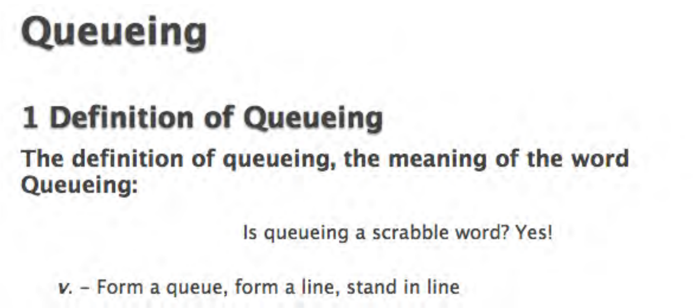
Tutorial

Bandit Processes and Index Policies

Richard Weber, University of Cambridge

Young European Queueing Theorists (YEQT VII) workshop on
Scheduling and priorities in queueing systems,
Eindhoven, November 4–5–6, 2013



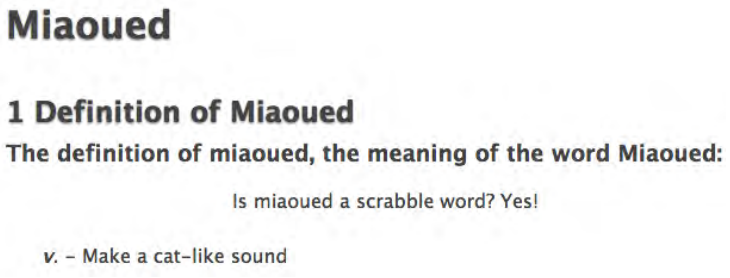
Queueing

1 Definition of Queueing

The definition of queueing, the meaning of the word Queueing:

Is queueing a scrabble word? Yes!

v. – Form a queue, form a line, stand in line



Miaoued

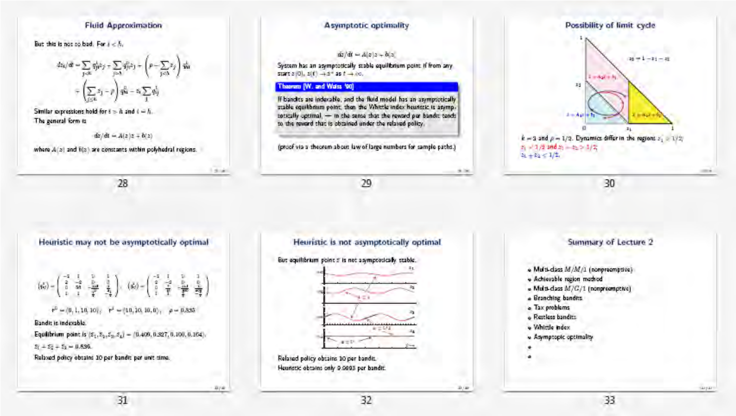
1 Definition of Miaoued

The definition of miaoued, the meaning of the word Miaoued:

Is miaoued a scrabble word? Yes!

v. - Make a cat-like sound

Tutorial slides



Fluid Approximation

But this is not so bad. For $i < \bar{k}$,

$$dz_i/dt = \sum_{j<\bar{k}} q^1_{ji} z_j + \sum_{j>\bar{k}} q^0_{ji} z_j + \left(\rho - \sum_{j<\bar{k}} z_j\right) q^1_{\bar{k}i} + \left(\sum_{j\le\bar{k}} z_j - \rho\right) q^0_{\bar{k}i} - z_i \sum_j q^1_{ij}$$

Similar expressions hold for $i > \bar{k}$ and $i = \bar{k}$.
The general form is

$$dz/dt = A(z)z + b(z)$$

where $A(z)$ and $b(z)$ are constants within polyhedral regions.

28

Asymptotic optimality

$$dz/dt = A(z)z + b(z)$$

System has an asymptotically stable equilibrium point if from any start $z(0)$, $z(t) \to z^*$ as $t \to \infty$.

Theorem [W. and Weiss '90]

If bandits are indexable, and the fluid model has an asymptotically stable equilibrium point, then the Whittle index heuristic is asymptotically optimal, — in the sense that the reward per bandit tends to the reward that is obtained under the relaxed policy.

(proof via a theorem about law of large numbers for sample paths.)

29

Possibility of limit cycle

$\bar{k} = 3$ and $\rho = 1/2$. Dynamics differ in the regions $z_1 \ge 1/2$;
$z_1 < 1/2$ and $z_1 + z_2 > 1/2$;
$z_1 + z_2 \le 1/2$.

30

Heuristic may not be asymptotically optimal

$$(q^0_{ij}) = \begin{pmatrix} -2 & 1 & 0 & 1 \\ 2 & -2 & 0 & 0 \\ 0 & 56 & -\frac{113}{2} & \frac{1}{2} \\ 1 & 1 & \frac{1}{2} & -\frac{5}{2} \end{pmatrix}, \quad (q^1_{ij}) = \begin{pmatrix} -2 & 1 & 0 & 1 \\ 2 & -2 & 0 & 0 \\ 0 & \frac{7}{25} & -\frac{113}{400} & \frac{1}{400} \\ 1 & 1 & \frac{1}{2} & -\frac{5}{2} \end{pmatrix}$$

$$r^0 = (0, 1, 10, 10), \quad r^1 = (10, 10, 10, 0), \quad \rho = 0.835$$

Bandit is indexable.

Equilibrium point is $(\bar{z}_1, \bar{z}_2, \bar{z}_3, \bar{z}_4) = (0.409, 0.327, 0.100, 0.164)$.

$\bar{z}_1 + \bar{z}_2 + \bar{z}_3 = 0.836$.

Relaxed policy obtains 10 per bandit per unit time.

31

Heuristic is not asymptotically optimal

But equilibrium point $\bar{z}$ is not asymptotically stable.

Relaxed policy obtains 10 per bandit.
Heuristic obtains only 9.9993 per bandit.

32

Summary of Lecture 2

- Multi-class $M/M/1$ (nonpreemptive)
- Achievable region method
- Multi-class $M/G/1$ (nonpreemptive)
- Branching bandits
- Tax problems
- Restless bandits
- Whittle index
- Asymptopic optimality
-
-

33

http://www.statslab.cam.ac.uk/~rrw1/talks/yetq.pdf

Abstract

We consider the problem of optimal control for a number of alternative Markov decision processes, where at each moment in time exactly one process must be “continued” while all other processes are “frozen”. Only the process that is continued produces any reward and changes its state. The aim is to maximize expected total discounted reward. A familiar example would be the problem of optimally scheduling the processing of jobs that reside in a single-server queue. A each moment one of the jobs is processed by the server, while all the other jobs are made to wait. Our aim is to minimize the expected total holding cost incurred until all jobs are complete. Another example is that of hunting for an apartment; we must choose the order in which to view apartments and decide when to stop viewing and rent the best apartment of those that have been viewed. This type of problem has a beautiful and surprising solution in terms of Gittins indices. In this tutorial I will review the theory of bandit processes and Gittins indices, describe some applications in scheduling and queueing, and tell you about some frontiers of research in the field.

Reading list

D. Bertsimas and J. Niño-Mora, Conservation laws, extended polymatroids and multiarmed bandit problems; a polyhedral approach to indexable systems, Math. Operat Res., 21:257–306, 1996.

J. C. Gittins, K. D. Glazebrook, and R. R. Weber. Multiarmed Bandit Allocation Indices, (2nd editon), Wiley, 2011.

K. D. Glazebrook, D. J. Hodge, C. Kirkbride, R. J. Minty, Stochastic scheduling: A short history of index policies and new approaches to index generation for dynamic resource allocation, J Scheduling., 2013.

K. Liu, R. R. Weber and Q. Zhao. Indexability and Whittle Index for restless bandit problems involving reset processes, Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on, 12-15 December, 7690–7696, 2011,

R. R. Weber and G. Weiss. On an index policy for restless bandits. J Appl Probab, 27:637–648, 1990.

P. Whittle. Restless bandits: activity allocation in a changing world. In A Celebration of Applied Probability, ed. J. Gani, J Appl Probab, 25A:287–298, 1998.

P. Whittle. Risk-sensitivity, large deviations and stochastic control, Eur J Oper Res, 73:295-303, 1994

Roadmap

Warm up

- Single machine scheduling
- Interchange arguments.
- Index policies.
- Pandora’s problem
- Scheduling $M/M/1$ queue.

Bandit processess

- Bandit processes.
- Gittins index theorem.
- Playing golf with more than one ball.
- Pandora plays golf.

Multi-class queues

- $M/M/1$.
- Achievable region method.
- Tax problems.
- Branching bandits.
- $M/G/1$.

Restless bandits

- Restless bandits.
- Whittle index.
- Asymptotic optimality.
- Risk-sensitive indices.

Warm up

Single machine scheduling

- N jobs are to be processed successively on one machine.

In what order should we process them?

Single machine scheduling

- N jobs are to be processed successively on one machine.

In what order should we process them?

- Job i has a known processing times t_i, a positive integer.

Single machine scheduling

- N jobs are to be processed successively on one machine.

In what order should we process them?

- Job i has a known processing times t_i, a positive integer.
- On completion of job i a reward r_i is obtained.

Single machine scheduling

- N jobs are to be processed successively on one machine.

In what order should we process them?

- Job i has a known processing times t_i, a positive integer.
- On completion of job i a reward r_i is obtained.
- If processed in the order $1, 2, \ldots, N$, total **discounted** reward is

$$r_1\beta^{t_1} + r_2\beta^{t_1+t_2} + \cdots + r_N\beta^{t_1+\cdots+t_N}$$

where $0 < \beta < 1$.

Dynamic programming solution

Let $S_k \subset \{1, \ldots, N\}$ be a set of k uncompleted jobs.

The dynamic programming equation is

$$F(S_k) = \max_{i \in S_k} \Big[\beta^{t_i} r_i + \beta^{t_i} F(S_k - \{i\}) \Big],$$

with $F_0(\varnothing) = 0$.

In principle we can solve the scheduing problem with dynamic programming.

But of course there is an easier way ...

Interchange arguments

Consider processing jobs in the order:

$$i_1, \ldots, i_k, i, j, i_{k+3}, \ldots, i_N.$$

Interchange arguments

Consider processing jobs in the order:

$$i_1, \ldots, i_k, i, j, i_{k+3}, \ldots, i_N.$$

Consider interchanging the order of jobs i and j:

$$i_1, \ldots, i_k, j, i, i_{k+3}, \ldots, i_N.$$

Interchange arguments

Consider processing jobs in the order:

$$i_1, \dots, i_k, i, j, i_{k+3}, \dots, i_N.$$

Consider interchanging the order of jobs i and j:

$$i_1, \dots, i_k, j, i, i_{k+3}, \dots, i_N.$$

Rewards under the two schedules are respectively

$$R_1 + \beta^{T+t_i} r_i + \beta^{T+t_i+t_j} r_j + R_2$$
$$R_1 + \beta^{T+t_j} r_j + \beta^{T+t_j+t_i} r_i + R_2.$$

$T = t_{i_1} + \cdots + t_{i_k}$, and

R_1 and R_2 are rewards accruing from jobs coming before and after i, j; (which is the same in both schedules).

Simple algebra $\implies$ Reward of the first schedule is greater if

$$r_i\beta^{t_i}/(1-\beta^{t_i}) > r_j\beta^{t_j}/(1-\beta^{t_j}).$$

Hence a schedule can be optimal only if the jobs are taken in decreasing order of the **indices** $r_i\beta^{t_i}/(1-\beta^{t_i})$.

Theorem

Total discounted reward is maximized by the **index policy** of always processing the uncompleted job of greatest index, computed as

$$G_i = \frac{r_i\beta^{t_i}(1-\beta)}{(1-\beta^{t_i})}.$$

Simple algebra $\implies$ Reward of the first schedule is greater if

$$r_i\beta^{t_i}/(1-\beta^{t_i}) > r_j\beta^{t_j}/(1-\beta^{t_j}).$$

Hence a schedule can be optimal only if the jobs are taken in decreasing order of the **indices** $r_i\beta^{t_i}/(1-\beta^{t_i})$.

Theorem

Total discounted reward is maximized by the **index policy** of always processing the uncompleted job of greatest index, computed as

$$G_i = \frac{r_i\beta^{t_i}(1-\beta)}{(1-\beta^{t_i})}.$$

Notice that $\dfrac{1-\beta^{t_i}}{1-\beta} = 1+\beta+\cdots+\beta^{t_i-1}$,
so $G_i \to r_i/t_i$ as $\beta \to 1$.

Weitzman's Pandora problem

Weitzman, M. L. (1979). Optimal search for the best alternative. *Econometrica*, 47:641–654.

Martin L. Weitzman is Professor of Economics at Harvard University.

Weitzman's Pandora problem

- Pandora has n boxes.

Weitzman's Pandora problem

- Pandora has n boxes.
- Box i contains a prize, of value x_i, distributed with known c.d.f. F_i.

Weitzman's Pandora problem

- Pandora has n boxes.
- Box i contains a prize, of value x_i, distributed with known c.d.f. F_i.
- At known cost c_i she can open box i and discover x_i.

