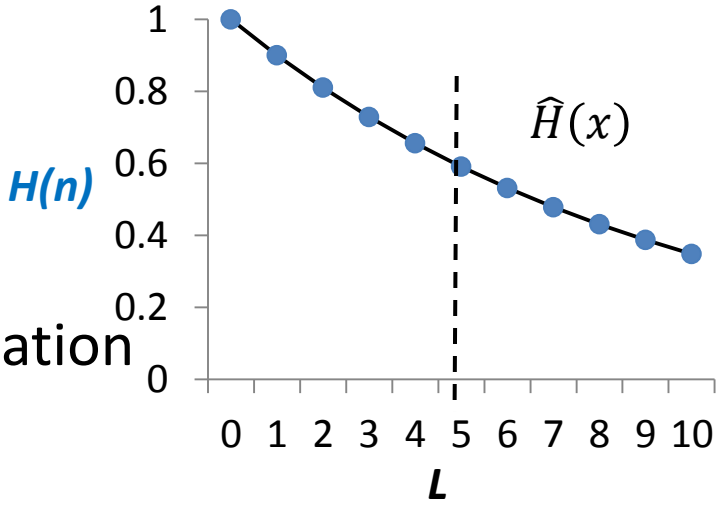
IDEA: Define what we mean by "faithful"
And reverse engineer the asymptotic scaling

Proposal

- Feed the original system M/M/ input
- Let

$$H(n) = Prob(N \geq n)$$

and $\widehat{H}$ a continuous differentiable interpolation

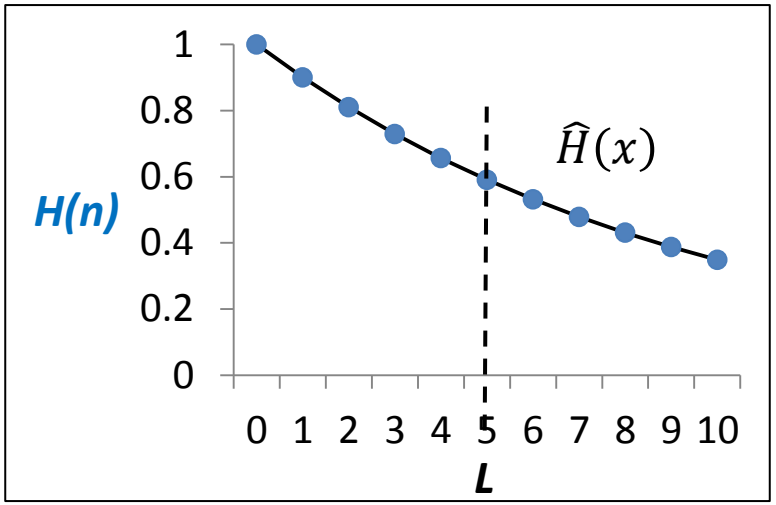


SCALING: In the r^{th} system

- $L^{(r)} = Lr$
- Under M/M/ input: $H^{(r)}(rx) \to \widehat{H}(x)$

SCALING: In the r^{th} system

- $L^{(r)} = Lr$
- Under M/M/ input: $H^{(r)}(rx) \to \widehat{H}(x)$



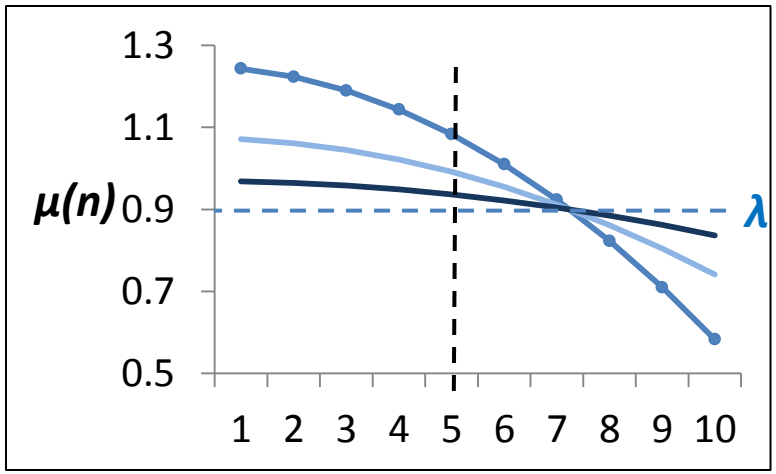
- We guarantee a limit that depends on the entire $\mu(n)$

