Index Policies

Gittins and Whittle Indices

Cambridge, November 06, 2016

Igor Kadota

Outline

- Introduction
  - Bandit Process, Objective Function
- Gittins Index
  - Index Theorem, Examples, Gittins Index, Proof
- Whittle Index
  - Three optimization problems, Decoupled Problem, Indexability, Whittle Index

Markov Bandit Process

- Markov decision process on **countable state space** E.
- Discrete decision times: $t \in \{0,1,2,\dots\}$.
- Controls applied at decision time t :
  - $u(t) = 0$ freezes the process and gives no reward;
  - $u(t) = 1$ continues the process and gives instantaneous reward $a^t r(\xi(t))$,

  where $\xi(t)$ is the state at time t, $a \in (0,1)$ is the discount factor and r(.) is the positive (and bounded) reward .
- State Transitions are instantaneous with $P(y|\xi)$ **when $\boldsymbol{u(t) = 1}$.**
- **Realization of the process "does <u>not</u> depend on the sequence of controls".**

Simple Family of Alternative Bandit Processes

- n Markov Bandit Processes with state space $\vec{E} = E_1 \times E_2 \times \cdots \times E_n$.
  - Notice that it is $|\vec{E}|$ is exponential on the number of bandits.

- Control $\boldsymbol{u}(\boldsymbol{t}) = \mathbf{1}$ **is applied to a single bandit $\boldsymbol{i_t}$** at each decision time t.
  - Control $u(t) = 0$ is applied to all **other bandits.**

- Sequence of selected bandits $\{\boldsymbol{i_1}, \boldsymbol{i_2}, \ldots,\}$
  State of the selected bandit $\boldsymbol{i_t}$ at each decision time t: $\xi_{\boldsymbol{i_t}}(t) = \xi_{\boldsymbol{i_t}}$.

- Reward accrued from the selected bandit: $a^t r_{i_t}(\xi_{i_t})$.

- Transition probability $P_{i_t}(y|\xi_{i_t})$. All other bandits remain in the same state.

Objective Function

- Problem: sequentially allocate effort between different processes so as to maximize the **infinite-horizon expected discounted sum of rewards**. Maximize:

$$J_\pi(\vec{\xi}\,) = \lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T-1} a^t r_{i_t}(\xi_{i_t}) \middle| \vec{\xi}(0) = \vec{\xi}\,\right]$$

- **At time t, we know** the state $\vec{\xi} = [\xi_1, \ldots, \xi_n]$, the probabilities $P_i(y|\xi_i)$, the discount factor a and the reward function $r_i(.)$ for each project.

- **Theorem**: for this problem, there is at least one optimal policy which is **deterministic, stationary and Markov**.
  - Thus, policy is a mapping from $\vec{E}$ to {1,2,…,n}.

[3] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Chapter 6, 2008.

Gittins Index

Multi Armed Bandit Problem

(open problem for almost 40 years)

Gittins Index

- Objective is to Maximize:

$$J_\pi(\vec{\xi}\,) = \lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T-1} a^t r_{i_t}(\xi_{i_t}) \,\middle|\, \vec{\xi}(0) = \vec{\xi}\,\right]$$

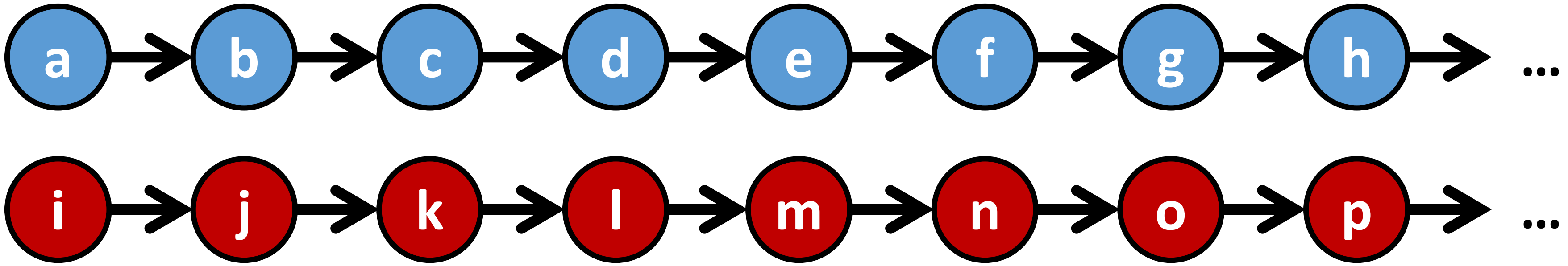
- **<u>Index Theorem</u>**: Optimal policy for this problem **is an Index policy.**
- **<u>Index policy</u>:** there exists a function $v_i(\xi_i)$, computed **separately for each bandit**, such that, for every state $\vec{\xi}$, the optimal policy continues the bandit:

$$i_t = \underset{i\in\{1,\dots,n\}}{\operatorname{argmax}} \{v_i(\xi_i)\}$$

Notice that computing the index is simple, for it only depends on the parameters associated with a single bandit. **But, how such function should be designed?**

Example 1

- Consider 2 bandits, each evolving according to a deterministic state sequence.