Weitzman's Pandora problem

- Pandora has n boxes.
- Box i contains a prize, of value x_i, distributed with known c.d.f. F_i.
- At known cost c_i she can open box i and discover x_i.
- Pandora may open boxes in any order, and stop at will.

Weitzman's Pandora problem

- Pandora has n boxes.
- Box i contains a prize, of value x_i, distributed with known c.d.f. F_i.
- At known cost c_i she can open box i and discover x_i.
- Pandora may open boxes in any order, and stop at will..
- She opens a subset of boxes $S \subseteq \{1, \dots, n\}$ and then stops. She wishes to maximize the expected value of

$$R = \max_{i \in S} x_i - \sum_{i \in S} c_i.$$

Reasons for liking Weitzman's problem

Weitzmans' problem is attractive.

1. It has many applications:
   - hunting for a house
   - selling a house (accepting the best offer)
   - searching for a job,
   - looking for research project to focus upon.

Reasons for liking Weitzman's problem

Weitzmans' problem is attractive.

1. It has many applications:
   - hunting for a house
   - selling a house (accepting the best offer)
   - searching for a job,
   - looking for research project to focus upon.
2. It has an index policy solution, a so-called **Pandora rule**

   - Calculate a 'reservation prize' value for each box.

   - Open boxes in descending order of reservation prizes until a prize is found whose value exceeds the reservation prize of any unopened box.

Index policy for Pandora's problem

We seek to maximize expected value of

$$R = \max_{i \in S} x_i - \sum_{i \in S} c_i.$$

where S is the set of opened boxes.

The reservation value (index) of box i is

$$x_i^* = \inf \Big\{ y : y \geq -c_i + E[\max(x_i, y)] \Big\}.$$

Index policy for Pandora's problem

We seek to maximize expected value of

$$R = \max_{i \in S} x_i - \sum_{i \in S} c_i.$$

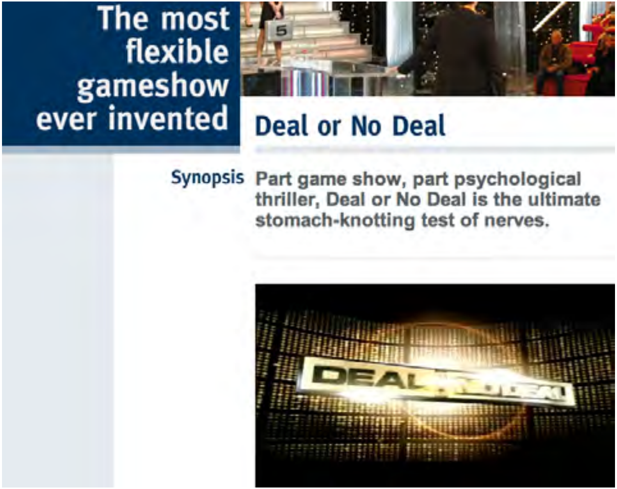
where S is the set of opened boxes.

The reservation value (index) of box i is

$$x_i^* = \inf \Big\{ y : y \geq -c_i + E[\max(x_i, y)] \Big\}.$$

Weitzman's Pandora rule. *Open the unopened box with greatest reservation value, until all reservations values are less than the greatest prize that has been found.*

Deal or no deal



The most flexible gameshow ever invented

Deal or No Deal

Synopsis Part game show, part psychological thriller, Deal or No Deal is the ultimate stomach-knotting test of nerves.

Scheduling in a two-class $M/M/1$ queue

- Jobs of two types arrive at a single server according to independent Poisson streams, with rates λ_i, $i = 1, 2$.
- Service times of type (class) i jobs are
  - exponentially distributed with mean μ_i^{-1}, $i = 1, 2$.
  - independent of each other and of the arrival streams.
- Assume overall rate at which work arrives at the system, namely $\lambda_1/\mu_1 + \lambda_2/\mu_2$, is less than 1.

Our goal: minimize holding cost rate

We wish to choose policy π

— which is to be past-measurable [non-anticipative], non-idling and preemptive —,

Our goal: minimize holding cost rate

We wish to choose policy π

— which is to be past-measurable [non-anticipative], non-idling and preemptive —,

to minimize the long-term holding cost rate, i.e.

$$\underset{\pi}{\text{minimize}}\ \Big\{c_1 E_\pi(N_1) + c_2 E_\pi(N_2)\Big\}.$$

c_i is a positive-valued class i holding cost rate.

N_i is the number of class i jobs in the system.

E_π is expectation under steady state distribution induced by π.

A conservation law

Work-in-system process, $W^\pi(t)$, is the aggregate of the remaining service times of all jobs in the system at t.

Pollaczek–Khinchine formula for the $M/G/1$ queue states that if X has the distribution of a service time then the mean work in system is $E_\pi[W] = \frac{1}{2}\lambda EX^2/(1-\lambda EX)$.

Since π is non-idling, $E_\pi[W]$ is independent of π, and so

$$E_\pi[W] = \frac{E_\pi(N_1)}{\mu_1} + \frac{E_\pi(N_2)}{\mu_2} = \frac{\rho_1\mu_1^{-1} + \rho_2\mu_2^{-1}}{1-\rho_1-\rho_2},$$

$\rho_i = \lambda_i/\mu_i$ is the rate at which class i work enters the system.

Some constraints

Let $x_i^\pi = E_\pi(N_i)/\mu_i$ be the mean work-in-system of class i.

Some constraints

Let $x_i^\pi = E_\pi(N_i)/\mu_i$ be the mean work-in-system of class i.

The work-in-system of type 1 jobs is no less than it would be if we always gave them priority. Hence

$$x_1^\pi = \frac{E_\pi(N_1)}{\mu_1} \geq \frac{\rho_1 \mu_1^{-1}}{1-\rho_1}.$$

with equality under the policy (denoted $1 \to 2$) that always gives priority to type 1 jobs.

Some constraints

Let $x_i^\pi = E_\pi(N_i)/\mu_i$ be the mean work-in-system of class i.

The work-in-system of type 1 jobs is no less than it would be if we always gave them priority. Hence

$$x_1^\pi = \frac{E_\pi(N_1)}{\mu_1} \geq \frac{\rho_1 \mu_1^{-1}}{1-\rho_1}.$$

with equality under the policy (denoted $1 \to 2$) that always gives priority to type 1 jobs.

Similiarly,

$$x_2^\pi = \frac{E_\pi(N_2)}{\mu_2} \geq \frac{\rho_2 \mu_2^{-1}}{1-\rho_2}.$$

under $2 \to 1$.

A mathematical program

Consider the optimization problem

$$\begin{aligned}\underset{\pi}{\text{minimize}}\left\{c_1 E_\pi(N_1) + c_2 E_\pi(N_2)\right\} &= \underset{\pi}{\text{minimize}}\left\{c_1\mu_1 x_1^\pi + c_2\mu_2 x_2^\pi\right\}\\ &= \underset{(x_1,x_2)\in X}{\text{minimize}}\left\{c_1\mu_1 x_1 + c_2\mu_2 x_2\right\},\end{aligned}$$

where X is the set of (x_1^π, x_2^π) that are **achievable** for some π.

A mathematical program

Consider the optimization problem

$$\underset{\pi}{\text{minimize}}\left\{c_1E_\pi(N_1)+c_2E_\pi(N_2)\right\}=\underset{\pi}{\text{minimize}}\left\{c_1\mu_1x_1^\pi+c_2\mu_2x_2^\pi\right\}$$
$$=\underset{(x_1,x_2)\in X}{\text{minimize}}\left\{c_1\mu_1x_1+c_2\mu_2x_2\right\},$$

where X is the set of (x_1^π, x_2^π) that are **achievable** for some π.

Have argued that

$$X\subseteq P=\left\{(x_1,x_2)\ :x_1\geq\frac{\rho_1\mu_1^{-1}}{1-\rho_1},\ x_2\geq\frac{\rho_2\mu_2^{-1}}{1-\rho_2},\right.$$
$$\left.x_1+x_2=\frac{\rho_1\mu_1^{-1}+\rho_2\mu_2^{-1}}{1-\rho_1-\rho_2}\right\}.$$

A linear program relaxation

LP relaxation of our optimization problem is

$$\underset{(x_1,x_2)\in P}{\text{minimize}} \left\{ c_1\mu_1 x_1 + c_2\mu_2 x_2 \right\}$$

$$P = \left\{ (x_1, x_2) \ : x_1 \geq \frac{\rho_1 \mu_1^{-1}}{1-\rho_1}, \ x_2 \geq \frac{\rho_2 \mu_2^{-1}}{1-\rho_2}, \right. \\ \left. x_1 + x_1 = \frac{\rho_1\mu_1^{-1} + \rho_2\mu_2^{-1}}{1-\rho_1-\rho_2} \right\}.$$

Easy to solve!

If $c_1\mu_1 > c_2\mu_2$ then the optimal solution is at x^* where

$$x_1^* = \frac{\rho_1\mu_1^{-1}}{1-\rho_1}, \quad x_2^* = \frac{\rho_1\mu_1^{-1} + \rho_2\mu_2^{-1}}{1-\rho_1-\rho_2} - \frac{\rho_1\mu_1^{-1}}{1-\rho_1}.$$

Furthermore $x^* \in X$ and achieved by the policy $1 \to 2$.

Lessons

- Long term holding cost rate is maximized by giving priority to the job class of greatest $c_i\mu_i$.

- The policy $1 \rightarrow 2$ is an **index policy**.

  At all times the server devotes its effort to the job of greatest index, i.e. greatest $c_i\mu_i$.

- The obvious generalizaton of this result is true if there were more than 2 job types.

Given the hint of this simple example, how would you prove it?

Proof strategy

- Find a set of linear inequalities that must be satisfied. These define a polytope P, such that achievable x must lie in P.

  There are constraints like

$$x_i \geq \frac{\rho_i \mu_1^{-1}}{1-\rho_1}, \quad x_1 + x_2 \geq \frac{\rho_1 \mu_1^{-1} + \rho_2 \mu_2^{-1}}{1-\rho_1-\rho_2}$$

$$\sum_{i \in S} x_i \geq \frac{\sum_{i \in S} \rho_i \mu_1^{-1}}{1 - \sum_{i \in S} \rho_i}, \quad \text{for all } S \subseteq \{1, \ldots, m\}.$$

Proof strategy

- Find a set of linear inequalities that must be satisfied. These define a polytope P, such that achievable x must lie in P.

  There are constraints like

$$x_i \geq \frac{\rho_i \mu_1^{-1}}{1-\rho_1}, \quad x_1 + x_2 \geq \frac{\rho_1 \mu_1^{-1} + \rho_2 \mu_2^{-1}}{1-\rho_1-\rho_2}$$

$$\sum_{i \in S} x_i \geq \frac{\sum_{i \in S} \rho_i \mu_1^{-1}}{1-\sum_{i \in S} \rho_i}, \quad \text{for all } S \subseteq \{1, \ldots, m\}.$$

- Minimize $\sum_i c_i \mu_i x_i$ over $x \in P$.

Proof strategy

- Find a set of linear inequalities that must be satisfied. These define a polytope P, such that achievable x must lie in P.

  There are constraints like

$$x_i \geq \frac{\rho_i \mu_1^{-1}}{1-\rho_1}, \quad x_1 + x_2 \geq \frac{\rho_1 \mu_1^{-1} + \rho_2 \mu_2^{-1}}{1-\rho_1-\rho_2}$$

$$\sum_{i \in S} x_i \geq \frac{\sum_{i \in S} \rho_i \mu_1^{-1}}{1 - \sum_{i \in S} \rho_i}, \quad \text{for all } S \subseteq \{1, \ldots, m\}.$$

- Minimize $\sum_i c_i \mu_i x_i$ over $x \in P$.
- Show optimal solution, x^*, is in X and is achieved by index policy that prioritizes jobs in decreasing order of $c_i \mu_i$.

Bandit processes

Two-job scheduling problem

$(r_1, t_1) = (9, 5)$, $(r_2, t_2) = (6, 2)$.

0, 0, 0, 0, 9, 0, ...

0, 6, 0, 0, 0, 0, ...

Two-job scheduling problem

$(r_1, t_1) = (9, 5)$, $(r_2, t_2) = (6, 2)$.

0, 0, 0, 0, 9, 0, ...

, 6, 0, 0, 0, 0, ... $\longrightarrow$ 0

Two-job scheduling problem

$(r_1, t_1) = (9, 5)$, $(r_2, t_2) = (6, 2)$.

0, 0, 0, 0, 9, 0, ...

, , 0, 0, 0, 0, ...

⟶ 0, 6

Two-job scheduling problem

$(r_1, t_1) = (9, 5)$, $(r_2, t_2) = (6, 2)$.

, 0, 0, 0, 9, 0, ...

,   , 0, 0, 0, 0, ...

$\longrightarrow$ 0, 6, 0

Two-job scheduling problem

$(r_1, t_1) = (9, 5)$, $(r_2, t_2) = (6, 2)$.

