Online Lower Bounds via Duality

Yossi Azar, Ian Reuven Cohen, Alan Roytman

Outline

- Background & Intuition
- Multidimensional Vector Bin Packing
- Online Capital Investment

Background & Intuition

- Competitive ratio is the benchmark to measure the performance of an online algorithm
  - For any input sequence σ
  - $ALG(\sigma) \leq c \cdot OPT(\sigma) + a$
  - Then ALG is c-competitive

- Need to know if an online algorithm is “good enough”
- Try to prove there is a lower bound for all online algorithms’ competitive ratio.

Cont’d

- Find a linear programming problem where the variables represent online problem’s solution and the objective function is the competitive ratio of that solution.
- The optimal solution would be a lower bound for any online algorithms’ competitive ratios.
- Solving such problem is hard
- Duality theory
  - Dual solutions are primal solutions’ lower bounds

Outline

- Construct some parameterized collections of input sequences.
- Encode constraints that any feasible algorithm must obey.
- Select proper objective function such that optimizing it is equivalent to minimizing the competitive ratio.

- Derive parameterized dual linear program.
- Find feasible solutions to get a valid lower bound.

Multidimensional Vector Bin Packing

- Vectors $\{v_1, \ldots, v_n\}$ arrive in an online manner
  - $v_i = (v_i(1), \ldots, v_i(d)) \in [0,1]^d$
- Assign incoming vectors into bins whose capacities for any coordinate is 1.
- $\sum_{i \in B_j} v_i(k) \leq 1$ for all Bin B_j and all coordinates k
- Here we consider the relaxed version that all vectors are "splittable".
- $v \rightarrow \alpha_1 v, \alpha_2 v, \cdots, \alpha_n v$ where $\sum a_i = 1$

Cont’d

- Input Sequences:
- d phases, each phase contains the same type of vector
  - A of $v_1 = (1,0,0,\cdots,0)$
  - $2A$ of $v_2 = \left(\frac{1}{2},1,0,\cdots,0\right)$
  - …
  - dA of $v_d = \left(\frac{1}{d},\frac{1}{d},\frac{1}{d},\cdots,\frac{1}{d},1\right)$
- Offline optimal at the end of phase j is $j \cdot A$
- Any c-competitive algorithm must open at most a total amount of $c \cdot j \cdot A$ fractional bins.

Cont'd

- $x_{i,j}$ be the total fraction of vectors of type v_i that the online algorithm assigns to bins that are opened in phase $j (i \geq j)$.
- $\sum_{r=j}^{d} x_{r,j} \cdot v_r(k) \cdot A \leq c \cdot A$
- In phase j, the opened bins can store the vectors coming within and after phase j.

- $\sum_{r=1}^{i} x_{i,r} = i$
- Vectors coming in at phase i should be fully assigned.

- The objective function is c

Cont'd

- Ignored some constraints then we get dual program.

- normalizing all variables by the term $\sum_k \sum_j z_{k,j}$, competitive ratio becomes $\frac{\sum_r y_r}{\sum_k \sum_j z_{k,j}}$

The dual linear program:

$$\max \sum_{r=1}^{d} y_r$$

$$\text{s.t.:} \sum_{k=1}^{d} \sum_{j=1}^{d} z_{k,j} \leq 1$$

$$y_i \leq i \cdot z_{i,j} + \sum_{r=j}^{i-1} z_{r,j}$$

$$z_{k,j} \geq 0,$$

Cont’d

- Try $y_i = \frac{1}{i}$,
- Consider a differentiable function $f_j(x)$, where $f_j{}'(k)$ approximately represents $z_{k,j}$

$$i \cdot f_j'(i) + \int_j^i f_j'(x)dx \geq \frac{1}{i} \iff$$

$$x \cdot f_j'(x) + f_j(x) - f_j(j) \geq \frac{1}{x}.$$

- By solving the differential equation,

$$f_j(x) = \frac{\ln(x/j)}{x}, \quad f_j'(x) = \frac{1 - \ln(x/j)}{x^2}.$$

- Feasible dual variables assignment

$$y_i = \frac{1}{i}, \quad z_{k,j} = \begin{cases} \frac{1-\ln(k/j)}{k^2} & \text{if } j \leq k \leq \lfloor e \cdot j \rfloor, \\ 0 & \text{otherwise.} \end{cases}$$

Cont’d

- Verify if this solution is feasible

- Then the competitive ratio has a lower bound:

$$\frac{\sum_{i=1}^{d} y_i}{\sum_{k=1}^{d}\sum_{j=1}^{d} z_{k,j}} \geq \frac{H(d)}{\frac{H(d)}{e} + \sum_{j} \frac{1}{j^2}} \to e,$$

- In their literature review, a $(1+\varepsilon)e$-competitive algorithm exists for original problem when all vector’s coordinates are at most $O\left(\frac{\varepsilon^2}{\log d}\right)$
- This lower bound indicates that algorithm is almost optimal

Online Capital Investment

- Produce many units of commodities at minimum cost, where orders for units arrive online.
- Each machine m_i has a capital cost c_i and production cost p_i.
  - Algorithm choose to buy a machine for cost c_i
  - Algorithm choose to use machine produce units for p_i per unit.
- Here we consider the relaxed version that the machine can be bought partially.

