Whittle Index Policy for Crawling Ephemeral Content

K. Avrachenkov (Inria) and V. Borkar (IIT Bombay)

MAS 2016, Grenoble

Search engine building blocks

The architecture of most web search engines consist of the following main building blocks:

- Crawler or Web Robot;
- Indexing engine;
- Interface handling users' queries.

Search engine building blocks

Crawler is responsible for discovering new web pages and updating old pages.

The document collection should be quite fresh and satisfy current interests of users (Cho and Garcia-Molina, 2000, 2003).

Ephemeral content

Nowadays an overwhelming majority of people find new information on the web at news sites, blogs, forums and social networking groups.

However, most information consumed is *ephemeral* in nature. That is, people tend to lose their interest in the content in several days or hours.

The interest in a content can be measured in terms of clicks or number of relevant search requests.

Ephemeral content

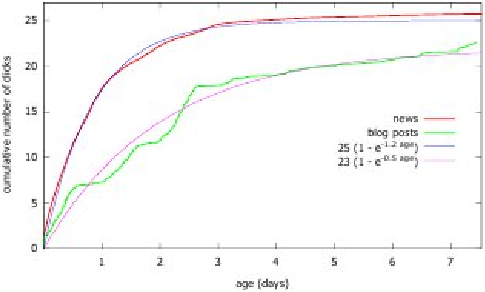
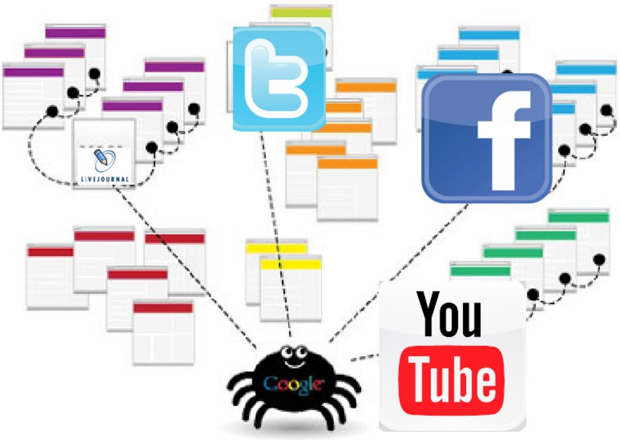


Figure: from (Lefortier, 2013).

Ephemeral content



Model

There are N sources of ephemeral content.

A content at source $i \in \{1, ..., N\}$ is published with an initial utility modelled by a nonnegative random variable ξ_i, with mean $\bar{\xi}_i$, and decreasing exponentially over time with a deterministic rate μ_i.

Thus, if source i's content is crawled τ time units after its creation, its utility is given by $\xi_i \exp(-\mu_i \tau)$.

The new content arrives at source i according to a time-homogeneous Poisson process with rate Λ_i.

Model

We assume that the crawler crawls periodically at multiples of time $T > 0$ and has to choose at each such instant which sources to crawl, subject to a constraint on the number of sources per period.

When the crawler crawls a content source, we assume that the crawling is done in an exhaustive manner.

The crawler obtains the following expected reward from crawling source i:

$$u_i = \Lambda_i E[\xi_i \exp(-\mu_i \tau)] = \frac{\Lambda_i \bar{\xi}_i}{\mu_i} (1 - \exp(-\mu_i T)) . \quad (1)$$

Set $\alpha_i = \exp(-\mu_i T)$.

Model

Let us define the state of source i at time t as the total expected utility of its content, denoted by $X_i(t)$.

Then, if we do not crawl source i at epoch t ($v_i(t) = 0$), we obtain zero reward $r_i(X_i(t), v_i(t)) = 0$ and the state evolves as follows:

$$X_i(t+1) = \alpha_i X_i(t) + u_i. \tag{2}$$

On the other hand, if we crawl source i ($v_i(t) = 1$), we obtain the expected reward $r_i(X_i(t), v_i(t)) = X_i(t)$ and the next state of the source is given by

$$X_i(t+1) = u_i. \tag{3}$$

Model

Our aim is to maximize the long run average reward

$$\limsup_{t\uparrow\infty} \sum_{i=1}^{N} \frac{1}{t} \sum_{m=0}^{t} r(X_i(t), v_i(t)) \tag{4}$$

subject to the constraint

$$\sum_{i=1}^{N} C_i v_i(t) = M, \tag{5}$$

for a prescribed $M > 0$.

Whittle index

With large M the model quickly becomes intractable even numerically, so-called "curse of dimensionality".

Fortunately, there is a concept of Whittle index (Whittle, 1988) which helps to decompose multi-dimensional problems.

Examples of applications:

- sensor scheduling (Nino-Mora, Vilar)
- multi-UAV coordination (Ny, Dahleh, Feron)
- congestion control (Avrachenkov, Ayesta, Doncel, Jacko)
- cognitive radio (Liu, Zhao)
- real time wireless multicast (Raghunathan, Borkar, Cao, Kumar)

Whittle index

The first idea is to substitute the strict constraint (5) with the average constraint

$$\limsup_{t\uparrow\infty} \sum_{i=1}^{N} \frac{1}{t} \sum_{m=0}^{t} C_i v_i(t) = M \tag{6}$$

By the way, this becomes an instance of Average Reward Markov Decision Process with Constraints (Piunovskiy 1997, Altman, 1999).