, , 0, 0, 9, 0, ...

, , 0, 0, 0, 0, ...

⟶ 0, 6, 0, 0

Two-job scheduling problem

$(r_1, t_1) = (9, 5)$, $(r_2, t_2) = (6, 2)$.

, , , 0, 9, 0, ...

, , 0, 0, 0, 0, ...

$\longrightarrow$ 0, 6, 0, 0, 0

Two-job scheduling problem

$(r_1, t_1) = (9, 5)$, $(r_2, t_2) = (6, 2)$.

, , , , 9, 0, ...

, , 0, 0, 0, 0, ...

$\longrightarrow$ 0, 6, 0, 0, 0, 0

Two-job scheduling problem

$(r_1, t_1) = (9, 5)$, $(r_2, t_2) = (6, 2)$.

, , , , , 0, ...

, , , , 0, 0, ...

$\longrightarrow$ 0, 6, 0, 0, 0, 0, 9,

Reward = $0 + 6\beta + 0\beta^2 + 0\beta^3 + 0\beta^4 + 0\beta^5 + 0\beta^6 + 9\beta^7$

$0 < \beta < 1$.

Two-armed bandit

3, 10, 4, 9, 12, 1, ...

5, 6, 2, 15, 2, 7, ...

Two-armed bandit

3, 10, 4, 9, 12, 1, ...

, 6, 2, 15, 2, 7, ...

→ 5

Two-armed bandit

3, 10, 4, 9, 12, 1, ...

,   , 2, 15, 2, 7, ...   ⟶ 5, 6

Two-armed bandit

, 10, 4, 9, 12, 1, ...

,   , 2, 15, 2, 7, ...

⟶ 5, 6, 3

Two-armed bandit

, , 4, 9, 12, 1, ...

, , 2, 15, 2, 7, ...

$\longrightarrow$ 5, 6, 3, 10,

Two-armed bandit

, , , 9, 12, 1, ...

, , 2, 15, 2, 7, ...

→ 5, 6, 3, 10, 4

Two-armed bandit

, , , , 12, 1, ...

, , 2, 15, 2, 7, ...

→ 5, 6, 3, 10, 4, 9

Two-armed bandit

, , , , , 1, ...

, , 2, 15, 2, 7, ...

→ 5, 6, 3, 10, 4, 9, 12

Two-armed bandit

, , , , , 1, ...

, , , 15, 2, 7, ...

⟶ 5, 6, 3, 10, 4, 9, 12, 2

Two-armed bandit

, , , , , 1, ...

, , , , 2, 7, ...

$\longrightarrow$ 5, 6, 3, 10, 4, 9, 12, 2, 15

Two-armed bandit

, , , , , 1, ...

, , , , 2, 7, ...

$\longrightarrow$ 5, 6, 3, 10, 4, 9, 12, 2, 15

Reward = $5 + 6\,\beta + 3\,\beta^2 + 10\,\beta^3 + \cdots$

$0 < \beta < 1.$

Two-armed bandit

, , , , , 1, ...

, , , , 2, 7, ...

$\longrightarrow$ 5, 6, 3, 10, 4, 9, 12, 2, 15

Reward = $5 + 6\,\beta + 3\,\beta^2 + 10\,\beta^3 + \cdots$

$0 < \beta < 1$. Of course, in practice we must choose which arms to pull without knowing the future sequences of rewards.

Each of the two arms is a **bandit process**.

Bandit processes

A bandit process is a special type of Markov Decision Process in which there are just two possible actions:

- $u = 1$ (**continue**)

  produces reward $r(x_t)$ and the state changes, to x_{t+1}, according to Markov dynamics $P_i(x_t, x_{t+1})$.

- $u = 0$ (**freeze**)

  produces no reward and the state does not change (hence the term ‘freeze’).

Bandit processes

A bandit process is a special type of Markov Decision Process in which there are just two possible actions:

- $u = 1$ (**continue**)

  produces reward $r(x_t)$ and the state changes, to x_{t+1}, according to Markov dynamics $P_i(x_t, x_{t+1})$.

- $u = 0$ (**freeze**)

  produces no reward and the state does not change (hence the term ‘freeze’).

A **simple family of alternative bandit processes** (SFABP) is a collection of N such bandit processes, in known states $x_1(t), \dots, x_N(t)$.

SFABP

At each time, $t \in \{0, 1, 2, \ldots\}$,

- One bandit process is to be activated (pulled/**continued**)

  If arm i activated then it changes state:

  $$x \to y \quad \text{with probability } P_i(x, y)$$

  and produces reward $r_i(x_i(t))$.

SFABP

At each time, $t \in \{0, 1, 2, \ldots\}$,

- One bandit process is to be activated (pulled/**continued**)

  If arm i activated then it changes state:

  $$x \to y \quad \text{with probability } P_i(x, y)$$

  and produces reward $r_i(x_i(t))$.

- All other bandit processes remain passive (not pulled/**frozen**).

SFABP

At each time, $t \in \{0, 1, 2, \ldots\}$,

- One bandit process is to be activated (pulled/**continued**)

  If arm i activated then it changes state:

  $$x \to y \quad \text{with probability } P_i(x, y)$$

  and produces reward $r_i(x_i(t))$.

- All other bandit processes remain passive (not pulled/**frozen**).

Objective: maximize the expected total β-discounted reward

$$E\left[\sum_{t=0}^{\infty} r_{i_t}(x_{i_t}(t))\,\beta^t\right],$$

where i_t is the arm pulled at time t, $(0 < \beta < 1)$.

Dynamic effort allocation

- **Job Scheduling**: in what order should I work on the tasks in my in-tray?

Dynamic effort allocation

- **Job Scheduling**: in what order should I work on the tasks in my in-tray?
- **Research projects**: how should I allocate my research time amongst my favorite open problems so as to maximize the value of my completed research?

Dynamic effort allocation

- **Job Scheduling**: in what order should I work on the tasks in my in-tray?
- **Research projects**: how should I allocate my research time amongst my favorite open problems so as to maximize the value of my completed research?
- **Searching for information**: shall I spend more time browsing the web, or go to the library, or ask a friend?

Dynamic effort allocation

- **Job Scheduling**: in what order should I work on the tasks in my in-tray?
- **Research projects**: how should I allocate my research time amongst my favorite open problems so as to maximize the value of my completed research?
- **Searching for information**: shall I spend more time browsing the web, or go to the library, or ask a friend?
- **Dating strategy**: should I contact a new prospect, or try another date with someone I have dated before?

Dynamic programming solution

The dynamic programming equation is

$$F(x_1, \ldots, x_N)$$

$$= \max_i \Big\{ r_i(x_i) + \beta \sum_y P_i(x_i, y) F(x_1, \ldots, x_{i-1}, y, x_{i+1}, \ldots, x_N) \Big\}$$

Dynamic programming solution

The dynamic programming equation is

$$F(x_1, \ldots, x_N)$$

$$= \max_i \Big\{ r_i(x_i) + \beta \sum_y P_i(x_i, y) F(x_1, \ldots, x_{i-1}, y, x_{i+1}, \ldots, x_N) \Big\}$$

If bandit i moves on a state space of size k_i, then $(x_1, \ldots, x_N)$ moves on a state space of size $\prod_i k_i$ (exponential in N).

Gittins Index solution

Theorem [Gittins, ‘74, ‘79, ‘89]

Expected discounted reward is maximized by always continuing the bandit having greatest value of ‘dynamic allocation index’

$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\Big[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \;\Big|\; x_i(0) = x_i\Big]}{E\Big[\sum_{t=0}^{\tau-1} \beta^t \;\Big|\; x_i(0) = x_i\Big]}$$

where τ is a (past-measurable) stopping-time.

Gittins Index solution

Theorem [Gittins, '74, '79, '89]

Expected discounted reward is maximized by always continuing the bandit having greatest value of 'dynamic allocation index'

$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\Big[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \;\Big|\; x_i(0) = x_i\Big]}{E\Big[\sum_{t=0}^{\tau-1} \beta^t \;\Big|\; x_i(0) = x_i\Big]}$$

where τ is a (past-measurable) stopping-time.

$G_i(x_i)$ is called the **Gittins index**.

Gittins Index solution

Theorem [Gittins, ‘74, ‘79, ‘89]

Expected discounted reward is maximized by always continuing the bandit having greatest value of ‘dynamic allocation index’

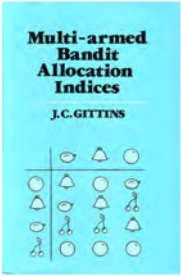
$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\left[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \,\middle|\, x_i(0) = x_i\right]}{E\left[\sum_{t=0}^{\tau-1} \beta^t \,\middle|\, x_i(0) = x_i\right]}$$

where τ is a (past-measurable) stopping-time.

$G_i(x_i)$ is called the **Gittins index**.

Gittins and Jones (1974). A dynamic allocation index for the sequential design of experiments. In Gani, J., editor, Progress in Statistics, pages 241–66. North-Holland, Amsterdam, NL. Read at the 1972 European Meeting of Statisticians, Budapest.

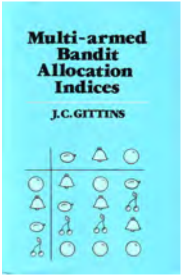
Multi-armed bandit allocation indices

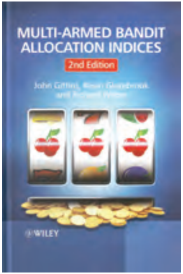


1st Edition edition, 1989, Gittins

Many applications to clinical trials, job scheduling, search, etc.

Multi-armed bandit allocation indices



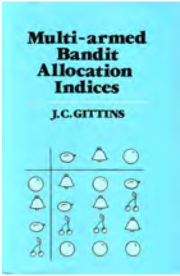

MULTI-ARMED BANDIT ALLOCATION INDICES
2nd Edition
John Gittins, Kevin Glazebrook and Richard Weber
WILEY

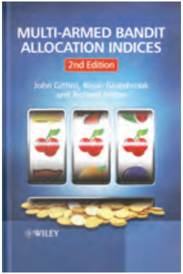
1st Edition edition, 1989, Gittins

Many applications to clinical trials, job scheduling, search, etc.

2nd Edition edition, 2011, Gittins, Glazebrook and Weber

Multi-armed bandit allocation indices





1st Edition edition, 1989, Gittins

Many applications to clinical trials, job scheduling, search, etc.

2nd Edition edition, 2011, Gittins, Glazebrook and Weber

Exploration versus exploitation

"Bandit problems embody in essential form a conflict evident in all human action: information versus immediate payoff."

— *Peter Whittle (1989)*

Gittins index

$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\left[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \;\middle|\; x_i(0) = x_i \right]}{E\left[\sum_{t=0}^{\tau-1} \beta^t \;\middle|\; x_i(0) = x_i \right]}$$

Gittins index

$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\Big[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \;\Big|\; x_i(0) = x_i\Big]}{E\Big[\sum_{t=0}^{\tau-1} \beta^t \;\Big|\; x_i(0) = x_i\Big]}$$

Discounted reward up to τ.

Gittins index

$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\left[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \;\middle|\; x_i(0) = x_i \right]}{E\left[\sum_{t=0}^{\tau-1} \beta^t \;\middle|\; x_i(0) = x_i \right]}$$

Discounted reward up to τ.

Discounted time up to τ.

Gittins index

$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\Big[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \;\Big|\; x_i(0) = x_i\Big]}{E\Big[\sum_{t=0}^{\tau-1} \beta^t \;\Big|\; x_i(0) = x_i\Big]}$$

Discounted reward up to τ.

Discounted time up to τ.

Note the role of the **stopping time** τ.
Stopping times are times recognisable when they occur.

Gittins index

$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\left[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \,\middle|\, x_i(0) = x_i \right]}{E\left[\sum_{t=0}^{\tau-1} \beta^t \,\middle|\, x_i(0) = x_i \right]}$$

Discounted reward up to τ.

Discounted time up to τ.

Note the role of the **stopping time** τ.
Stopping times are times recognisable when they occur.
How do you make perfect toast?

There is a rule for timing toast,
One never has to guess,
Just wait until it starts to smoke,
then 7 seconds less. (David Kendall)

Gittins index

$$G_i(x_i) = \sup_{\tau \geq 1} \frac{E\left[\sum_{t=0}^{\tau-1} r_i(x_i(t))\beta^t \;\middle|\; x_i(0) = x_i\right]}{E\left[\sum_{t=0}^{\tau-1} \beta^t \;\middle|\; x_i(0) = x_i\right]}$$

Discounted reward up to τ.

Discounted time up to τ.

In the scheduling problem $\tau = t_i$ and

$$G_i = \frac{r_i \beta^{t_i}}{1 + \beta + \cdots + \beta^{t_i - 1}} = (1 - \beta)\frac{r_i \beta^{t_i}}{1 - \beta^{t_i}}$$

Single machine scheduling

N jobs are to be processed successively on one machine.

Job i has a known processing times t_i, a positive integer.

On completion of job i a reward r_i is obtained.