- Let the sequences provide the rewards below:
  - Bandit 1 : { 10 , 9 , 8 , 7 , 6 , 0 , 0 , 0 , … }
  - Bandit 2 : { 5 , 4 , 3 , 2 , 1 , 0 , 0 , 0 , … }

- What is the policy that maximizes $\lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T-1} a^t r_{i_t}(\xi_{i_t})\right]$?

$$10a^0 + 9a^1 + 8a^2 + 7a^3 + 6a^4 + 5a^5 + \cdots$$

Example 2

- Consider the modification below:
  - Bandit 1 : { 10 , **2** , 8 , 7 , 6 , 0 , 0 , 0 , … }
  - Bandit 2 : { 5 , 4 , 3 , **9** , 1 , 0 , 0 , 0 , … }
- What is the policy that maximizes $\lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T-1} a^t r_{i_t}(\xi_{i_t})\right]$?

"Future is not so important"

Policy 1: $10a^0 + 5a^1 + 4a^2 + 3a^3 + 9a^4 + 2a^5 + 8a^6 + \cdots \qquad (a = 0.1)$

"Future is (almost) as important as the present"

Policy 2: $10a^0 + 2a^1 + 8a^2 + 7a^3 + 6a^4 + 5a^5 + 4a^6 + \cdots \qquad (a = 0.9)$

"Future is somewhat important"

Policy 3: $10a^0 + 5a^1 + 2a^2 + 8a^3 + 7a^4 + 6a^5 + 4a^6 + \cdots \qquad (a = 0.5)$

Questions

- How to design a function $v_i(\xi_i)$ that encodes the **value of choosing bandit i** ?
  - Value: present reward + future expected rewards
  - Future reward is to be considered? When a myopic policy is optimal?
  - Future reward is the expected value of choosing bandit i forever?

    Or up until a given horizon? How to characterize this horizon?

Gittins Index

$$v_i(\xi_i) = \sup_{\tau>0} \frac{\mathbb{E}\left[\sum_{t=0}^{\tau-1} a^t \, r_i(\xi_i(t)) \mid \xi_i(0) = \xi_i \right]}{\mathbb{E}\left[\sum_{t=0}^{\tau-1} a^t \mid \xi_i(0) = \xi_i \right]}$$

where τ is the stopping-time.

- Numerator is the **discounted REWARD up to time $\boldsymbol{\tau}$**.
- Denominator is the **discounted TIME up to time $\boldsymbol{\tau}$**.
- $v_i(\xi_i)$ a maximum reward per unit time ("reward density").
- <u>Interpretation</u> from [1]: "greatest **per period rent** that one would be willing to pay for ownership of the rewards arising from the bandit as it is continued for one or more periods."
- **GITTINS INDEX POLICY** chooses the bandit with highest $v_i(\xi_i)$ at every decision time t.

Gittins Index

- Next, we prove that the Gittins Index Policy is optimal. (adapted from [4])

- This proof is instructive because:
    - shows the origin of the expression for the Gittins index;
    - provides insight into why the Gittins Index Policy is optimal;
    - provides insight into why it is NOT optimal for the **restless** case;
    - used in the Whittle Index part of this presentation.

[1] J. Gittins, K. Glazebrook and R. Weber, *Multi-armed Bandit Allocation Indices*, 2 Ed., 2011.
[4] R. Weber, On the Gittins Index for Multiarmed Bandits, 1992.

Gittins Index – Proof

- Consider a **single** bandit i with a "**playing charge**" of λ.
- Optimal Policy is a **stopping rule**.
  - if at time τ it is optimal to stop, at time $\tau + 1$ it is also optimal to stop.
- **Optimal Reward**:

$$J(\xi_i) = \max_{\pi} J_\pi(\xi_i) = \sup_{\tau > 0} \mathbb{E}\left[\sum_{t=0}^{\tau-1} a^t [r_i(\xi_i(t)) - \lambda] \middle| \xi_i(0) = \xi_i \right]$$

- **Optimal Policy**:

At every decision time, calculate $J(\xi_i)$:

Play, if $J(\xi_i) \geq 0$; Stop, otherwise.

