TD 10 : Algorithmes d'approximation, en ligne et probabiliste

1 Gestion de mémoires caches : Least Recently Used (LRU)

Consider a cache of size k that can hold up to k pages. We are given a sequence of page requests $\sigma = (p_1, \ldots, p_n)$ where each p_i is a request to a page.

> Least Recently Used (LRU) algorithm :
> — If a requested page is already in the cache (a cache hit), no eviction occurs.
> — If the requested page is not in the cache (a cache miss), the algorithm evicts the page for which the longest time has passed since it was last requested and inserts the requested page into the cache.

Assume that the cost of serving a request is 0 if it is a cache hit and 1 if it is a cache miss. What is the competitive ratio of the LRU caching algorithm ?

Solution. Let $\sigma = \{p_1, \ldots, p_n\}$ be the online sequence of page requests. Let $\mathsf{OPT}(\sigma)$ be the least number of page faults when σ is known ahead of time.

Given a sequence σ we decompose it into phases. Each phase is a maximal sequence consisting of requests to at most k different pages. Let P_σ be the number of phases in σ.

We first prove that $\mathsf{LRU}(\sigma) \leq k \cdot P_\sigma$. To this end, we show that in each phase, LRUhas at most k cache misses. Suppose we have two cache misses in some phase that are due to the same page request p. Then there is a point where p is evicted in this phase. Then at this point, all pages in the cache can only belong to the current phase. Indeed, if there is some page q in the cache that was requested in some previous phase but not in the current one, then q would have been the least recently used page and should have been evicted instead of p.

Since there are at most k different elements in the current phase, either the cache already contained p at the time of its eviction or had an empty spot meaning p did not need to be evicted. Thus

Lemma 1.1. $\mathsf{LRU}(\sigma) \leq k \cdot P_\sigma$.

We now show that $\mathsf{OPT}(\sigma) \geq P_\sigma - 1$. Let the P_σ phases be $\Pi_1, \ldots, \Pi_{P_\sigma}$. Consider the shifted phases $\Pi'_1, \ldots, \Pi'_{P_\sigma}$ where Π'_i is obtained by moving the first element of P_{i+1} to P_i ; shifted phase Π'_1 retains the first element of Π_1. We claim that OPThas at least one cache miss on every shifted

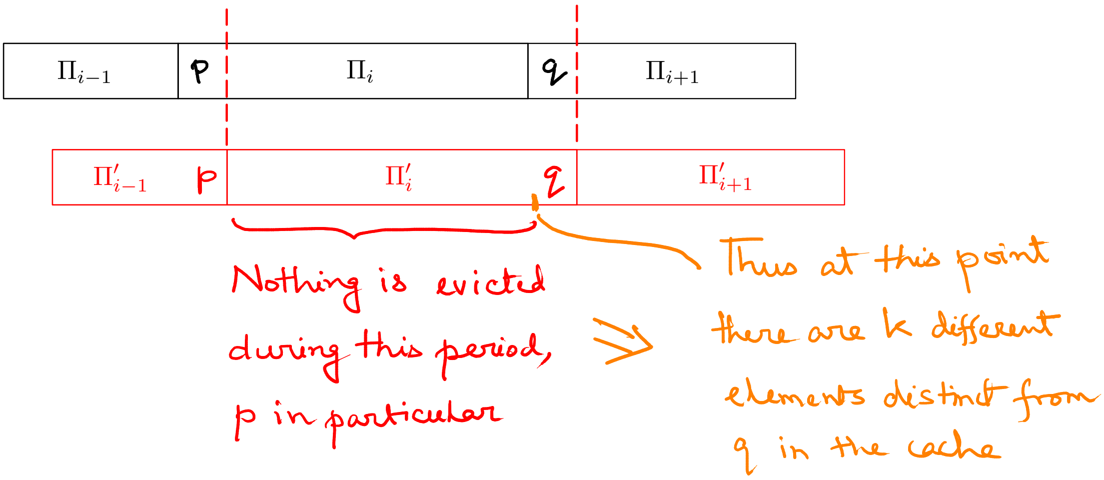

FIGURE 1 – Proving the lower bound for OPT

phase except possibly the last. Suppose on the contrary that OPT does not evict any pages during Π'_i. Suppose Π_i ends at position t. At this point of running OPT, all different pages of Π_i are in the cache (as there are no evictions during Π'_i by OPT). Let q be the element that occurs at position $t+1$; note that this is included in Π'_i. As q is different from all elements in Π_i, OPTmust evict some page from the cache when it reaches its position $t+1$. Thus OPTevicts some page during phase Π'_i, contradiction. Thus

Lemma 1.2. $\mathsf{OPT}(\sigma) \geq P_\sigma - 1$.