Total discounted reward is maximized by the **index policy** which processes jobs in decreasing order of **indices**, G_i.

$$G_i = \frac{r_i \beta^{t_i}}{1 + \beta + \cdots + \beta^{t_i - 1}} = (1 - \beta) \frac{r_i \beta^{t_i}}{1 - \beta^{t_i}}$$

Gittins index via calibration

Can also define Gittins index by comparing to arm of fixed pay rate:

$$G_i(x_i) = \sup\Bigg\{\lambda : \\ \sum_{t=0}^{\infty}\beta^t\lambda \le \sup_{\tau\ge 1} E\left[\sum_{t=0}^{\tau-1}\beta^t r_i(x_i(t)) + \sum_{t=\tau}^{\infty}\beta^t\lambda \,\middle|\, x_i(0)=x_i\right]\Bigg\}.$$

Consider a problem with two bandit processes: the bandit process B_i and a **calibrating bandit process**, say Λ, which pays out a known reward λ at each step it is continued.

The Gittins index of B_i is the value of λ for which we are indifferent as to which of B_i and Λ to continue initially.

Gittins index theorem is surprising!

Peter Whittle tells the story:

“A colleague of high repute asked an equally well-known colleague:

— **What would you say if you were told that the multi-armed bandit problem had been solved?’**

Gittins index theorem is surprising!

Peter Whittle tells the story:

“A colleague of high repute asked an equally well-known colleague:

— **What would you say if you were told that the multi-armed bandit problem had been solved?’**

— **Sir, the multi-armed bandit problem is not of such a nature that it <u>can</u> be solved.’**

What has happened since 1989?

- Index theorem has become better known.
- Alternative proofs have been explored.

  Playing golf with N balls

  Achievable performance region approach
- Many applications (economics, engineering, . . .).
- Notions of indexation have been generalized.

  Restless Bandits

Proofs of the Index Theorem

Since Gittins (1974, 1979), many researchers have reproved, remodelled and resituated the index theorem.

Beale (1979)

Karatzas (1984)

Varaiya, Walrand, Buyukkoc (1985)

Chen, Katehakis (1986)

Kallenberg (1986)

Katehakis, Veinott (1986)

Eplett (1986)

Kertz (1986)

Tsitsiklis (1986)

Mandelbaum (1986, 1987)

Lai, Ying (1988)

Whittle (1988)

Weber (1992)

El Karoui, Karatzas (1993)

Ishikida and Varaiya (1994)

Tsitsiklis (1994)

Bertsimas, Niño-Mora (1996)

Glazebrook, Garbe (1996)

Kaspi, Mandelbaum (1998)

Bäuerle, Stidham (2001)

Dimitriu, Tetali, Winkler (2003)

Proofs of the Index Theorem

- Interchange arguments (but cunning ones!)
- Economic/gaming argument
- Linear programming relaxation (achievable region method)

Golf with N balls

Dumitriu, Tetali and Winkler, (2003). On playing golf with two balls.

N balls are strewn about a golf course at locations $x_1, \ldots, x_N$.

Golf with N balls

N balls are strewn about a golf course at locations $x_1, \dots, x_N$.

Hitting a ball i, that is in location x_i, costs $c(x_i)$,

$$x_i \to y \quad \text{with probability } P(x_i, y)$$

Ball goes in the hole with probability $P(x_i, 0)$.

Objective

Minimize the expected total cost incurred up to sinking a first ball.

Golf with N balls

N balls are strewn about a golf course at locations $x_1, \dots, x_N$.

Hitting a ball i, that is in location x_i, costs $c(x_i)$,

$$x_i \to y \quad \text{with probability } P(x_i, y)$$

Ball goes in the hole with probability $P(x_i, 0)$.

Objective

Minimize the expected total cost incurred up to sinking a first ball.

Answer

When ball i is in location x_i it has an index $\gamma_i(x_i)$.

Play the ball of smallest index, until a ball goes in the whole.

Gittins index theorem for golf with N balls

Golf with one ball

Consider golf with one ball, initially in location x_i.

Let's offer the golfer a prize λ, obtained when he sinks this ball in the hole (state 0).

Gittins index theorem for golf with N balls

Golf with one ball

Consider golf with one ball, initially in location x_i.

Let's offer the golfer a prize λ, obtained when he sinks this ball in the hole (state 0).

We might ask, *what is the least λ for which it is optimal for him to take at least one more stroke* — allowing him the option to retire at any point thereafter?

Gittins index theorem for golf

Golf with one ball

Consider golf with one ball, initially in location x_i.

Let's offer the golfer a prize λ, obtained when he sinks this ball in the hole (state 0).

We might ask, *what is the least λ for which it is optimal for him to take at least one more stroke* — allowing him the option to retire at any point thereafter?

$$\gamma_i(x_i) = \inf \left\{ \lambda : 0 \leq \sup_{\tau \geq 1} E \left[\lambda 1_{\{x_i(\tau)=0\}} - \sum_{t=0}^{\tau-1} c_i(x_i(t)) \,\middle|\, x_i(0) = x_i \right] \right\}.$$

Call $\gamma_i(x_i)$ the fair prize, (or Gittins index).

How to play golf
with one ball and an increasing fair prize

Having been offered a fair prize the golfer will play until the ball

- goes in the hole, or
- reaches a state $x_i(t)$ from which the offered prize is no longer great enough to tempt him to play further.

How to play golf with one ball and an increasing fair prize

Having been offered a fair prize the golfer will play until the ball

- goes in the hole, or
- reaches a state $x_i(t)$ from which the offered prize is no longer great enough to tempt him to play further.

  If the latter occurs, let us increase the prize to $\gamma_i(x_i(t))$.

It becomes the 'prevailing prize' at time t, i.e. $\max_{s\leq t}\gamma_i(x_i(s))$.

The prevailing prize is nondecreasing in t.

How to play golf with one ball and an increasing fair prize

Having been offered a fair prize the golfer will play until the ball

- goes in the hole, or
- reaches a state $x_i(t)$ from which the offered prize is no longer great enough to tempt him to play further.

  If the latter occurs, let us increase the prize to $\gamma_i(x_i(t))$.

It becomes the 'prevailing prize' at time t, i.e. $\max_{s \leq t} \gamma_i(x_i(s))$.

The prevailing prize is nondecreasing in t.

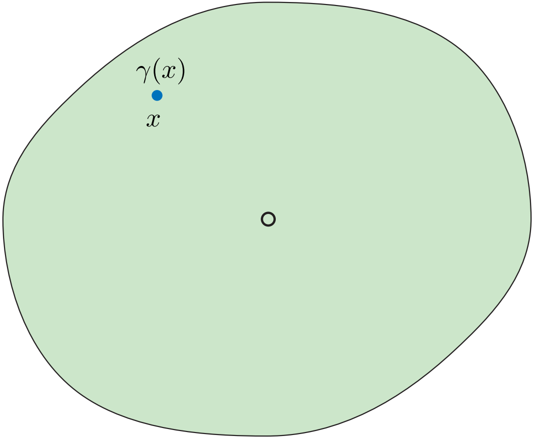
Now the golfer need never retire and can keep playing until the ball goes in the hole, say at time τ.

But his expected profit is just 0.

$$E\left[\gamma_i(x_i(\tau-1)) - \sum_{t=0}^{\tau-1} c_i(x_i(t) \,\middle|\, x_i(0) = x_i\right] = 0.$$

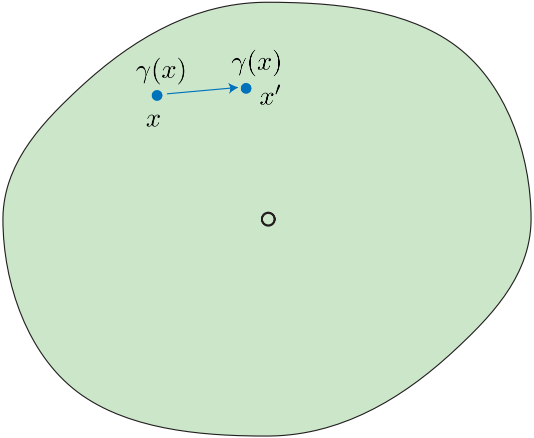
Golf with 1 ball

$\gamma(x) = 3.0$



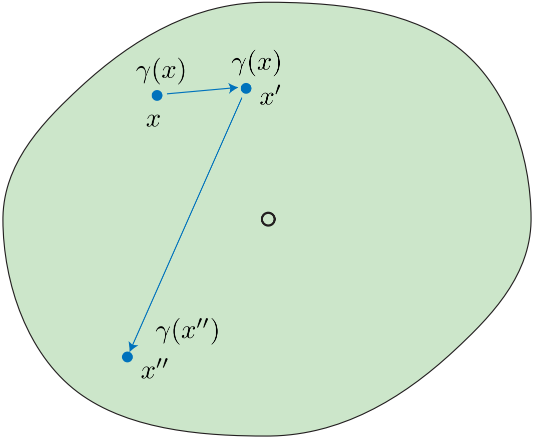
Golf with 1 ball

$\gamma(x) = 3.0$, $\gamma(x') = 2.5$



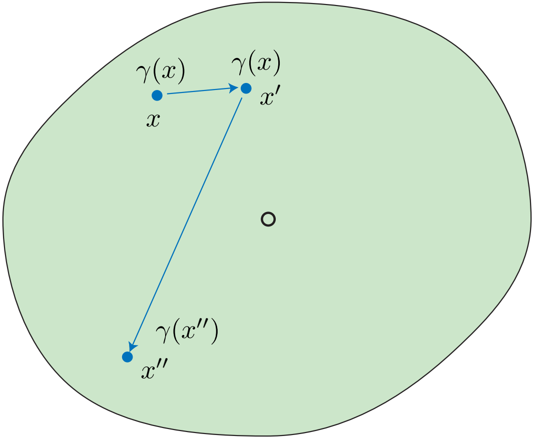
Golf with 1 ball

$\gamma(x) = 3.0$, $\gamma(x') = 2.5$, $\gamma(x'') = 4.0$



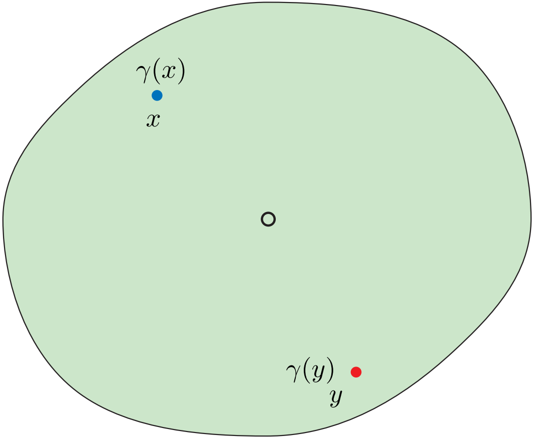
Golf with 1 ball

$\gamma(x) = 3.0$, $\gamma(x') = 2.5$, $\gamma(x'') = 4.0$
Prevailing prize sequence is 3.0, 3.0, 4.0, $\ldots$



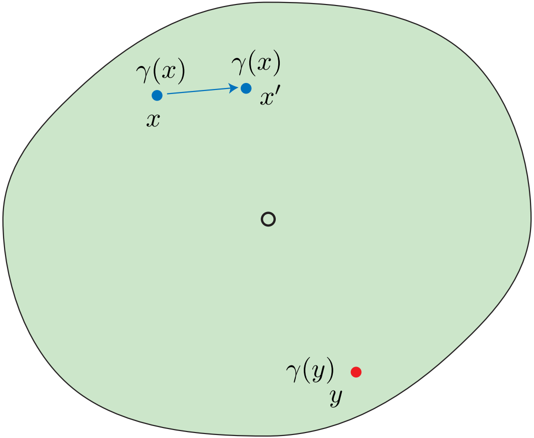
Golf with 2 balls

$\gamma(x) = \overline{3.0}$
$\gamma(y) = \overline{3.2}$



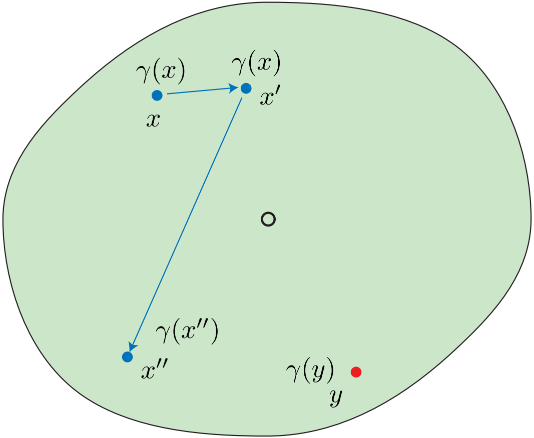
Golf with 2 balls

$\gamma(x) = \overline{3.0}$, $\gamma(x') = 2.5$
$\gamma(y) = \overline{3.2}$



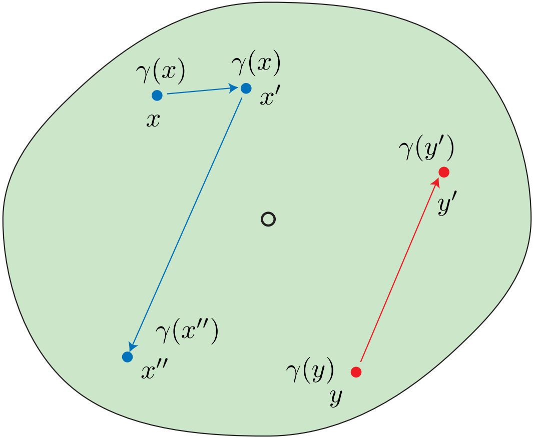
Golf with 2 balls

$\gamma(x) = 3.0$, $\gamma(x') = 2.5$, $\gamma(x'') = \overline{4.0}$
$\gamma(y) = \overline{3.2}$



