# 1 Outline

In this lecture, we cover

- the online resource allocation problem,
- the online dual mirror descent method.

# 2 Online Resoure Allocation

In this lecture, we study the online resoure allocation problem. Let us describe the formulation due to [BLM20, BLM23]. Over a finite horizon of $T$, at each time $t \in [T]$, the decision-maker receives a request $\gamma_t = (f_t, g_t, \mathcal{X}_t)$ where

- $f_t : \mathcal{X}_t \to [0,1]$ is a concave reward function,
- $g_t : \mathcal{X}_t \to [0,1]$ is a convex resource consumption function, and
- $\mathcal{X}_t \subseteq \mathbb{R}^d$ is a convex action set.

Upon observing the request, the decision-maker chooses an action $x_t \in \mathcal{X}_t$ to serve the request, generating a reward of $f_t(x_t)$ and consuming $g_t(x_t)$ amount of resources. The decision-maker has a total resource budget of $T \cdot \rho > 0$ to allocate over the horizon $T$. Here, $\rho$ can be viewed as the average per-time budget. The goal of the decision-maker is to maximize the cumulative reward over $T$ while ensuring that the total resource consumption does not exceed the budget $T\rho$.

Denoting by $\vec{\gamma} = (\gamma_1, \dots, \gamma_T)$, we consider the resource allocation problem formulated as the following constrained optimization problem:

$$\mathrm{OPT}(\vec{\gamma}) \quad = \quad \max_{x_1 \in \mathcal{X}_1, \dots, x_T \in \mathcal{X}_T} \quad \sum_{t=1}^{T} f_t(x_t) \quad \text{s.t.} \quad \sum_{t=1}^{T} g_t(x_t) \le T\rho. \tag{26.1}$$

Here, the optimal reward $\mathrm{OPT}(\vec{\gamma})$ is the maximum cumulative reward achievable when the entire sequence of requests $\vec{\gamma}$ is known in advance.

In the online setting, an online algorithm $\mathcal{A}$ makes a real-time decision $x_t$ based on the current request $\gamma_t$ and the past history $\{\gamma_\tau, x_\tau\}_{\tau=1}^{t-1}$ without knowledge of future requests $\{\gamma_\tau\}_{\tau=t+1}^{T}$.

**Assumption 26.1.** The requests $\gamma_1, \dots, \gamma_T$ are chosen independently and identically distributed (i.i.d.) from a probability distribution $\mathcal{D}$ over the set of requests.

The performance of an online algorithm $\mathcal{A}$ is measured by the regret defined as

$$\mathrm{Regret}(\mathcal{A}) = \mathbb{E}_{\vec{\gamma} \sim \mathcal{D}^T} \left[ \mathrm{OPT}(\vec{\gamma}) - \sum_{t=1}^{T} f_t(x_t) \right],$$

where the expectation is over the randomness in the requests $\vec{\gamma}$. The goal is to design an online algorithm $\mathcal{A}$ with small regret.

The online resource allocation problem has been studied in the context of or under the name of the AdWords problem [DH09, BJN07], bandits with knapsacks [BKS13, AD14, ISSS19], repeated auctions with budgets [BG19], online stochastic matching [KVV90, FMMM09], online linear programming [GGLM82, AWY14, GM16], assortment optimization with limited inventories [GNR14], online convex programming [AD15], online binary programming [LSY20], online stochastic optimization [JLZ22], online resource allocation with concave rewards [BLM20] and nonlinear rewards [BLM23].

# 3 Online Dual Mirror Descent

To solve the online resource allocation problem under Assumption 26.1, [BLM20, BLM23] proposed an online dual mirror descent (ODMD) algorithm that is an online variant of the dual subgradient method for constrained optimization.