- Reverse engineered service rates:

$$\lim_{r\to\infty} r\left(\lambda - \mu^{(r)}(rx)\right) = \lambda \frac{d\log(-\widehat{h}(x))}{dx} \doteq -\theta(x)$$

"drift function"

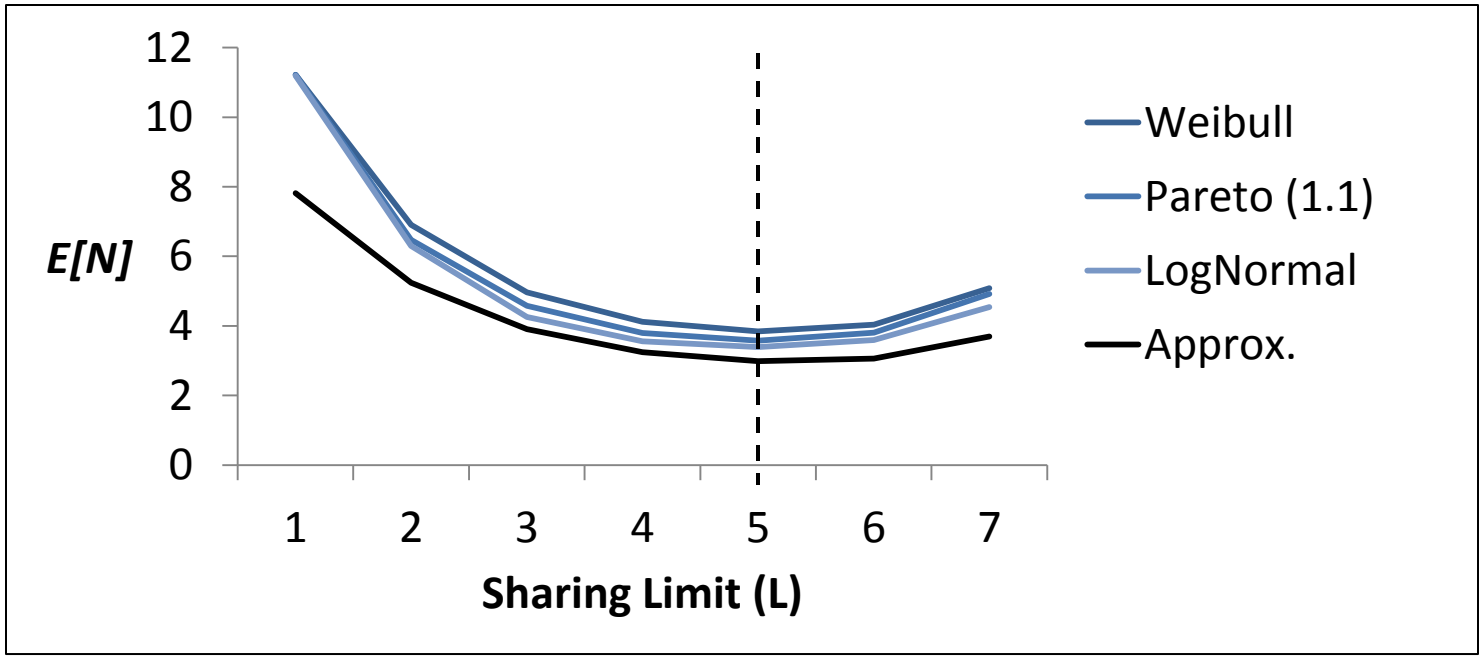
- Entire $\mu^{(r)}(\cdot)$ curve collapses to λ at rate $1/r$