Gittins Index – Proof

- For every ξ_i, there is a λ such that there is a null reward for playing:

$$J(\xi_i) = \sup_{\tau>0} \mathbb{E}\left[\sum_{t=0}^{\tau-1} a^t [r_i(\xi_i(t)) - \lambda] \,\middle|\, \xi_i(0) = \xi_i \right] = \mathbf{0}$$

- Notice that $J(\xi_i)$ is convex and decreasing on λ. Thus, it has a single root which is the Gittins Index, $v_i(\xi_i)$, given by:

$$v_i(\xi_i) = \sup_{\tau>0} \frac{\mathbb{E}\left[\sum_{t=0}^{\tau-1} a^t \, r_i(\xi_i(t)) \mid \xi_i(0) = \xi_i \right]}{\mathbb{E}\left[\sum_{t=0}^{\tau-1} a^t \mid \xi_i(0) = \xi_i \right]}$$

Details

- This $v_i(\xi_i)$ is called the **fair charge** during state ξ_i.
- **This is the charge that makes it equally desirable to play and to stop.**

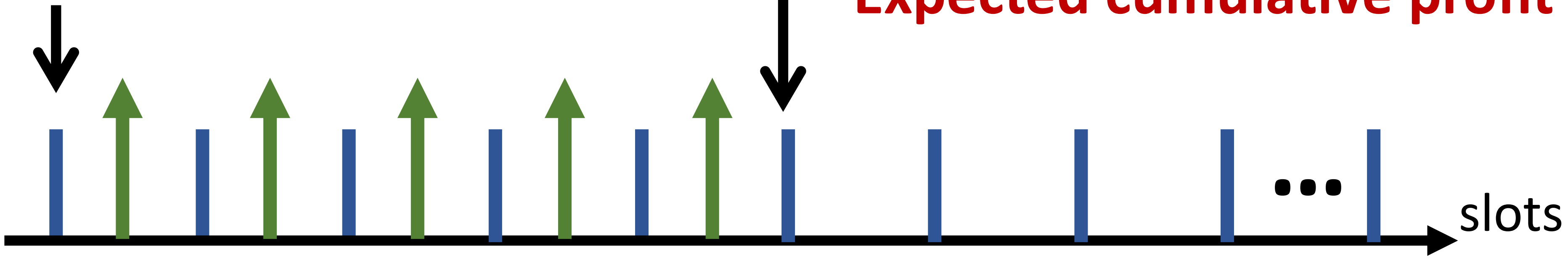
Gittins Index – Proof

- Suppose that at time $t = 0$ we are in state ξ_i with a **fair charge** of $v_i(\xi_i)$.
- If we set $\lambda = v_i(\xi_i)$ and **play bandit i optimally**, we expect 0 profit.
  - Optimal play is not profitable nor loss-making.
- If we deviate from the optimal policy, then we expect loss.
- **What is the optimal policy in this case?** (Stopping rule)

Set charge to FAIR CHARGE

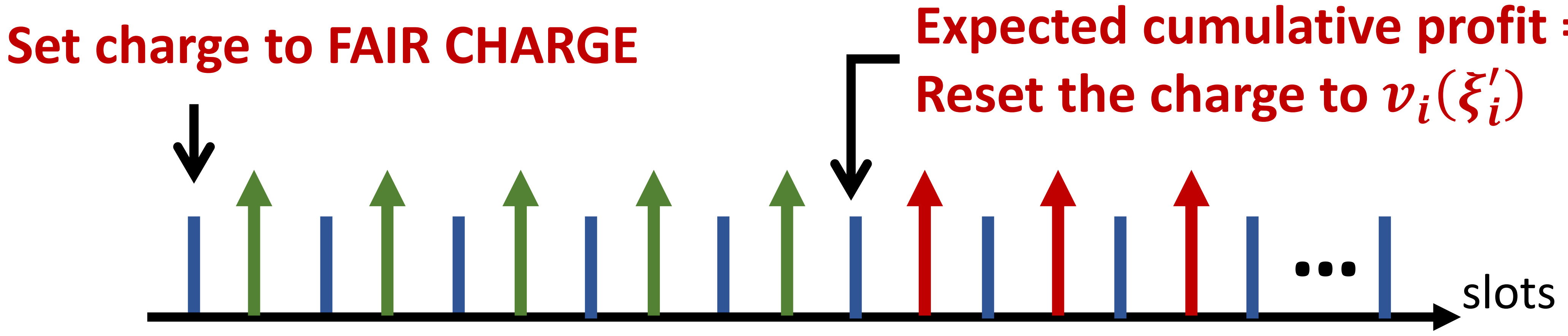
Stopping time.