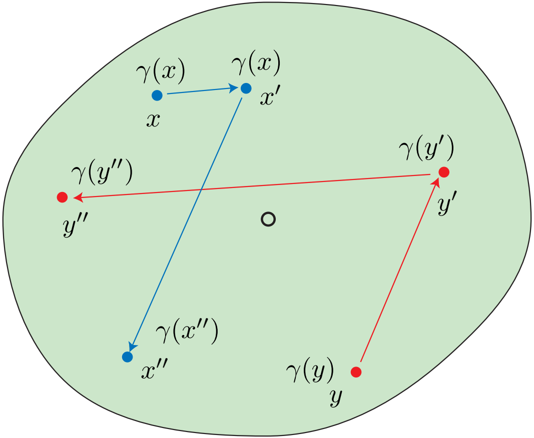
Golf with 2 balls

$\gamma(x) = 3.0$, $\gamma(x') = 2.5$, $\gamma(x'') = \overline{4.0}$
$\gamma(y) = 3.2$, $\gamma(y') = \overline{3.5}$



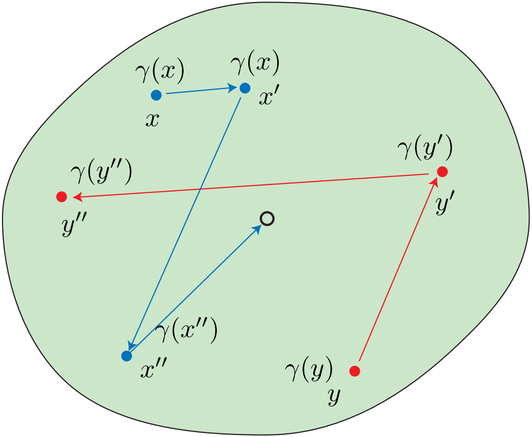
Golf with 2 balls

$\gamma(x) = 3.0$, $\gamma(x') = 2.5$, $\gamma(x'') = \overline{4.0}$
$\gamma(y) = 3.2$, $\gamma(y') = 3.5$, $\gamma(y'') = \overline{4.2}$



Golf with 2 balls

$\gamma(x) = 3.0$, $\gamma(x') = 2.5$, $\gamma(x'') = \overline{4.0}$
$\gamma(y) = 3.2$, $\gamma(y') = 3.5$, $\gamma(y'') = \overline{4.2}$



Index theorem for golf with N balls

Suppose the golfer keeps playing until a ball goes in the hole.

His prize is the prevailing prize of the ball he sinks.

Index theorem for golf with N balls

Suppose the golfer keeps playing until a ball goes in the hole.

His prize is the prevailing prize of the ball he sinks.

Prevailing prizes are defined in such a way that the golfer cannot make a strictly positive profit, and so for any policy σ,

$$E_\sigma(\text{cost incurred}) \geq E_\sigma(\text{prize eventually won}) \qquad (1)$$

Index theorem for golf with N balls

Suppose the golfer keeps playing until a ball goes in the hole.

His prize is the prevailing prize of the ball he sinks.

Prevailing prizes are defined in such a way that the golfer cannot make a strictly positive profit, and so for any policy σ,

$$E_\sigma(\text{cost incurred}) \geq E_\sigma(\text{prize eventually won}) \tag{1}$$

Let π be the policy: *always play the ball with least prevailing prize.*

Because each ball's sequence of prevailing prizes is nondecreasing.

$$E_\sigma(\text{prize eventually won}) \geq E_\pi(\text{prize eventually won}) \tag{2}$$

Index theorem for golf with N balls

Suppose the golfer keeps playing until a ball goes in the hole.

His prize is the prevailing prize of the ball he sinks.

Prevailing prizes are defined in such a way that the golfer cannot make a strictly positive profit, and so for any policy σ,

$$E_\sigma(\text{cost incurred}) \geq E_\sigma(\text{prize eventually won}) \tag{1}$$

Let π be the policy: *always play the ball with least prevailing prize.*

Because each ball's sequence of prevailing prizes is nondecreasing.

$$E_\sigma(\text{prize eventually won}) \geq E_\pi(\text{prize eventually won}) \tag{2}$$

But the golfer breaks even under π.

$$E_\pi(\text{prize eventually won}) = E_\pi(\text{cost incurred}) \tag{3}$$

Golf and the multi-armed bandit

Having solved the golf problem, the solution to the multi-armed bandit problem follows. Just let $P(x, 0) = 1 - \beta$ for all x.

The expected cost incurred until a first ball is sunk equals the expected total β-discounted cost over the infinite horizon.

Golf and the multi-armed bandit

Having solved the golf problem, the solution to the multi-armed bandit problem follows. Just let $P(x, 0) = 1 - \beta$ for all x.

The expected cost incurred until a first ball is sunk equals the expected total β-discounted cost over the infinite horizon.

The fair prize, $g(x)$, is $1/(1 - \beta)$ times the Gittins index, $G(x)$.

Golf and the multi-armed bandit

Having solved the golf problem, the solution to the multi-armed bandit problem follows. Just let $P(x,0)=1-\beta$ for all x.

The expected cost incurred until a first ball is sunk equals the expected total β-discounted cost over the infinite horizon.

The fair prize, $g(x)$, is $1/(1-\beta)$ times the Gittins index, $G(x)$.

$$\begin{aligned} g(x) = \inf\Biggl\{ g \ : \ \sup_{\tau\geq 1} E\Biggl[\sum_{t=0}^{\tau-1} -c(x(t))\beta^t \\ + (1-\beta)(1+\beta+\cdots+\beta^{\tau-1})g \,\Biggl|\, x(0)=x \Biggr] \geq 0 \Biggr\} \\ = \frac{1}{1-\beta} \inf_{\tau\geq 1} \frac{E\left[\sum_{t=0}^{\tau-1} c(x(t))\beta^t \,\middle|\, x(0)=x\right]}{E\left[\sum_{t=0}^{\tau-1} \beta^t \,\middle|\, x(0)=x\right]} \end{aligned}$$

Gittins index theorem and Weitzman's problem

Theorem (Gittins index theorem, 1972) *The problem posed by a family of alternative bandit processes, is solved by always continuing the bandit process having the greatest Gittins index.*

Compare this to the solution to the Weitzman's problem which is

Theorem (Weitzman's Pandora rule, 1979). *Pandora's problem is solved by always opening the unopened box with greatest reservation value, until all reservations values are less than the greatest prize that has been found.*

Pandora plays golf

Learn how to play golf with more than one ball.

- You can solve Weitzman's Pandora's boxes problem.

Pandora plays golf

Learn how to play golf with more than one ball.

- You can solve Weitzman's Pandora's boxes problem.

  - Each of Pandora's boxes is a ball, starting in state 1, say.
  - First time ball i is hit a cost c_i is incurred, and ball lands at location x_i (chosen as a sample from F_i).
  - Second time ball i is hit, (from its current state x_i), a cost $-x_i$ is incurred, the ball goes in the hole and the game ends.

Pandora plays golf

Learn how to play golf with more than one ball.

- You can solve Weitzman's Pandora's boxes problem.

- Each of Pandora's boxes is a ball, starting in state 1, say.
- First time ball i is hit a cost c_i is incurred, and ball lands at location x_i (chosen as a sample from F_i).
- Second time ball i is hit, (from its current state x_i), a cost $-x_i$ is incurred, the ball goes in the hole and the game ends.

Problem of minimizing the expected cost of putting a ball in the hole

$\equiv$

Problem of maximizing the expected value of Pandora's greatest discovered prize, net of costs of opening boxes.

Gittins $\implies$ Pandora, mentioned by Chade and Smith (2006)

Summary of Lecture 1

- Interchange arguments
- Index policies
- Pandora's problem
- Achievable region method for $M/M/1$ queue
- Bandit processes
- Gittins index theorem
- Golf with many balls
- Solution to Pandora's problem

Tutorial

Bandit Processes and Index Policies

Richard Weber, University of Cambridge

Young European Queueing Theorists (YEQT VII) workshop on
Scheduling and priorities in queueing systems,
Eindhoven, November 4–5–6, 2013

Summary of Lecture 1

Warmup

- Interchange arguments
- Index policies
- Pandora's problem
- Achievable region method for $M/M/1$ queue

Bandit processes

- Bandit processes
- Gittins index theorem
- Golf with many balls
- Solution to Pandora's problem

How do you make perfect toast?

There is a rule for timing toast,
One never has to guess,
Just wait until it starts to smoke,
then 7 seconds less.

(David Kendall)

Summary of Lecture 2

Index policies for multi-class queues

- $M/M/1$ (preemptive)
- Achievable region method
- Branching bandits
- Tax problems
- $M/G/1$ (nonpreemptive)

Index policies for restless bandits

- Restless bandits
- Whittle index
- Asymptopic optimality
- Risk-sensitive indices

Multi-class queues

Multi-class $M/M/1$ preemptive

- Let the average work-in-system of class i be $x_i^\pi = E_\pi[N_i]/\mu_i$. Find linear inequalities that must be satisfied by the x_i^π. These define a polytope P, such that achievable x^π are in P. In particular for any $S \subseteq E = \{1, 2, \dots, k\}$, consider

$$\sum_{i \in S} x_i^\pi = \text{average work-in-system due to job classes in } S$$
$$\leq f(S)$$

for some $f(S)$. Equality achieved by always giving priority to jobs of classes not in S.

Multi-class $M/M/1$ preemptive

- Let the average work-in-system of class i be $x_i^\pi = E_\pi[N_i]/\mu_i$. Find linear inequalities that must be satisfied by the x_i^π. These define a polytope P, such that achievable x^π are in P. In particular for any $S \subseteq E = \{1, 2, \dots, k\}$, consider

$$\sum_{i \in S} x_i^\pi = \text{average work-in-system due to job classes in } S$$
$$\leq f(S)$$

  for some $f(S)$. Equality achieved by always giving priority to jobs of classes not in S.

- Minimize $\sum_i c_i \mu_i x_i$ over $x \in P$.

Multi-class $M/M/1$ preemptive

- Let the average work-in-system of class i be $x_i^\pi = E_\pi[N_i]/\mu_i$.
  Find linear inequalities that must be satisfied by the x_i^π.
  These define a polytope P, such that achievable x^π are in P.
  In particular for any $S \subseteq E = \{1, 2, \ldots, k\}$, consider

  $$\sum_{i \in S} x_i^\pi = \text{average work-in-system due to job classes in } S$$
  $$\leq f(S)$$

  for some $f(S)$. Equality achieved by always giving priority to jobs of classes not in S.
- Minimize $\sum_i c_i \mu_i x_i$ over $x \in P$.
- Show optimal solution, x^*, is achievable by index policy that prioritizes jobs in decreasing order of $c_i \mu_i$.

Proof of $c\mu$-rule optimality

Linear program relaxation:

minimize $\sum_i c_i \mu_i x_i$

$\sum_{i \in S} x_i \leq f(S), \quad$ for all $S \subset E$

$\sum_{i \in E} x_i = f(E)$

$x_i \geq 0$

Proof of $c\mu$-rule optimality

Linear program relaxation:

minimize $\sum_i c_i \mu_i x_i$

$\sum_{i \in S} x_i \leq f(S), \quad$ for all $S \subset E$

$\sum_{i \in E} x_i = f(E)$

$x_i \geq 0$

Dual linear program:

maximize $\sum_S y_S f(S)$

$\sum_{S: i \in S} y_S \leq c_i \mu_i, \quad$ for all i

$y_S \leq 0, \quad S \subset E$

y_E is unconstrained

Proof of $c\mu$-rule optimality

Linear program relaxation:

minimize $\sum_i c_i \mu_i x_i$

$\sum_{i \in S} x_i \leq f(S), \quad$ for all $S \subset E$

$\sum_{i \in E} x_i = f(E)$

$x_i \geq 0$

Dual linear program:

maximize $\sum_S y_S f(S)$

$\sum_{S : i \in S} y_S \leq c_i \mu_i, \quad$ for all i

$y_S \leq 0, \quad S \subset E$

y_E is unconstrained

Assume $c_1\mu_1 > \cdots > c_k\mu_k$. Let π be policy $1 \to 2 \to \cdots \to k$.

Denote $S_j = \{j, \ldots, k\}$.