Final Approximation for mean number in system:

$$E[N] \approx \frac{\sum_{n=0}^{\infty}(n \wedge L)\,\pi(n)^{\frac{c_s^2+1}{c_s^2+c_a^2}}}{\sum_{n=0}^{\infty}\pi(n)^{\frac{c_s^2+1}{c_s^2+c_a^2}}} + \left(\frac{c_s^2+1}{2}\right)\frac{\sum_{n=0}^{\infty}(n-L)^{+}\,\pi(n)^{\frac{c_s^2+1}{c_s^2+c_a^2}}}{\sum_{n=0}^{\infty}\pi(n)^{\frac{c_s^2+1}{c_s^2+c_a^2}}}$$

where $\pi(n)$ is probability mass function for the original system under M/M/ input



2. Diffusion control problem for dynamic policies

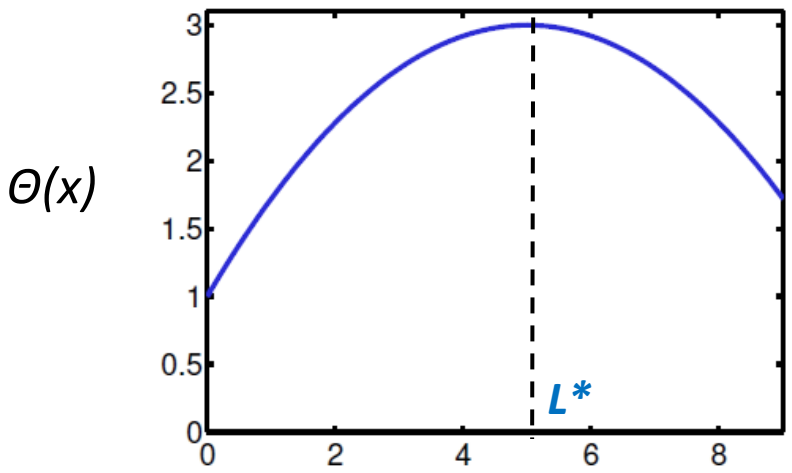
Dynamic policies

Static policies too sensitive to λ

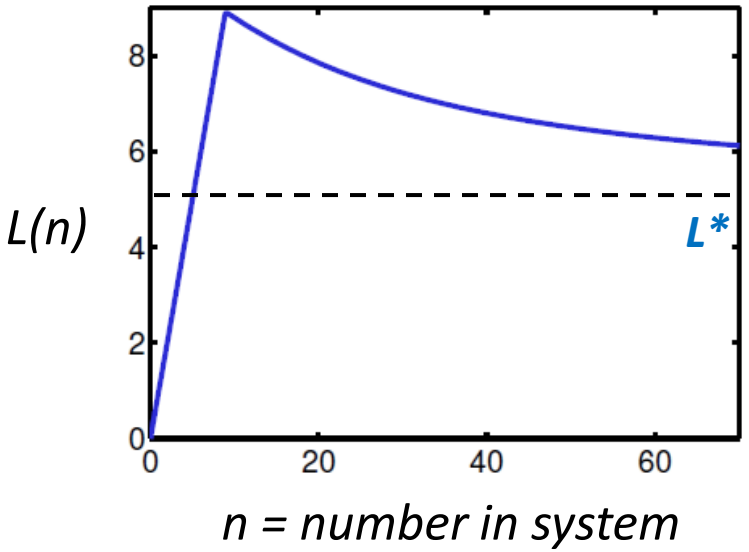
IDEA: Dynamically adjust L based on congestion

EXAMPLE:

Drift function



Policy ($c_s^2 = c_a^2 = 10$)