Expected cumulative profit = 0



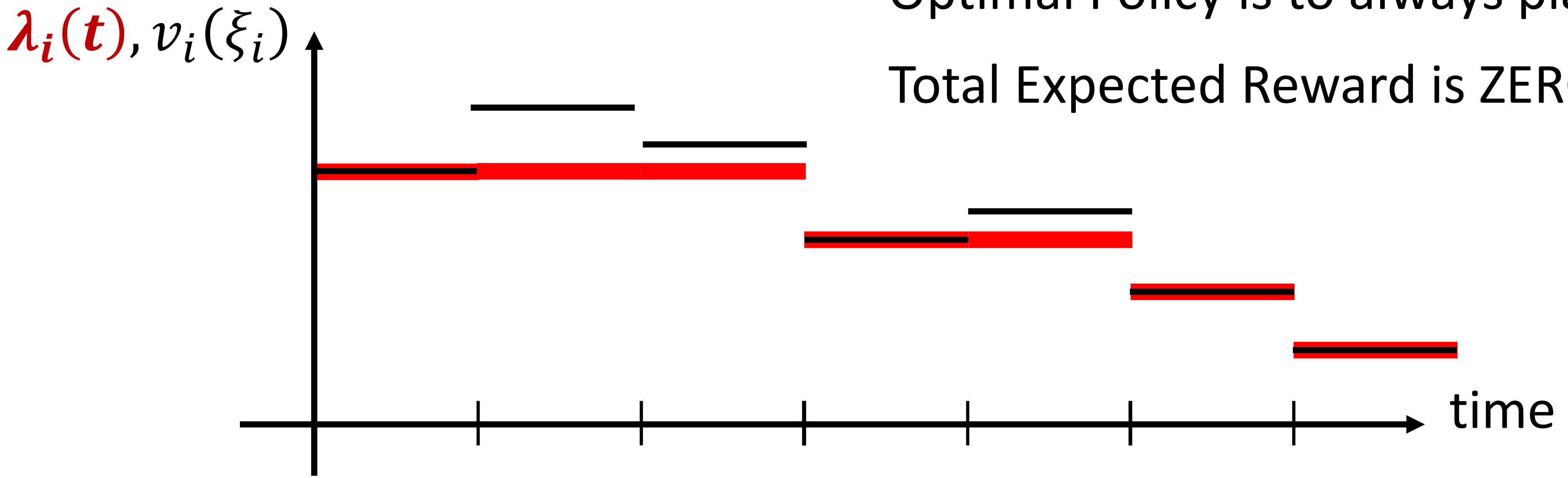
Gittins Index – Proof

- What if **at the stopping time**, we **reset the charge**.
- At the stopping time, instead of stopping, we reset the charge to $v_i(\xi_i')$ and continue playing.
- If we do this **repeatedly**, the expected profit would still be ZERO.
  - The bandit is **continuously playing a fair game** with optimum policy.



Gittins Index – Proof

- Notice that as the game evolves, the charge is reset several times.
- Let $\lambda_i(\mathrm{t})$ be the current fee and $v_i(\xi_i)$ the calculated fair fee.
- $\lambda_i(\mathrm{t})$ is non-increasing and is equal to the minimum fair charge “so far”.