Combining the two lemmas gives

$$\mathsf{LRU}(\sigma) \leq k \cdot \mathsf{OPT}(\sigma) + k.$$

□

2 Problème de la location de skis (Ski Rental Problem)

You are going skiing in the mountains and you want to keep skiing until the day you crash badly. Renting a ski involves paying 1 unit of currency at the start of each day, and buying it involves paying $B > 1 \in \mathbb{N}$ units at the start of the day. Suppose you ski for x (entire) days, but x is not known in advance. What is the best strategy for buying (or not buying) the skis ?

Solution. One needs to be precise about what determines the best strategy. Here we are interested in finding an ALG with minimal competitive ratio, i.e., we are trying to minimize $\sup_{x\geq 0} \frac{ALG(x)}{OPT(x)}$.

Note that $\text{OPT}(x) = \min(x, B)$. Let ALG_i be the cost of the trip if one buys the skis at the start of the i-th day. It is not hard to see that

$$\text{ALG}_i(x) = \begin{cases} x, & \text{if } x < i \\ i - 1 + B & \text{otherwise.} \end{cases}$$

We claim that ALG_B is the optimal strategy. The competitive ratio of ALG_B is $\frac{2B-1}{B} = 2 - \frac{1}{B}$, which corresponds to the worst case scenario of crashing on the day of buying the ski.

If $i < B$, then the competitive ratio of ALG_i is at least $\frac{i-1+B}{i} = 1 + (B-1)/i \geq 2$ which corresponds to the worst case of crashing on the day of buying the ski. If $i > B$, then the competitive ratio of ALG_i is at least $\frac{i-1+B}{B} > 2 - \frac{1}{B}$. □

3 MAX-SAT

The Maximum Satisfiability problem is defined as follows.

Problem. Given a conjunctive normal form $f = C_1 \wedge \cdots \wedge C_m$ formula on Boolean variables $x_1, \ldots, x_n$, find a truth assignment to the Boolean variables that maximizes the total number of satisfied clauses.

1. Give a randomized $(1 - 1/e)$-approximation algorithm $\mathcal{A}_1$ for the MAX-SAT problem using the randomized rounding technique.
2. Give a simple randomized $1/2$-approximation algorithm $\mathcal{A}_2$.
3. Show how to derandomize both of the algorithms above.
4. Explain why $\mathcal{A}_1$ is good for dealing with small clauses while $\mathcal{A}_2$ is good for dealing with with large clauses. Show how one can combine $\mathcal{A}_1$ and $\mathcal{A}_2$ to obtain a randomized $3/4$-approximation algorithm.
5. Give a deterministic $3/4$-approximation algorithm for MAX-SAT.

Solution to Problem 1. For each Boolean variable x_i, we associate an LP variable y_i and for each clause C_j, we associate an LP variable z_j. For each clause C_j, let P_j and N_j be the set of indices of variables that are non-negated and negated in C_j respectively. Then the following is an integral LP formulation of the MAX-SAT problem :

Maximize

$$\sum_{1\leq j\leq m} z_j$$

subject to

$$\forall C_j : \quad \sum_{i\in P_j} y_i + \sum_{i\in N_j} (1-y_i) \geq z_j.$$

$$\forall j : \ y_i \in \{0,1\}, \quad \forall j : \ z_j \in \{0,1\}$$

The main condition says that if a clause C_j is true, then at least one of its literals evaluates to true. The LP relaxation of this is :

Maximize

$$\sum_{1\leq j\leq m} z_j$$

subject to

$$\forall C_j : \quad \sum_{i\in P_j} y_i + \sum_{i\in N_j} (1-y_i) \geq z_j.$$

$$\forall\, i : 0 \leq y_i \leq 1, \quad \forall\, j : 0 \leq z_j \leq 1.$$

Let $(\boldsymbol{y}^*, \boldsymbol{z}^*)$ be an optimal solution to this LP.

Randomized rounding :
For $1 \leq i \leq n$, choose variable x_i to be 1 with probability y_i^*.