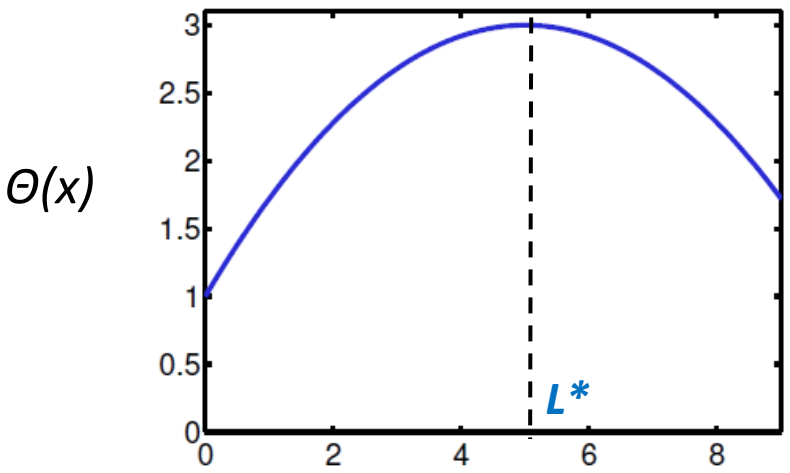
Dynamic policies

Static policies too sensitive to λ

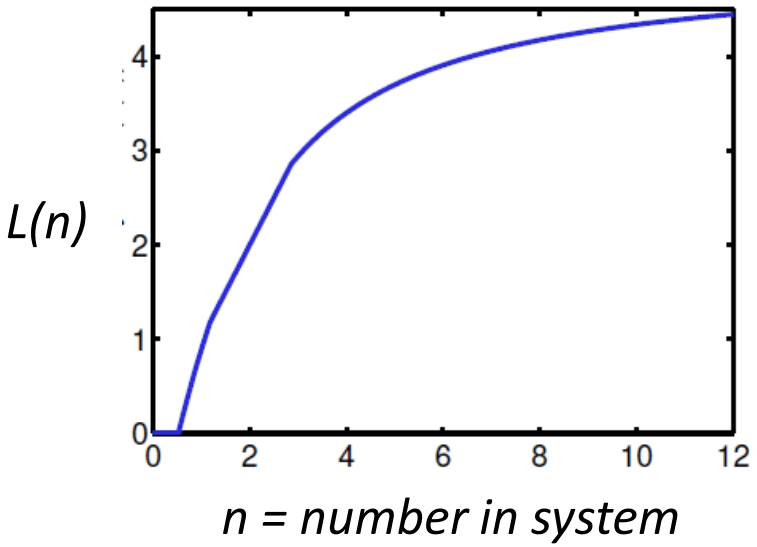
IDEA: Dynamically adjust *L* based on congestion

EXAMPLE:

Drift function

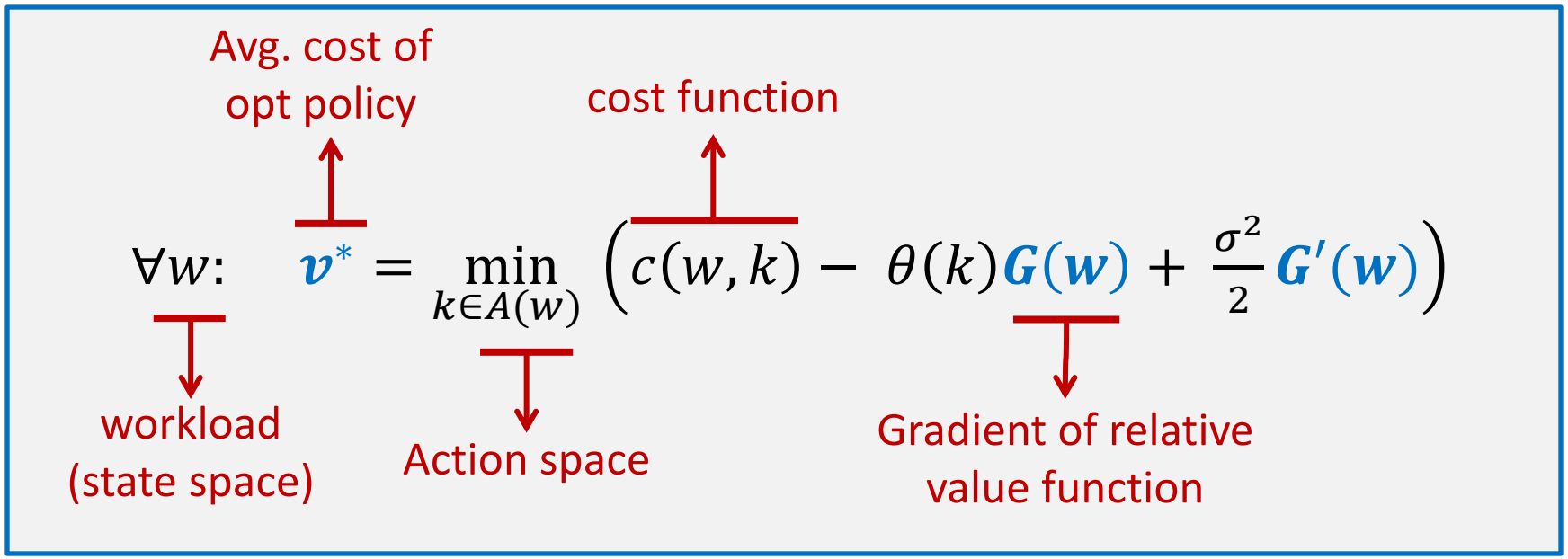


Policy ($c_s^2 = c_a^2 = 0.3$)



Dynamic policies

CRUX: an average cost diffusion control problem



- Arbitrary $\theta(k) \Rightarrow$ Must resort to numerical methods
- State of the art: discretize into a locally consistent MDP (see Kushner, Dupuis)

Average cost diffusion control problem

$$\forall w: \quad \boldsymbol{v}^* = \min_{k \in A(w)} \left(c(w,k) - \theta(k)\boldsymbol{G}(\boldsymbol{w}) + \frac{\sigma^2}{2}\boldsymbol{G}'(\boldsymbol{w}) \right)$$

Would be done if knew ***G(0)*** and $\boldsymbol{v}^*$

FACT: *G(0)* = 0 since the diffusion reflects at *w* = 0

What about $\boldsymbol{v}^*$?

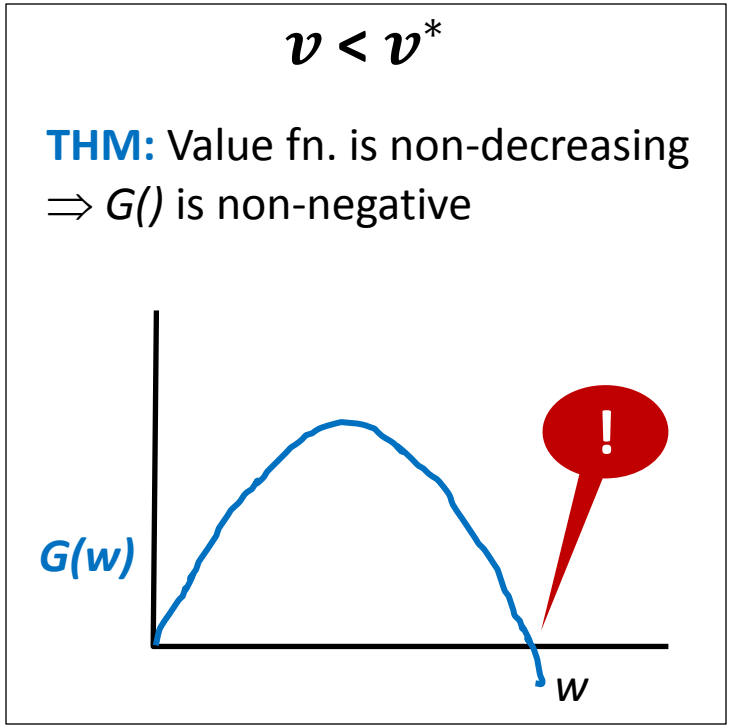
AVG. COST ITERATION ALGORITHM:

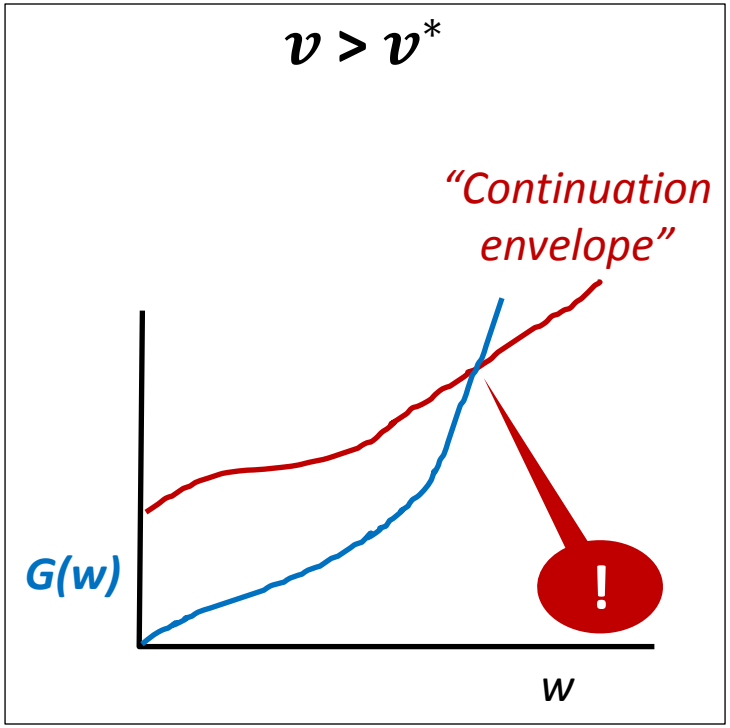
1. Guess $\boldsymbol{v}$
2. Check if $\boldsymbol{v} < \boldsymbol{v}^*$, or $\boldsymbol{v} > \boldsymbol{v}^*$
3. Refine guess and iterate until ε-optimal

$$\forall w: \quad \boldsymbol{v}^* = \min_{k \in A(w)} \left(c(w,k) - \theta(k)\boldsymbol{G}(\boldsymbol{w}) + \frac{\sigma^2}{2}\boldsymbol{G}'(\boldsymbol{w}) \right)$$

AVG. COST ITERATION ALGORITHM:

1. Guess $\boldsymbol{v}$
2. Check if $\boldsymbol{v} < \boldsymbol{v}^*$, or $\boldsymbol{v} > \boldsymbol{v}^*$
3. Refine guess and iterate until ε-optimal





$$\forall w: \quad \boldsymbol{v}^* = \min_{k \in A(w)} \left(c(w,k) - \theta(k)\boldsymbol{G}(\boldsymbol{w}) + \frac{\sigma^2}{2}\boldsymbol{G}'(\boldsymbol{w}) \right)$$

ALGORITHM 1:

1. Guess $\boldsymbol{v}$

2. Check if $\boldsymbol{v} < \boldsymbol{v}^*$, or $\boldsymbol{v} > \boldsymbol{v}^*$

   - Test events occur for $w = O\left(\log \frac{1}{|v - v^*|}\right)$

3. Refine guess and iterate until ε-optimal

   - $O\left(\log \frac{1}{\varepsilon}\right)$ iterations using binary search

ALGORITHM 2: More sophisticated; based on Newton-Raphson root finding method – needs $O\left(\log \log \frac{1}{\varepsilon}\right)$ iterations

SUMMARY

Two general concepts for control of systems with state-dependent parameters

1. An axiomatic approach to asymptotic scaling
   - Fix the limit under a tractable arrival process,
   - and reverse engineer the sequence to guarantee the limit

2. A numerical tool
   - Average cost iteration algorithm for 1-dimensional diffusions with one known reflection