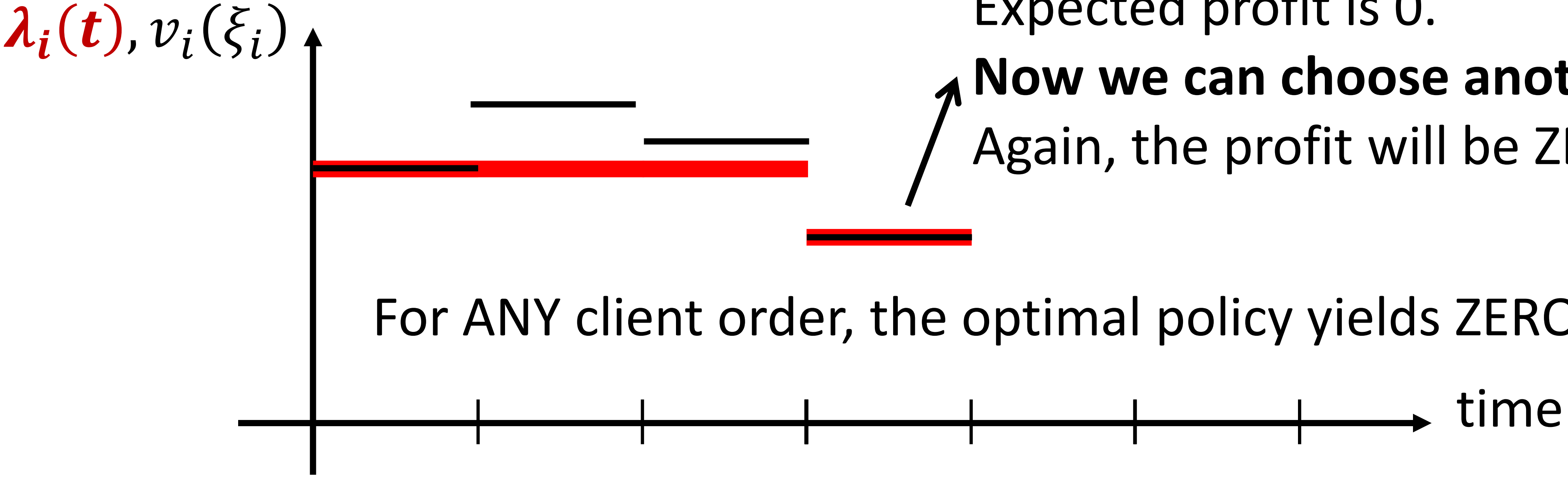


Optimal Policy is to always play.

Total Expected Reward is ZERO.

Gittins Index – Proof

- Consider **n bandits**, each with a different initial state ξ_i.
- We set **each initial charge as $\boldsymbol{\lambda_i = v_i(\xi_i)}$**, $\forall i$ and update them as before.
- Assume we selected bandit i. The optimal policy tells us to play client i until **$\boldsymbol{\lambda_i}$ is reset.** If we don't, we will incur in a loss.



Expected profit is 0.

Now we can choose another client.

Again, the profit will be ZERO.

For ANY client order, the optimal policy yields ZERO profit.

Gittins Index – Proof

- Consider the policy that selects the bandit with highest $\lambda_i(t)$ at every slot.
- This policy has NULL profit. And **incurs the HIGHEST sum of discounted charges.**
  - This is because it selects the highest charges first, in a non-increasing order. (recall Example 1 in slide 7)
  - Since Profit = Reward – Charges → This policy incurs highest Reward.
- Notice that choosing the bandit with highest $\lambda_i(t)$ is EQUIVALENT to choosing the bandit with highest $\nu_i(\xi_i)$. **Thus the Gittins Index Policy is optimal**. ■

Gittins Index – Intuition

$$v_i(\xi_i) = \sup_{\tau>0} \frac{\mathbb{E}\left[\,\sum_{t=0}^{\tau-1} a^t\, r_i(\xi_i(t)) \mid \xi_i(0) = \xi_i\,\right]}{\mathbb{E}\left[\,\sum_{t=0}^{\tau-1} a^t \mid \xi_i(0) = \xi_i\,\right]}$$

where τ is the stopping-time.

- If $\tau = 1$ for all bandits and all states, then the Gittins Policy is actually a myopic policy (a.k.a. one-step look-ahead policy)
- In general, the Gittins policy can be seen as a τ-step look-ahead policy.

- What happens when the bandits are restless? RMAB problems next.

Whittle Index

Restless Multi Armed Bandit Problem

Whittle's index

- Whittle **extends the notion of index to restless bandits.**

- Generalizations in comparison to the MAB problem:

1. At each time t, exactly **m out of n** bandits are given the action $u = 1$
   Formally, $u_i(t) \in \{0,1\}, \forall i, t \quad \text{and} \quad \sum_{i=1}^{n} u_i(t) = m, \forall t$

2. Action $u = 0$ no longer freezes the bandit. [Reward + Evolution]
   They evolve (possibly) in a distinct way than when $u = 1$.
   <u>Use cases</u>: work / rest ; high speed / low speed .