Proof of $c\mu$-rule optimality

Linear program relaxation:

minimize $\sum_i c_i\mu_i x_i$

$\sum_{i\in S} x_i \leq f(S)$, for all $S \subset E$

$\sum_{i\in E} x_i = f(E)$

$x_i \geq 0$

Dual linear program:

maximize $\sum_S y_S f(S)$

$\sum_{S:i\in S} y_S \leq c_i\mu_i$, for all i

$y_S \leq 0$, $S \subset E$

y_E is unconstrained

Assume $c_1\mu_1 > \cdots > c_k\mu_k$. Let π be policy $1 \to 2 \to \cdots \to k$.

Denote $S_j = \{j, \ldots, k\}$.
Taking $x = x^\pi \implies$
$\sum_{i\in S_j} x_i = f(S_j)$, $j = 1, \ldots, k$,

Primal feasibility holds.
Complementary slackness and dual feasibilibilty hold if set y_S as follows:

Proof of $c\mu$-rule optimality

Linear program relaxation:

minimize $\sum_i c_i\mu_i x_i$

$\sum_{i\in S} x_i \leq f(S)$, for all $S \subset E$

$\sum_{i\in E} x_i = f(E)$

$x_i \geq 0$

Dual linear program:

maximize $\sum_S y_S f(S)$

$\sum_{S:i\in S} y_S \leq c_i\mu_i$, for all i

$y_S \leq 0$, $S \subset E$

y_E is unconstrained

Assume $c_1\mu_1 > \cdots > c_k\mu_k$. Let π be policy $1 \to 2 \to \cdots \to k$.

Denote $S_j = \{j, \dots, k\}$.
Taking $x = x^\pi \implies$
$\sum_{i\in S_j} x_i = f(S_j)$, $j = 1, \dots, k$,

Primal feasibility holds.
Complementary slackness and dual feasibilty hold if set y_S as follows:

$$\begin{aligned} y_E &= c_1\mu_1, \\ y_{S_2} &= c_2\mu_2 - c_1\mu_1, \\ y_{S_3} &= c_3\mu_3 - c_2\mu_2, \\ &\vdots \\ y_{S_k} &= c_k\mu_k - c_{k-1}\mu_{k-1}, \end{aligned}$$

and all other $y_S = 0$.

Proof of $c\mu$-rule optimality

Linear program relaxation:

minimize $\sum_i c_i \mu_i x_i$

$\sum_{i \in S} x_i \leq f(S)$, for all $S \subset E$

$\sum_{i \in E} x_i = f(E)$

$x_i \geq 0$

Dual linear program:

maximize $\sum_S y_S f(S)$

$\sum_{S : i \in S} y_S \leq c_i \mu_i$, for all i

$y_S \leq 0$, $S \subset E$

y_E is unconstrained

Assume $c_1\mu_1 > \cdots > c_k\mu_k$. Let π be policy $1 \to 2 \to \cdots \to k$.

Denote $S_j = \{j, \ldots, k\}$.
Taking $x = x^\pi \implies$
$\sum_{i \in S_j} x_i = f(S_j)$, $j = 1, \ldots, k$,

Primal feasibility holds.
Complementary slackness and dual feasibilibty hold if set y_S as follows:

$$y_E = c_1\mu_1,$$
$$y_{S_2} = c_2\mu_2 - c_1\mu_1,$$
$$y_{S_3} = c_3\mu_3 - c_2\mu_2,$$
$$\vdots$$
$$y_{S_k} = c_k\mu_k - c_{k-1}\mu_{k-1},$$

and all other $y_S = 0$.

$\implies$ π is optimal.

Nonpreemptive multi-class $M/G/1$

- Jobs again of k classes $\{1, \dots, k\}$. Holding cost rates c_i.
- Class i job has service time t_i, chosen $\sim F_i$.
- While processing it a random number of new jobs of type j arrive, distributed as a Poisson random variable with mean $\lambda_j t_i$.

Nonpreemptive multi-class $M/G/1$

- Jobs again of k classes $\{1, \ldots, k\}$. Holding cost rates c_i.
- Class i job has service time t_i, chosen $\sim F_i$.
- While processing it a random number of new jobs of type j arrive, distributed as a Poisson random variable with mean $\lambda_j t_i$.

  Example of a **branching bandit**.

  Gittins index theorem holds.

  (Proof similar to that is last lecture
  — but when hit, golf ball splits into many golf balls.)
- Problem: minimize time-average holding cost.

Nonpreemptive multi-class $M/G/1$

Let $\mu_i^{-1} = Et_i$.

$c_1\mu_1 > \cdots > c_k\mu_k$.

Assume Gittins index theorem for branching bandits.

We now prove that the nonpreemptive scheduling policy that minimizes the expected weighted holding cost is the $c\mu$-rule, the priority policy $\pi : 1 \to 2 \to \cdots \to k$.

Tax problems

Consider a job which enters at time 0, leaves at time t, and pays tax in between: *discounted* cost is

$$\int_0^t c_i e^{-\alpha s} ds = \frac{1}{\alpha}[c_i - c_i e^{-\alpha t}].$$

Tax problems

Consider a job which enters at time 0, leaves at time t, and pays tax in between: *discounted* cost is

$$\int_0^t c_i e^{-\alpha s} ds = \frac{1}{\alpha}[c_i - c_i e^{-\alpha t}].$$

An alternative view (on the r.h.s.) is that we

- 'pay c_i/α on entry',
- 'receive refund c_i/α on exit' (with discount factor of $e^{-\alpha t}$ applied).

If we cannot control when jobs enter, then we just want to maximize collection of refunds.

Gittins index in tax problems

The Gittins index is

$$G_i = \sup_{\tau} \frac{E[\text{sum of discounted refunds collected up to } \tau]}{E[\text{integral of discounted time up to } \tau]}$$

which in the limit $\alpha \to 0$

$$\to \sup_{\tau} \frac{E[\text{sum of refunds collected up to } \tau]}{E\tau}$$

(A policy which is α-discount optimal for all sufficiently small α is average-cost optimal.)

Nonpreemptive $M/G/1$ queue

Suppose that $G_1 > G_2 > \cdots > G_k$.

$$G_1 = \sup_{\tau} \frac{E[\text{sum of refunds collected up to } \tau]}{E\tau} = \frac{c_1}{Et_1} = c_1\mu_1.$$

To find G_i we start with one class i job, process it, and then any ‘daughters’ in classes $1, \dots, i-1$ until the system is again clear of jobs in classes $1, \dots, i-1$.

Nonpreemptive $M/G/1$ queue

Suppose that $G_1 > G_2 > \cdots > G_k$.

$$G_1 = \sup_{\tau} \frac{E[\text{sum of refunds collected up to } \tau]}{E\tau} = \frac{c_1}{Et_1} = c_1\mu_1.$$

To find G_i we start with one class i job, process it, and then any 'daughters' in classes $1, \ldots, i-1$ until the system is again clear of jobs in classes $1, \ldots, i-1$.

Poisson arrivals $\implies$ expected refunds C, and expected time T, accumulating during clearing is proportional to t_i. So for some θ,

$$G_i = \frac{E[c_i + (\theta t_i)C]}{E[t_i + (\theta t_i)T]}.$$

Nonpreemptive $M/G/1$ queue

Similarly, we might start with one job in class $i-1$, process it, and then clear all daughter jobs in classes $1,\dots,i-2$.

But the index calculation gives the same value if we also clear all daughter class $i-1$ jobs, and so for the same θ as above

$$G_{i-1} = \frac{E[c_{i-1} + (\theta t_{i-1})C]}{E[t_{i-1} + (\theta t_{i-1})T]}$$

Nonpreemptive $M/G/1$ queue

Similarly, we might start with one job in class $i-1$, process it, and then clear all daughter jobs in classes $1,\dots,i-2$.

But the index calculation gives the same value if we also clear all daughter class $i-1$ jobs, and so for the same θ as above

$$G_{i-1} = \frac{E[c_{i-1} + (\theta t_{i-1})C]}{E[t_{i-1} + (\theta t_{i-1})T]}$$

$$G_i = \frac{E[c_i + (\theta t_i)C]}{E[t_i + (\theta t_i)T]}.$$

$$G_{i-1} \geq G_i \implies c_{i-1}/Et_{i-1} \geq c_i/Et_i.$$

Optimality of $c\mu$-rule in multi-class $M/G/1$

Gittins index theorem for branching bandits
$+$ Gittins indices ordered the same as $c_i\mu_i \implies$

Theorem

The average waiting cost in a multi-class $M/G/1$ queue is minimized (amongst nonpreemptive strategies) by always processing the job of greatest $c_i\mu_i$, where $\mu_i = 1/Et_i$.

Notice that the G_i do depend on the arrival rates, but their ordering does not.

Restless bandits

Spinning plates

Restless bandits

[Whittle '88]

- Two actions are available: **active** $(a = 1)$ or **passive** $(a = 0)$.

Restless bandits

[Whittle ‘88]

- Two actions are available: **active** $(a = 1)$ or **passive** $(a = 0)$.
- Rewards, $r(x, a)$, and transitions, $P(y \mid x, a)$, depend on the state and the action taken.

Restless bandits

[Whittle '88]

- Two actions are available: **active** $(a = 1)$ or **passive** $(a = 0)$.
- Rewards, $r(x, a)$, and transitions, $P(y \mid x, a)$, depend on the state and the action taken.
- **Objective**: Maximize time-average reward from n restless bandits under a constraint that only m $(m < n)$ of them receive the active action simultaneously.

Restless bandits

[Whittle '88]

- Two actions are available: **active** $(a = 1)$ or **passive** $(a = 0)$.
- Rewards, $r(x, a)$, and transitions, $P(y \mid x, a)$, depend on the state and the action taken.
- **Objective**: Maximize time-average reward from n restless bandits under a constraint that only m $(m < n)$ of them receive the active action simultaneously.

active $a = 1$	passive $a = 0$
work, increasing fatigue	rest, recovery

Restless bandits

[Whittle ‘88]

- Two actions are available: **active** $(a = 1)$ or **passive** $(a = 0)$.
- Rewards, $r(x, a)$, and transitions, $P(y \,|\, x, a)$, depend on the state and the action taken.
- **Objective**: Maximize time-average reward from n restless bandits under a constraint that only m $(m < n)$ of them receive the active action simultaneously.

active $a = 1$	passive $a = 0$
work, increasing fatigue	rest, recovery
high speed	low speed

$$P(y \,|\, x, 0) = \epsilon P(y \,|\, x, 1)\,, \quad y \neq x$$

Restless bandits

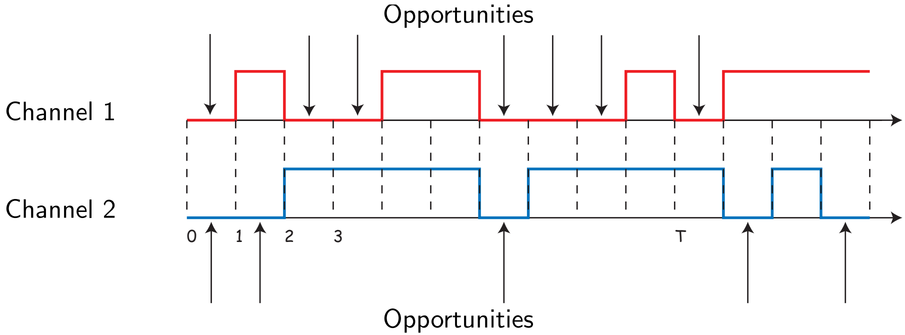
[Whittle ‘88]

- Two actions are available: **active** $(a = 1)$ or **passive** $(a = 0)$.
- Rewards, $r(x, a)$, and transitions, $P(y \mid x, a)$, depend on the state and the action taken.
- **Objective**: Maximize time-average reward from n restless bandits under a constraint that only m $(m < n)$ of them receive the active action simultaneously.

active $a = 1$	passive $a = 0$
work, increasing fatigue	rest, recovery
high speed	low speed
inspection	no inspection

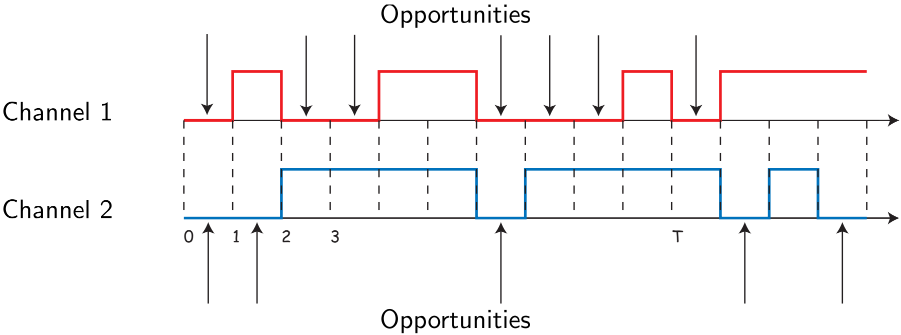
Opportunistic spectrum access

Communication channels may be busy or free.



Opportunistic spectrum access

Communication channels may be busy or free.



Aim is to ‘inspect’ m out of n channels, maximizing the number of these that are found to be free.