Whittle index

Then, we can use the technique of Lagrange multiplier

$$\limsup_{t\uparrow\infty} \frac{1}{t} \sum_{s=0}^{t} E[r_i(X_i(t))v_i(s) + \lambda(v_i(s) - M/N)]. \tag{7}$$

with the associated dynamic programming equation for the above average reward problem

$$V_i(x)+\beta = \max\left(r_i(x)+\int p_i(dy|x)V_i(y), \lambda+\int q_i(dy|x)V_i(y)\right). \tag{8}$$

Whittle index

In our particular case, the dynamic programming equation takes the form:

$$V(x) + \beta = \max\left(C\lambda + V(\alpha x + u),\ x\right) \quad (9)$$

$$= \max_{v\in\{0,1\}} \Big(vx + (1-v)(C\lambda + V(\alpha x + u))\Big) \quad (10)$$

Whittle index

The problem is called *indexable* if the set of passive states

$$B(\lambda) := \left\{ x : \lambda + \int q_i(dy|x)V_i(y) \geq r_i(x) + \int p_i(dy|x)V_i(y) \right\}.$$

increases monotonically from ϕ to the full state space as λ increases from $-\infty$ to ∞.
Then, for each source i, the *Whittle index* is defined as

$$\gamma_i(x_i) :=$$

$$\{\lambda' : \lambda' + \int q_i(dy|x_i)V(y) = r_i(x_i) + \int p_i(dy|x_i)V(y)\}.$$

Whittle index

The *Whittle index policy* is to set $v_i(t) = 1$ for the i with the top M indices and $v_j(t) = 0$ for the rest.

Under quite general conditions, the Whittle index policy has been shown asymptotically optimal:

- by (Weber & Weiss, 1990) for symmetric case;
- by (Verloop, 2015) for finite number of types of bandits.

Whittle index

Our main result is

Theorem The problem of crawling ephemeral content sources is indexable with the Whittle index given by

$$\gamma_i(x_i) = \frac{1}{C_i}\left(\eta_i(x_i)((1-\alpha_i)x_i - u_i) + x_i\right),$$

where

$$\eta_i(x_i) = \left\lceil \log^{+}_{\alpha_i}\left(\frac{u_i - (1-\alpha_i)x_i}{u_i}\right)\right\rceil,$$

and

$$u_i = \frac{\Lambda_i \bar{\xi}_i}{\mu_i}\left(1 - \exp(-\mu_i T)\right).$$

Whittle index

Proof key points:

- Proving key properties of the value function first for the discount case and then passing to the limit;
- The sets B and B^c are of the form $[u, a)$ and $[a, u^*]$, resp., for some $a \in [u, u^*]$;
- The optimal policy for separate arm is of threshold type;
- The value of the threshold monotonically increase with λ.

Numerical example

Consider an illustrative example with four information sources.

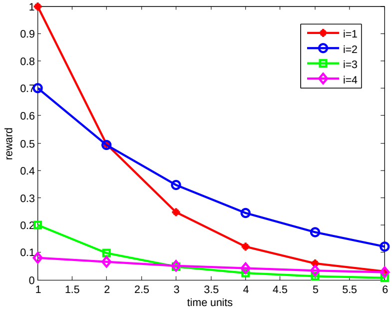


Figure: Content value as a function of time.

Numerical example

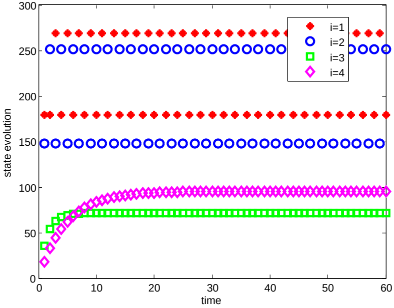


Figure: The case of $M = 1$.

Note that if one crawls only the "best" source 1, he obtains the average reward 179.79. In contrast, the index policy involving two sources results in average reward 254.66.

Numerical example

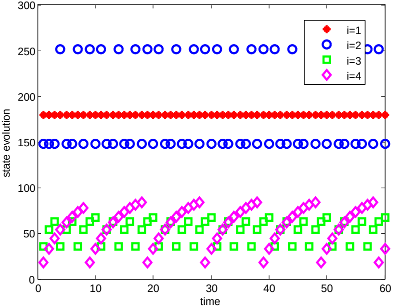


Figure: The case of $M = 2$.

It is interesting that now the policy becomes much less regular.

Numerical example (stochastic setting)

When we observe the states in the stochastic setting, we can still apply the deterministic Whittle index.

Even though in the stochastic case the deterministic Whittle index is just an heuristic, it performs quite well. Take $M = 1$.

Round Robin policy: 208

Greedy policy, $\max_i \frac{\Lambda_i \bar{\xi}_i}{\mu_i}(1 - \exp(-\mu_i \times TLastCrawl_i))$: 260

Deterministic Whittle index: 285

Numerical example (stochastic setting)

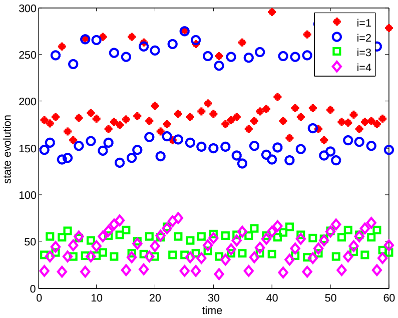


Figure: The case of $M = 2$ (stochastic model).

In the stochastic setting source 1 is crawled from time to time.

Other contributions

Some other developments not described in the talk:

- We proved Whittle indexability in the fully stochastic case;
- Described numerical procedures for the Whittle index in the stochastic case (unfortunately, there is no nice explicit expression);
- Online dual descent type method for the relaxed control problem.

Thank you!

Any questions are welcome.