Three Optimization Problems

- **[Original].** Original Problem:

$$\textbf{maximize} \lim_{T\to\infty} \mathbb{E}[\textstyle\sum_{t=0}^{T-1} a^t \sum_{i=1}^{n} r_i(\xi_i, u_i)]$$

$$\text{s.t.} \quad \textstyle\sum_{i=1}^{n} u_i(t) = m, \forall t$$

$$u_i(t) \in \{0,1\}, \forall i$$

- **[Relaxed].** Problem with Relaxed activation constraint.

$$\textstyle\sum_{t=0}^{\infty} a^t \sum_{i=1}^{n} u_i(t) = m/(1-a)$$

- **[Lagrange]**. The Lagrange Dual Function is given by:

$$\mathcal{L}(\lambda) = \textbf{maximize} \lim_{T\to\infty} \mathbb{E}\left[\textstyle\sum_{t=0}^{T-1} a^t \sum_{i=1}^{n}\left(r_i(\xi_i, u_i) - \lambda u_i(t)\right)\right] + \lambda(m/(1-a))$$

$$\text{s.t.} \quad u_i(t) \in \{0,1\}, \forall i$$

Decoupling the [Lagrange] Problem

- **[Lagrange]**. The Lagrange Dual Function is given by:

$$\mathcal{L}(\lambda) = \textbf{maximize} \lim_{T\to\infty} \mathbb{E}\left[\sum_{i=1}^{n} \sum_{t=0}^{T-1} a^t \big(r_i(\xi_i, u_i) - \lambda u_i(t)\big)\right] + \lambda(m/(1-a))$$

$$\text{s.t. } u_i(t) \in \{0,1\}, \forall i$$

- Notice that we can decouple this problem into the bandits and neglect the last term (constant). Then, for a fixed λ and for each bandit, we have:

[Decoupled Problem]

$$\textbf{maximize} \lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T-1} a^t \left(r_i(\xi_i, u_i) - \lambda u_i(t)\right)\right]$$

$$\text{s.t. } u_i(t) \in \{0,1\}, \forall i$$

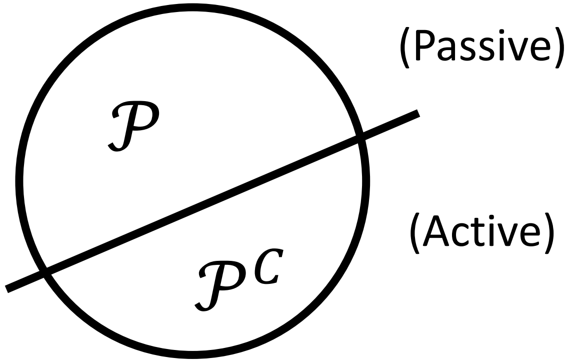
[Similar to Gittins!]

Decoupled Problem

- Main differences when compared to the MAB problem:
  - Passive bandits may give reward.
  - Passive bandits may change states.
- Thus, the optimal policy is NOT a stopping rule.

- Again, there exists at least one optimal policy which is **deterministic, stationary and Markov**. In general, this optimal policy divides the state space into two subsets:
  - Let $\mathcal{P}(\lambda)$ be the set of ALL states for which it is optimal to idle when the playing charge is λ.
  - **<u>Optimal Policy</u>**: play, if $\xi_i \in \mathcal{P}^C(\lambda)$; stop, otherwise.