If ℓ_j is the number of literals in C_j then

$$\begin{aligned}\mathbb{P}(\text{clause } C_j \text{ is not true}) &= \prod_{i\in P_j}(1-y_i^*)\prod_{i\in N_j} y_i^* \\ &\leq \left[\frac{1}{\ell_j}\left(\sum_{i\in P_j}(1-y_i^*) + \sum_{i\in N_j} y_i^*\right)\right]^{\ell_j} \qquad \text{(By AM-GM inequality)} \\ &= \left[1-\frac{1}{\ell_j}\left(\sum_{i\in P_j} y_i^* + \sum_{i\in N_j}(1-y_i^*)\right)\right]^{\ell_j} \\ &\leq \left(1-\frac{z_j^*}{\ell_j}\right)^{\ell_j},\end{aligned}$$

where the first inequality follows from the AM-GM inequality and the second inequality follows from the main LP condition $\sum_{i\in P_j}(1-y_i^*) + \sum_{i\in N_j} y_i^* \geq z_j^*$. From this we get that

$$\mathbb{P}(\text{clause } C_j \text{ is true}) \geq 1 - \left(1-\frac{z_j^*}{\ell_j}\right)^{\ell_j}.$$

For fixed $k \geq 1$, consider the function $f : [0,1] \to \mathbb{R}$ defined as :

$$f(x) = 1 - \left(1-\frac{x}{k}\right)^k.$$

Since $f''(x) = -\frac{k-1}{k}\left(1-\frac{x}{k}\right)^{k-2} \leq 0$ for $0 \leq x \leq 1$, the function f is concave on $[0,1]$. Hence $f(\lambda \cdot 1 + (1-\lambda)\cdot 0) \geq \lambda f(1) + (1-\lambda) f(0)$, and since $f(0)=0$, we have that $f(\lambda) \geq \lambda f(1)$. Using this, we get that

$$\mathbb{P}(\text{clause } C_j \text{ is true}) \geq z_j^* \left(1 - \left(1 - \frac{1}{\ell_j}\right)^{\ell_j}\right).$$

Using the fact that $1+x \leq e^x$ for all real x and $\sum_j z_j^* \geq \mathrm{OPT}$ where OPT is the optimal solution to MAX-SAT, we get the expected number of satisfied clauses to be

$$\mathbb{E}[\text{number of true } C_j] \geq \mathrm{OPT}\left(1 - \frac{1}{e}\right).$$

Hence randomized rounding gives a $(1-1/e)$-approximation algorithm in expectation. □

Solution to Problem 2. We set each x_i to be true with probability $1/2$. It can easily be shown using linearity of expectation that the expected number of satisfied clauses is at least $m/2 \geq \mathrm{OPT}/2$ where OPT is the optimal solution to MAX-SAT. □

Solution to Problem 3. Not difficult, use the same style of arguments applied to the problem of finding a cut of size $m/2$ in the previous TD. □

Solution to Problem 4. Consider a clause C_j of length ℓ_j. Algorithm $\mathcal{A}_1$ satisfies the clause with probability at least $\left[1 - \left(1 - \frac{1}{\ell_j}\right)^{\ell_j}\right] z_j^*$, while $\mathcal{A}_j$ satisfies the clause with probability $1 - 2^{-\ell_j}$. Thus when the clause is short, it is more likely to be solved by $\mathcal{A}_1$ and not by $\mathcal{A}_2$. If the clause is long, it is more likely to be solved by $\mathcal{A}_2$ and not by $\mathcal{A}_1$.

We can obtain a randomized 3/4-approximation algorithm by running both algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ and choosing the better solution. Observe that

$$\begin{aligned}
\mathbb{E}[\max(\mathcal{A}_1, \mathcal{A}_2)] &= \frac{1}{2}\mathbb{E}[\mathcal{A}_1] + \frac{1}{2}\mathbb{E}[\mathcal{A}_2] \\
&\geq \frac{1}{2}\sum_{j=1}^{m} \left[1 - \left(1 - \frac{1}{\ell_j}\right)^{\ell_j}\right] z_j^* + \frac{1}{2}\sum_{j=1}^{m}\left(1 - 2^{-\ell_j}\right) \\
&\geq z_j^* \left[\frac{1}{2}\sum_{j=1}^{m} \left[1 - \left(1 - \frac{1}{\ell_j}\right)^{\ell_j}\right] + \frac{1}{2}\sum_{j=1}^{m}\left(1 - 2^{-\ell_j}\right)\right].
\end{aligned}$$

It remains to show for all $\ell \geq 1$ that

$$\frac{1}{2}\sum_{j=1}^{m} \left[1 - \left(1 - \frac{1}{\ell}\right)^{\ell}\right] + \frac{1}{2}\sum_{j=1}^{m}\left(1 - 2^{-\ell}\right) \geq \frac{3}{4}.$$

This is not hard to show. □