Relaxed problem for a single restless bandit

Consider a **relaxed problem**, posed for 1 bandit only.

Seek to maximize average reward obtained from this bandit under a constraint that $a = 1$ for only a fraction $\rho = m/n$ of the time.

LP for the relaxed problem

Let z_x^a be proportion of time that the bandit is in state x and action a is taken (under a stationary Markov policy).

An upper bound for our problem can found from a LP in variables $\{z_x^a : x \in E,\ a \in \{0,1\}\}$:

$$\text{maximize } \sum_{x,a} r(x,a) z_x^a$$

LP for the relaxed problem

Let z_x^a be proportion of time that the bandit is in state x and action a is taken (under a stationary Markov policy).

An upper bound for our problem can found from a LP in variables $\{z_x^a : x \in E,\ a \in \{0,1\}\}$:

$$\text{maximize } \sum_{x,a} r(x,a) z_x^a$$

$$\text{s.t. } z_x^a \geq 0\,,\ \text{ for all } x, a$$

LP for the relaxed problem

Let z_x^a be proportion of time that the bandit is in state x and action a is taken (under a stationary Markov policy).

An upper bound for our problem can found from a LP in variables $\{z_x^a \,:\, x \in E,\ a \in \{0,1\}\}$:

$$\text{maximize } \sum_{x,a} r(x,a) z_x^a$$

$$\text{s.t. } z_x^a \geq 0\,,\ \text{for all } x, a\,;\ \sum_{x,a} z_x^a = 1$$

LP for the relaxed problem

Let z_x^a be proportion of time that the bandit is in state x and action a is taken (under a stationary Markov policy).

An upper bound for our problem can found from a LP in variables $\{z_x^a : x \in E,\ a \in \{0,1\}\}$:

$$\text{maximize } \sum_{x,a} r(x,a) z_x^a$$

$$\text{s.t. } z_x^a \geq 0\,,\ \text{for all } x, a\,;\ \sum_{x,a} z_x^a = 1\,;$$

$$\sum_a z_x^a = \sum_y z_y^a P(x \,|\, y, a(y))\,,\ \text{for all } x$$

LP for the relaxed problem

Let z_x^a be proportion of time that the bandit is in state x and action a is taken (under a stationary Markov policy).

An upper bound for our problem can found from a LP in variables $\{z_x^a : x \in E,\ a \in \{0,1\}\}$:

$$\text{maximize } \sum_{x,a} r(x,a) z_x^a$$

$$\text{s.t. } z_x^a \geq 0\,,\ \text{for all } x, a\,;\ \sum_{x,a} z_x^a = 1\,;$$

$$\sum_a z_x^a = \sum_y z_y^a P(x \mid y, a(y))\,,\ \text{for all } x\,; \quad \sum_x z_x^0 = 1 - \rho\,.$$

The subsidy problem

Optimal value of the dual LP problem is g, where this can be found from the average-cost dynamic programming equation

$$\phi(x) + g = \max_{a\in\{0,1\}} \left\{ r(x,a) + \lambda(1-a) + \sum_y \phi(y) P(y \,|\, x,a) \right\}.$$

g, $\phi(x)$ and λ are the Lagrange multipliers for constraints.

λ may be interpreted as a *subsidy* for taking $a = 0$.

The subsidy problem

Optimal value of the dual LP problem is g, where this can be found from the average-cost dynamic programming equation

$$\phi(x) + g = \max_{a\in\{0,1\}} \left\{ r(x,a) + \lambda(1-a) + \sum_y \phi(y) P(y \mid x, a) \right\}.$$

g, $\phi(x)$ and λ are the Lagrange multipliers for constraints.

λ may be interpreted as a *subsidy* for taking $a = 0$.

Solution partitions state space into sets: E_0 $(a = 0)$, E_1 $(a = 1)$ and E_{01} (randomization between $a = 0$ and $a = 1$).

Indexability

Reasonable that as the subsidy λ (for $a = 0$) increases from $-\infty$ to $+\infty$ the set of states E_0 (where $a = 0$ optimal) should increase monotonically.

If it does then we say the bandit is **indexable**.

Indexability

Reasonable that as the subsidy λ (for $a = 0$) increases from $-\infty$ to $+\infty$ the set of states E_0 (where $a = 0$ optimal) should increase monotonically.

If it does then we say the bandit is **indexable**.

Whittle index, $W(x)$, is the least subsidy for which it can be optimal to take $a = 0$ in state x.

Indexability

Reasonable that as the subsidy λ (for $a = 0$) increases from $-\infty$ to $+\infty$ the set of states E_0 (where $a = 0$ optimal) should increase monotonically.

If it does then we say the bandit is **indexable**.

Whittle index, $W(x)$, is the least subsidy for which it can be optimal to take $a = 0$ in state x.

This motivates a heuristic policy:

Apply the active action to the m bandits with the greatest Whittle indices.

Indexability

Reasonable that as the subsidy λ (for $a = 0$) increases from $-\infty$ to $+\infty$ the set of states E_0 (where $a = 0$ optimal) should increase monotonically.

If it does then we say the bandit is **indexable**.

Whittle index, $W(x)$, is the least subsidy for which it can be optimal to take $a = 0$ in state x.

This motivates a heuristic policy:

Apply the active action to the m bandits with the greatest Whittle indices.

Like Gittins indices for classical bandits, Whittle indices can be computed separately for each bandit.

Same as the Gittins index when $a = 0$ is freezing action.

Two questions

- **Under what assumptions is a restless bandit indexable?**

Two questions

- **Under what assumptions is a restless bandit indexable?**

  This is somewhat mysterious.

  Special classes of restless bandits are indexable: such as ‘dual speed’, Glazebrook, Niño-Mora, Ansell (2002), W. (2007).

Two questions

- **Under what assumptions is a restless bandit indexable?**

  This is somewhat mysterious.

  Special classes of restless bandits are indexable: such as ‘dual speed’, Glazebrook, Niño-Mora, Ansell (2002), W. (2007).

  Indexability can be proved in some problems (such as the opportunistic spectrum access problem, Liu and Zhao (2009)).

Two questions

- **Under what assumptions is a restless bandit indexable?**

  This is somewhat mysterious.

  Special classes of restless bandits are indexable: such as ‘dual speed’, Glazebrook, Niño-Mora, Ansell (2002), W. (2007).

  Indexability can be proved in some problems (such as the opportunistic spectrum access problem, Liu and Zhao (2009)).

- **How good is the heuristic policy using Whittle indices?**

Two questions

- **Under what assumptions is a restless bandit indexable?**

  This is somewhat mysterious.

  Special classes of restless bandits are indexable: such as 'dual speed', Glazebrook, Niño-Mora, Ansell (2002), W. (2007).

  Indexability can be proved in some problems (such as the opportunistic spectrum access problem, Liu and Zhao (2009)).

- **How good is the heuristic policy using Whittle indices?**

  It may be optimal. (opportunistic spectrum access — identical channels, Ahmad, Liu, Javidi, and Zhao (2009)).

Two questions

- **Under what assumptions is a restless bandit indexable?**

  This is somewhat mysterious.

  Special classes of restless bandits are indexable: such as 'dual speed', Glazebrook, Niño-Mora, Ansell (2002), W. (2007).

  Indexability can be proved in some problems (such as the opportunistic spectrum access problem, Liu and Zhao (2009)).

- **How good is the heuristic policy using Whittle indices?**

  It may be optimal. (opportunistic spectrum access — identical channels, Ahmad, Liu, Javidi, and Zhao (2009)).

  Lots of papers with numerical work.

Two questions

- **Under what assumptions is a restless bandit indexable?**

  This is somewhat mysterious.

  Special classes of restless bandits are indexable: such as ‘dual speed’, Glazebrook, Niño-Mora, Ansell (2002), W. (2007).

  Indexability can be proved in some problems (such as the opportunistic spectrum access problem, Liu and Zhao (2009)).

- **How good is the heuristic policy using Whittle indices?**

  It may be optimal. (opportunistic spectrum access — identical channels, Ahmad, Liu, Javidi, and Zhao (2009)).

  Lots of papers with numerical work.

  It is often asymptotically optimal, W. and Weiss (1990).

Asymptotic optimality

At time t there are $(n_1, \ldots, n_k)$ bandits in states $1, \ldots, k$. Suppose a priority policy orders the states $1, 2, \ldots$. Let

$m = \rho n$.

Asymptotic optimality

At time t there are $(n_1, \ldots, n_k)$ bandits in states $1, \ldots, k$. Suppose a priority policy orders the states $1, 2, \ldots$. Let

$m = \rho n$.

$z_i = n_i/n$ be proportion in state i.

Asymptotic optimality

At time t there are $(n_1, \dots, n_k)$ bandits in states $1, \dots, k$. Suppose a priority policy orders the states $1, 2, \dots$. Let

$m = \rho n$.

$z_i = n_i/n$ be proportion in state i.

$n_i^a =$ number that receive action a.

Asymptotic optimality

At time t there are $(n_1, \ldots, n_k)$ bandits in states $1, \ldots, k$.
Suppose a priority policy orders the states $1, 2, \ldots$. Let

$m = \rho n$.

$z_i = n_i/n$ be proportion in state i.

$n_i^a =$ number that receive action a.

$u_i^a(z) = n_i^a/n_i$.

Asymptotic optimality

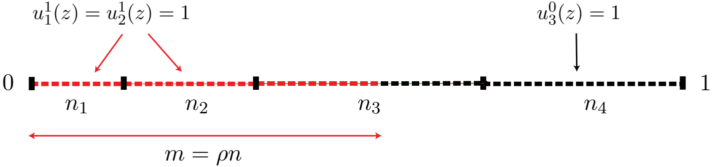
At time t there are $(n_1, \dots, n_k)$ bandits in states $1, \dots, k$. Suppose a priority policy orders the states $1, 2, \dots$. Let

$m = \rho n$.

$z_i = n_i/n$ be proportion in state i.

$n_i^a =$ number that receive action a.

$u_i^a(z) = n_i^a/n_i$.



Asymptotic optimality

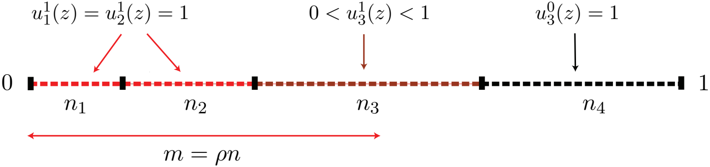
At time t there are $(n_1, \ldots, n_k)$ bandits in states $1, \ldots, k$. Suppose a priority policy orders the states $1, 2, \ldots$. Let

$m = \rho n$.

$z_i = n_i/n$ be proportion in state i.

$n_i^a =$ number that receive action a.

$u_i^a(z) = n_i^a/n_i$.



Asymptotic optimality

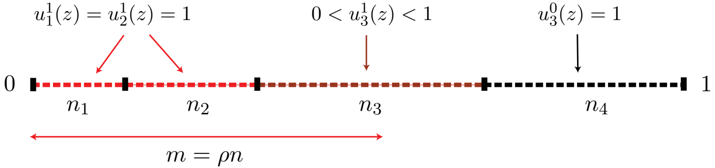
At time t there are $(n_1, \ldots, n_k)$ bandits in states $1, \ldots, k$. Suppose a priority policy orders the states $1, 2, \ldots$. Let

$m = \rho n$.

$z_i = n_i/n$ be proportion in state i.

$n_i^a =$ number that receive action a.

$u_i^a(z) = n_i^a/n_i$.



$q_{ij}^a =$ rate a bandit in state i jumps to state j under action a;

$$q_{ij}(z) = u_i^0(z)q_{ij}^0 + u_i^1(z)q_{ij}^1$$

Fluid model approximation

Now suppose n is large. Transitions will be happening very fast.

By the law of large numbers we expect that the ‘path’ $z(t)$ evolves close to the fluid model in which

$$dz_i/dt = \sum_j [u_j q^1_{ji} + (1 - u_j) q^0_{ji}] z_j - z_i \sum_j [u_i q^1_{ij} + (1 - u_i) q^0_{ij}]$$

where $u_j = u_j(z)$ is a function of z so that $\sum_i u_i(z) z_i = \rho$.

Fluid model approximation

Now suppose n is large. Transitions will be happening very fast.

By the law of large numbers we expect that the 'path' $z(t)$ evolves close to the fluid model in which

$$dz_i/dt = \sum_j [u_j q_{ji}^1 + (1-u_j) q_{ji}^0] z_j - z_i \sum_j [u_i q_{ij}^1 + (1-u_i) q_{ij}^0]$$

where $u_j = u_j(z)$ is a function of z so that $\sum_i u_i(z) z_i = \rho$.

Suppose, in order of Whittle index, the states are $1, \dots, k$.