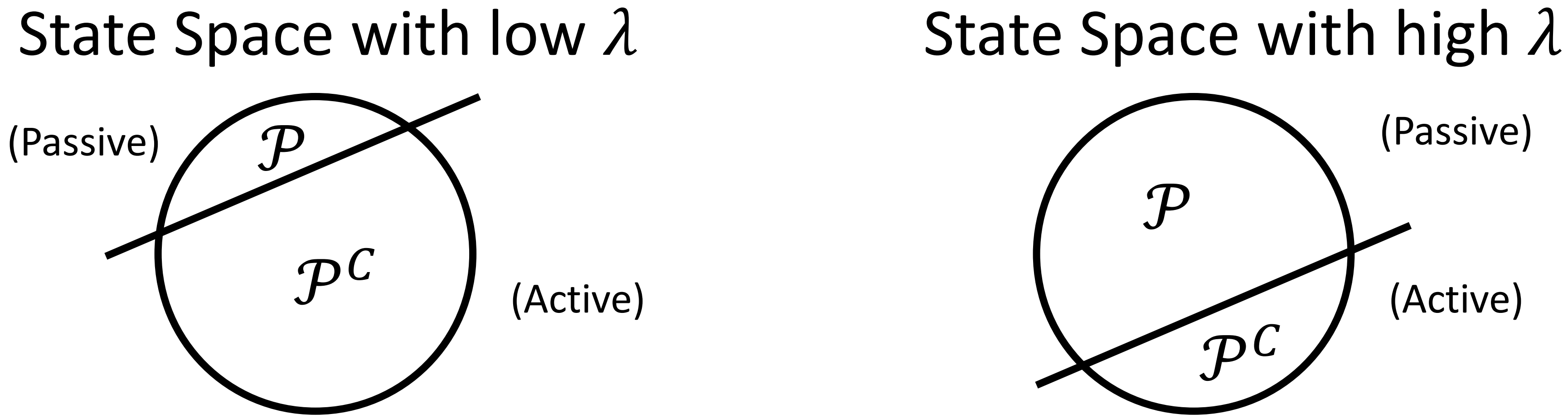
State Space with λ



Decoupled Problem – Indexability

- The set $\mathcal{P}(\lambda)$ is characterized by the solution of the Decoupled Problem.

- **Definition of Indexability**: The Decoupled Problem associated with bandit *i* is *indexable* if $\mathcal{P}(\lambda)$ increases **monotonically** from $\emptyset$ to the entire state space as λ increases from 0 to $+\infty$. The RMAB problem is *indexable* if the Decoupled Problem is *indexable* for all bandits.



- Means that if a bandit is rested with λ, it should also be rested when $\lambda' > \lambda$.

Decoupled Problem – Whittle Index

- **Definition of Index**: Consider the Decoupled Problem and denote by $v_i(\xi_i)$ the Whittle Index in state ξ_i. Given *indexability*, $v_i(\xi_i)$ is the infimum playing charge λ that makes it equally desirable to play and to stop in state ξ_i.

- Recall that this definition is the same as in the proof for Gittins. (slide 14)

- **Optimal Policy** for the [Lagrange] Problem with **n bandits and fixed λ**.
  - At every decision time, calculate the **fair charge** $v_i(\xi_i')$ for each bandit.
    - If $v_i(\xi_i') \geq \lambda$. “Current fee is **smaller** than the fair fee” → Play
    - If $v_i(\xi_i') < \lambda$. “Current fee is **higher** than the fair fee” → Stop

Whittle Index Policy

- Going back to our [Original] problem:
  - At each time t, exactly **m out of n** bandits are given the action $u = 1$
  - There is no "playing charge" λ.
- The Whittle Index Policy is one that, at every decision time, **selects the m bandits with higher values of $v_i(\xi_i')$.**
- The **Index Policy is a low-complexity heuristic** that has been extensively used in the literature and is known to have a strong performance in a range of applications.
- The **challenge** associated with this approach is that the Index Policy is only defined for problems that are *indexable*, a condition that is often difficult to establish. Moreover, it is often hard to find a closed-form expression to $v_i(\xi_i')$.
- Notice that if our RMAB problem is actually a MAB, then **Whittle ≡ Gittins**. Thus, in this case, Whittle is optimal.

Asymptotic Optimality (for average cost problems)

- **Intuition**: as $n \to \infty$, we expect a weaker coupling among different bandits.

- **Conjecture** [6]: with $m/n = \alpha$ and as $n \to \infty$, the **reward of the optimal policy** is asymptotically the same as the reward achieved by **Whittle's index policy**.

- From [5]: this **conjecture is NOT always satisfied in RMAB**. Using theory of large deviations, [5] derives sufficient conditions for the conjecture to hold. One of which is indexability.

- From [5]: "Evidence so far is that counterexamples to the conjecture are rare and that the degree of sub-optimality is very small. It appears that in most cases the index policy is a very good heuristic."

[5] R. Weber and Weiss, "On an Index Policy for Restless Bandits", 1990
[6] P. Whittle, "Restless Bandits: Activity Allocation in a Changing World", 1981

References

[1] J. Gittins, K. Glazebrook and R. Weber, *Multi-armed Bandit Allocation Indices*, 2 Ed., 2011.

[2] R. Weber, *Tutorial on Bandit Processes and Index Policies*, YEQT VII workshop, 2013.

[3] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, 2008.

[4] R. Weber, On the Gittins Index for Multiarmed Bandits, 1992.