Cont'd

- Problem setting:
- n machines, ith one have capital cost of $i + 1$ and a production cost of 2^{-i^2}

- Input Sequence:
  - n phases.
  - In phase k, $2^{k^2} - 2^{(\mathrm{k}-1)^2}$ orders are introduced
- Offline Optimal:
  - For input with only k phases, buy kth machine and produce units.
  - OPT = k+2

Cont'd

- Variables:
  - $x_{k,i}$ the fraction bought of the ith machine in the kth phase.
  - $q_{k,i}$ the fraction of products produced by the ith machine in the kth phase.

$$
\begin{aligned}
&\min\ c \\
&\text{s.t.:}\ \sum_{r=1}^{k} x_{r,i} \geq q_{k,i} && \text{cons. } y_{k,i} \\
&\sum_{i=1}^{n} q_{k,i} = 1 && \text{cons. } w_k \\
&\sum_{r=1}^{k}\sum_{i=1}^{n}(i+1)x_{r,i} + \sum_{i=1}^{n} 2^{k^2-i^2} q_{k,i} \\
&\qquad\qquad \leq c(k+2) && \text{cons. } z_k \\
&c, x_{k,i}, q_{k,i} \geq 0, && ,
\end{aligned}
$$

Cont'd

- Normalizing variables by the term $\sum_k (k+2) z_k$, competitive ratio becomes $\frac{\sum_k w_k}{\sum_k (k+2) z_k}$
- Try let $y_{k,i} = \begin{cases} w_k, k \leq i \\ 0, k > i \end{cases}$

The dual linear program:

$$\max \sum_{k=1}^{n} w_k$$

$$\text{s.t.:} \quad \sum_{k=1}^{n} (k+2) \cdot z_k \leq 1 \qquad \text{cons. } c$$

$$(i+1) \sum_{r=k}^{n} z_r \geq \sum_{r=k}^{n} y_{r,i} \qquad \text{cons. } x_{k,i}$$

$$y_{k,i} \geq w_k - z_k \cdot 2^{k^2 - i^2} \qquad \text{cons. } q_{k,i}$$

$$y_{k,i}, z_k \geq 0,$$

Cont'd

- By substituting y, we get

$$(i+1)\sum_{r=k}^{n} z_r \geq \sum_{r=k}^{i} w_r \qquad \text{cons. } x_{k,i}.$$

- Empirically take $w_r = \frac{1}{r}$, consider differentiable functions $f(x), g(x)$, where $f'(k), g'(i)$ approximately represents z_k, w_i. Then $g'(x) = \frac{1}{x}$

$$(i+1)\int_k^{n+1} f'(x)dx \geq \int_k^{i+1} g'(x)dx \iff$$

$$f(n+1) - f(k) \geq \frac{\ln(\frac{i+1}{k})}{i+1},$$

which holds for

$$f(x) = -\frac{1}{e \cdot x}, \quad f'(x) = \frac{1}{e \cdot x^2},$$

Cont'd

- Verify if this solution is feasible

- Then the competitive ratio has a lower bound:

$$\frac{\sum_{k=1}^{n} w_k}{\sum_{k=1}^{n} (k+2) z_k} = \frac{e \cdot \ln(n \cdot \epsilon) \cdot (1-\epsilon)}{\sum_{k=1}^{n} \frac{k+2}{k(k+1)}} \geq$$

$$\frac{e \cdot \ln(n \cdot \epsilon) \cdot (1-\epsilon)}{H(n) + C_1} \rightarrow \quad e \cdot (1-\epsilon),$$

- In their literature review, a e-competitive algorithm exists.
- This lower bound indicates that algorithm is almost optimal

Some drawbacks

- Input sequences can be difficult to construct.
- Need to formulate problems to LPs with competitive ratios as objective functions.
- Only obtained lower bound on the competitive ratios, may need to design algorithms to achieve such competitive ratios.

Lessons learned

- Due to the online settings, algorithms have no information about future. They need to balance all possible future inputs.
- A systematic method to obtain online deterministic and randomized lower bounds on the competitive ratio.
- This method can also help designing online algorithms.