For $z_1 + \cdots + z_{h-1} < \rho$, $z_1 + \cdots + z_{h+1} > \rho$, we have

$$u_j(z) = \begin{cases} 1 & j < h \\ \frac{\rho - \sum_{i<h} z_i}{z_j} & j = h \\ 0 & j > h \end{cases}$$

Fluid approximation

But this is not so bad. For $i < h$,

$$dz_i/dt = \sum_{j<h} q^1_{ji} z_j + \sum_{j>h} q^0_{ji} z_j + \left(\rho - \sum_{j<h} z_j\right) q^1_{hi}$$
$$+ \left(\sum_{j\le h} z_j - \rho\right) q^0_{hi} - z_i \sum_j q^1_{ij}$$

Similar expressions hold for $i > h$ and $i = h$.

The general form is

$$dz/dt = A(z)z + b(z)$$

where $A(z)$ and $b(z)$ are constants within polyhedral regions.

Asymptotic optimality

$$dz/dt = A(z)z + b(z)$$

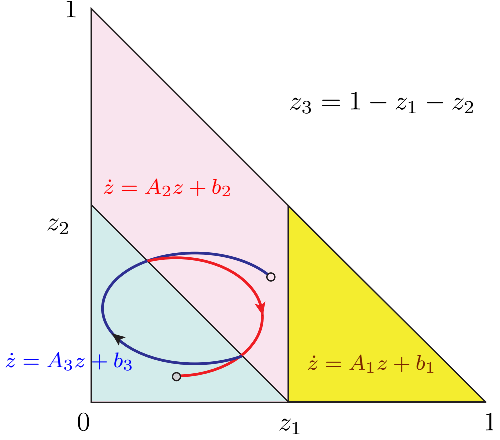
System has an asymptotically stable equilibrium point if from any start $z(0)$, $z(t) \to z^*$ as $t \to \infty$.

Theorem [W. and Weiss ‘90]

If bandits are indexable, and the fluid model for the Whittle index policy has an asymptotically stable equilibrium point, then the Whittle index policy is asymptotically optimal, — in the sense that the reward per bandit tends to the reward that is obtained under the relaxed policy.

(proof via large deviation theory for sample paths.)

Possibility of limit cycle



$k = 3$ and $\rho = 1/2$. Dynamics differ in the regions $z_1 > 1/2$,
$z_1 < 1/2$ and $z_1 + z_2 > 1/2$;
$z_1 + z_2 < 1/2$.

Heuristic may not be asymptotically optimal

$$\left(q^0_{ij}\right) = \begin{pmatrix} -2 & 1 & 0 & 1 \\ 2 & -2 & 0 & 0 \\ 0 & 56 & -\frac{113}{2} & \frac{1}{2} \\ 1 & 1 & \frac{1}{2} & -\frac{5}{2} \end{pmatrix}, \quad \left(q^1_{ij}\right) = \begin{pmatrix} -2 & 1 & 0 & 1 \\ 2 & -2 & 0 & 0 \\ 0 & \frac{7}{25} & -\frac{113}{400} & \frac{1}{400} \\ 1 & 1 & \frac{1}{2} & -\frac{5}{2} \end{pmatrix}$$

$$r^0 = (0, 1, 10, 10)\,, \quad r^1 = (10, 10, 10, 0)\,, \quad \rho = 0.835$$

Heuristic may not be asymptotically optimal

$$\left(q^0_{ij}\right) = \begin{pmatrix} -2 & 1 & 0 & 1 \\ 2 & -2 & 0 & 0 \\ 0 & 56 & -\frac{113}{2} & \frac{1}{2} \\ 1 & 1 & \frac{1}{2} & -\frac{5}{2} \end{pmatrix}, \quad \left(q^1_{ij}\right) = \begin{pmatrix} -2 & 1 & 0 & 1 \\ 2 & -2 & 0 & 0 \\ 0 & \frac{7}{25} & -\frac{113}{400} & \frac{1}{400} \\ 1 & 1 & \frac{1}{2} & -\frac{5}{2} \end{pmatrix}$$

$$r^0 = (0, 1, 10, 10)\,, \quad r^1 = (10, 10, 10, 0)\,, \quad \rho = 0.835$$

Bandit is indexable.

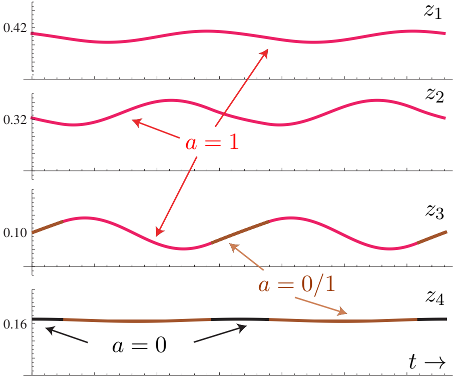
Equilibrium point is $(\bar{z}_1, \bar{z}_2, \bar{z}_3, \bar{z}_4) = (0.409, 0.327, 0.100, 0.164)$.

$\bar{z}_1 + \bar{z}_2 + \bar{z}_3 = 0.836$.

Relaxed policy obtains 10 per bandit per unit time.

Heuristic is not asymptotically optimal

But equilibrium point $\bar{z}$ is not asymptotically stable.



Relaxed policy obtains 10 per bandit.

Heuristic obtains only 9.9993 per bandit.

Equilibrium point optimization

Recall the fluid model equilibrium point optimization problem

$$\text{minimize}\ \ c(z,u):\ \ \dot{z}=a(z,u)=0,\ \textstyle\sum_i z_i u_i=\rho,\ \textstyle\sum_i z_i=1,$$

where

$$\begin{aligned} c(z,u) &= \textstyle\sum_i [c_i^1 u_i + c_i^0(1-u_i)]z_i \\ a(z,u) &= \textstyle\sum_j [u_j q_{ji}^1 + (1-u_j)q_{ji}^0]z_j - z_i \textstyle\sum_j [u_i q_{ij}^1 + (1-u_i)q_{ij}^0] \end{aligned}$$

Dual

$$\phi_i + g = \min\Big\{ c_i^0 - \lambda + \textstyle\sum_j q_{ij}^0 \phi_j\ ,\ c_i^1 + \textstyle\sum_j q_{ij}^1 \phi_j \Big\}$$

‘Large deviations’-inspired model

Suppose n is large, but we try to model more sensitively the dependence on n by considering deviations from the fluid model.

$$dz/dt = a(z,u) + \epsilon(z,u)$$

Assume

$$P(\epsilon(z,u)dt = \eta dt) \sim e^{-nI(z,u,\eta)dt}$$

(inspired by the theory of large deviations).

$I(z,u,0) = 0$ and $I(z,u,\eta)$ is convex increasing in η.

Risk-sensitive control

We are now positioned to consided a risk-sensitive performance measure

$$E\left[\int_0^T c(z,u)dt\right] \text{ replaced by } -\frac{1}{\theta}\log E\left[e^{-\theta\int_0^T c(z,u)dt}\right]$$

Risk-sensitive control

We are now positioned to consided a risk-sensitive performance measure

$$E\left[\int_0^T c(z,u)dt\right] \text{ replaced by } -\frac{1}{\theta}\log E\left[e^{-\theta\int_0^T c(z,u)dt}\right]$$

$$\approx E\left[\int_0^T c(z,u)\,dt\right] - \tfrac{1}{2}\theta\,\mathsf{var}\left(\int_0^T c(z,u)\,dt\right)$$

$\theta > 0$ corresponds to risk-seeking.

$\theta < 0$ corresponds to risk-aversion.

Large deviations

Consider a path $\{z(t), u(t), \epsilon(t),\ 0 \le t \le T\}$.

$$P(\text{path}) \times \text{cost}(\text{path}) = e^{-n \int_0^T I(z,u,\epsilon) dt} \times e^{-\theta \int_0^T c(z,u) dt}.$$

Let $\theta = \beta n$.

$E\Big(\exp(-\theta \int_0^T c(x,u) dt)\Big)$ is determined by summing the above over all paths, and so essentially

$$E\Big(\exp(-\theta \int_0^T c(x,u) dt)\Big)$$

$$\sim \exp\left(-n\beta \inf_{\epsilon} \left[\int_0^T c(z,u) + \beta^{-1} I(z,u,\epsilon) dt\right]\right).$$

Risk-sensitive optimal equilibrium point

The constraint $\dot{z} = a(z,u) + \epsilon$ can be imposed with a Lagrange multiplier, and so we seek z, u, ϵ, ϕ to extremize:

$$\int_0^T \Big[c(z,u) + \phi^T (\dot{z} - a(z,u) - \epsilon) + \beta^{-1} I(z,u,\epsilon) \Big] dt$$

One could now look for the extremizing path.

Risk-sensitive optimal equilibrium point

The constraint $\dot{z} = a(z,u) + \epsilon$ can be imposed with a Lagrange multiplier, and so we seek z, u, ϵ, ϕ to extremize:

$$\int_0^T \Big[c(z,u) + \phi^T(\dot{z} - a(z,u) - \epsilon) + \beta^{-1} I(z,u,\epsilon) \Big] dt$$

One could now look for the extremizing path.

But suppose a fixed point is reached at large T.

It is the point found by extremizing,

$$\begin{gathered} c(z,u) + \phi^T(-a(z,u) - \epsilon) + \beta^{-1} I(z,u,\epsilon) \\ + \lambda\Big(1 - \rho - \textstyle\sum_i z_i(1-u_i)\Big) \end{gathered}$$

Example

Consider a large number of components of a single type.

z_1, z_2 are the proportions of components in up and down states.

$u = 0$: up components go down at rate z_1;

$u = 1$: up components go down at rate z_1, and
down components go up at rate $z_2 = z_2 u$.

Stochastic model:

$$\dot{z}_2 = -z_2 u + z_1 + \epsilon_2$$

where $\epsilon_2 dt$ is Brownian motion with variance $z_1 + z_2 u$.

$$I(z, u, \epsilon_2) = \frac{\epsilon_2^2}{2(z_1 + z_2 u)}.$$

Risk-sensitive optimal equilibrium point

Suppose cost rate is $cz_2u - rz_1$.

In pursuit of an index we ask for what λ are both $u = 0$ and $u = 1$ optimal when extremizing (w.r.t. z_i, ϕ, ϵ_2)

$$cz_2 - rz_1 - \phi(-z_2u + z_1 + \epsilon_2) + \beta^{-1}\frac{\epsilon_2^2}{2(z_1 + z_2u)} - \lambda z_2(1 - u)$$

λ is a subsidy for not attending to components that are down.

Risk-sensitive optimal equilibrium point

Suppose cost rate is $cz_2u - rz_1$.

In pursuit of an index we ask for what λ are both $u = 0$ and $u = 1$ optimal when extremizing (w.r.t. z_i, ϕ, ϵ_2)

$$cz_2 - rz_1 - \phi(-z_2u + z_1 + \epsilon_2) + \beta^{-1}\frac{\epsilon_2^2}{2(z_1 + z_2u)} - \lambda z_2(1-u)$$

λ is a subsidy for not attending to components that are down.

... find a candidate for a risk-sensitive index:

$$\lambda_i^* = \frac{1}{2}(r_i - c_i) + \frac{1}{8}\beta(r_i + c_i)^2.$$

Consider two component types for which $r_1 - c_1 = r_2 - c_2$.

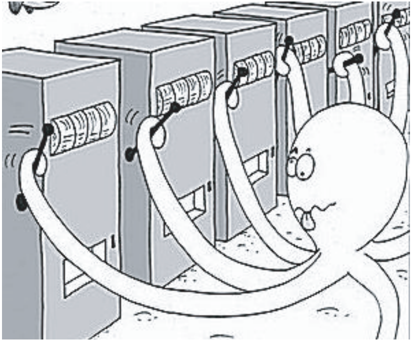
If $\beta = 0$ we should we are indifferent between these.

If $\beta > 0$, (risk-seeking), there is a distinct preference.

Summary of Lecture 2

- Multi-class $M/M/1$ (nonpreemptive)
- Achievable region method
- Multi-class $M/G/1$ (nonpreemptive)
- Branching bandits
- Tax problems
- Restless bandits
- Whittle index
- Asymptopic optimality
- Risk-sensitive indices

Questions



Reading list

D. Bertsimas and J. Niño-Mora, Conservation laws, extended polymatroids and multiarmed bandit problems; a polyhedral approach to indexable systems, Math. Operat Res., 21:257–306, 1996.

J. C. Gittins, K. D. Glazebrook, and R. R. Weber. Multiarmed Bandit Allocation Indices, (2nd editon), Wiley, 2011.

K. D. Glazebrook, D. J. Hodge, C. Kirkbride, R. J. Minty, Stochastic scheduling: A short history of index policies and new approaches to index generation for dynamic resource allocation, J Scheduling., 2013.

K. Liu, R. R. Weber and Q. Zhao. Indexability and Whittle Index for restless bandit problems involving reset processes, Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on, 12-15 December, 7690–7696, 2011,

R. R. Weber and G. Weiss. On an index policy for restless bandits. J Appl Probab, 27:637–648, 1990.

P. Whittle. Restless bandits: activity allocation in a changing world. In A Celebration of Applied Probability, ed. J. Gani, J Appl Probab, 25A:287–298, 1998.

P. Whittle. Risk-sensitivity, large deviations and stochastic control, Eur J Oper Res, 73:295-303, 1994