[5] R. Weber and Weiss, "On an Index Policy for Restless Bandits", 1990

[6] P. Whittle, "Restless Bandits: Activity Allocation in a Changing World", 1981

Supplementary Slides

Bandit Process

- Bandit process is a special type of semi-Markov decision process.
- Continuous time and a succession of (random) decision times $t_1, t_2, t_3, \ldots$
- Same controls applied at decision times
  - $u(t_i) = 0$ freezes the process and gives no reward.
    Time $t_i + \delta$ is another decision time.
  - $u(t_i) = 1$ continues the process and gives instantaneous reward $a^{t_i} r\big(x(t_i)\big)$.
    Time $t_i + s$ is another decision time, where s is drawn from $F(s|y, x)$.

  where $x(t)$ is the current state, y is the next state, $a \in (0,1)$ is the discount factor and r(.) is the positive (and bounded) reward .
- State Transitions are instantaneous with $P(y|x)$.
- Markov bandit process is a Bandit Process with discrete decision times t={0,1,…}

Decision Process Theory [3]

- Let D be a Markov decision process with state space $\vec{E}$ and control space U.
- Objective is to maximize the reward of the expected sum of discounted rewards up to the infinite horizon, i.e. to maximize:

$$J_\pi(\vec{\xi}\,) = \lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T-1} a^t r_{i_t}\big(\xi_{i_t}(t)\big)\middle|\, \vec{\xi}(0) = \vec{\xi}\,\right]$$

- Let $r_i(.)$ be bounded and U($\vec{\xi}$) be the FINITE set of controls for each $\vec{\xi} \in \vec{E}$.
- Theorem: there is at least one optimal policy which is **deterministic, stationary and Markov**.

[3] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, 2008.

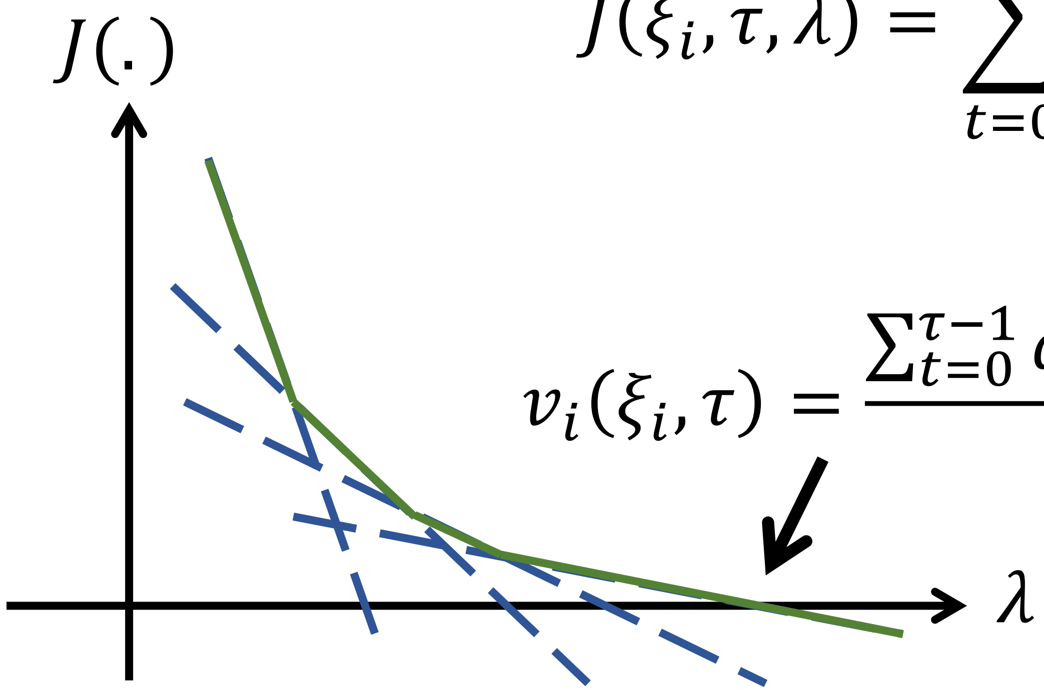
- Equation:

$$J(\xi_i) = \sup_{\tau>0} \mathbb{E}\left[\sum_{t=0}^{\tau-1} a^t [r_i(\xi_i(t)) - \lambda] \,\middle|\, \xi_i(0) = \xi_i \right] = 0$$

- For a fixed ξ_i and τ , the function $J(\xi_i, \tau, \lambda)$ is linear and decreasing on λ.

$$J(\xi_i, \tau, \lambda) = \sum_{t=0}^{\tau-1} a^t \mathbb{E}[r_i(\xi_i(t))] - \lambda \sum_{t=0}^{\tau-1} a^t$$

(Dashed blue lines for each τ)



$$v_i(\xi_i, \tau) = \frac{\sum_{t=0}^{\tau-1} a^t \mathbb{E}[r_i(\xi_i(t))]}{\sum_{t=0}^{\tau-1} a^t}$$

The Gittins Index is the highest $v_i(\xi_i, \tau)$

Necessary Conditions for Gittins

- Infinite Horizon
- Constant exponential discounting
- Single processor/server