## 3.1 Dual Formulation

For each $t \in [T]$, we define a dual function given by

$$f_t^*(\lambda) = \sup_{x \in \mathcal{X}_t} \{f_t(x) - \lambda g_t(x)\}$$

for $\lambda \geq 0$. Then we consider

$$D(\lambda; \vec{\gamma}) = \sum_{t=1}^{T} f_t^*(\lambda) + \lambda T \rho.$$

**Lemma 26.2.** *For any $\gamma \geq 0$, we have*

$$\mathrm{OPT}(\vec{\gamma}) \leq D(\lambda; \vec{\gamma}).$$

*Proof.* Since $\lambda \geq 0$, for any optimal solution $\{x_t\}_{t=1}^T$ to (26.1), we have

$$\lambda \sum_{t=1}^{T} g_t(x_t) \leq \lambda T \rho.$$

Therefore,

$$\mathrm{OPT}(\vec{\gamma}) \leq \sum_{t=1}^{T} f_t(x_t) - \lambda \sum_{t=1}^{T} g_t(x_t) + \lambda T \rho \leq \sum_{t=1}^{T} \sup_{x \in \mathcal{X}_t} \{f_t(x) - \lambda g_t(x)\} + \lambda T \rho = D(\lambda; \vec{\gamma}),$$

as required. □

We define

$$\bar{D}(\lambda) := \frac{1}{T} \mathbb{E}_{\vec{\gamma} \sim \mathcal{D}^T} [D(\lambda; \vec{\gamma})] = \mathbb{E}_{\gamma \sim \mathcal{D}} [f^*(\lambda)] + \lambda \rho$$

where $f^*(\lambda) = \sup_{x \in \mathcal{X}} \{f(x) - \lambda g(x)\}$ for a random request $\gamma = (f, g, \mathcal{X})$ drawn from $\mathcal{D}$. Then it follows from Theorem 26.2 that for any $\lambda \geq 0$,

$$\mathbb{E}_{\vec{\gamma} \sim \mathcal{D}^T} [\mathrm{OPT}(\vec{\gamma})] \leq T \cdot \bar{D}(\lambda). \tag{26.2}$$

### 3.2 Algorithm Description

In this section, we present Algorithm 1, the ODMD algorithm due to [BLM20, BLM23]. Running the algorithm requires the following assumption:

**Assumption 26.3.** There exists a dummy action $x_0$ such that $g_t(x_0) = 0$ for all $t \in [T]$. Moreover, $x_0 \in \mathcal{X}_t$ for all $t \in [T]$.

**Algorithm 1** Online Dual Mirror Descent for Online Resource Allocation

**Initialize:** dual variable $\lambda_1 = 0$, step size $\eta > 0$, initial budget $B_1 = T\rho$.
**for** $t = 1, \ldots, T$ **do**
**(1. Action selection)**
Observe request $\gamma_t = (f_t, g_t, \mathcal{X}_t)$.
Choose a temporary action $\tilde{x}_t \in \operatorname{argmax}_{x \in \mathcal{X}_t} \{f_t(x) - \lambda_t g_t(x)\}$.
**if** the remaining budget $B_t$ is at least $g_t(\tilde{x}_t)$ **then**
Set $x_t = \tilde{x}_t$
**else**
Set $x_t = x_0$ where $x_0$ is the dummy action in Assumption 26.3.
**end if**
Update the remaining budget as $B_{t+1} = B_t - g_t(x_t)$.
**(2. Dual variable update)**
Update dual variable $\lambda_{t+1}$ as

$$\lambda_{t+1} = [\lambda_t - \eta(\rho - g_t(x_t))]_+.$$

**end for**

Algorithm 1 maintains a dual variable $\lambda_t$ at each time $t \in [T]$. Upon observing request $\gamma_t = (f_t, g_t, \mathcal{X}_t)$, it chooses a temporary action $\tilde{x}_t$ that maximizes the Lagrangian $f_t(x) - \lambda_t g_t(x)$ over $x \in \mathcal{X}_t$. If the remaining budget is sufficient to accommodate the resource consumption $g_t(\tilde{x}_t)$, it sets $x_t = \tilde{x}_t$; otherwise, it chooses the dummy action $x_0$ in Assumption 26.3 that consumes no resources. Finally, it updates the dual variable $\lambda_{t+1}$ by taking a projected subgradient step with step size $\eta > 0$.

### 3.3 Regret Analysis

In this section, we analyze the regret of Algorithm 1 under Assumptions 26.1 and 26.3. Note that due to the design of Algorithm 1, the total resource consumption never exceeds the budget $T\rho$. The following theorem characterizes the regret of Algorithm 1.

**Theorem 26.4** ([BLM20, BLM23])**.** *Under Assumptions 26.1 and 26.3, the regret of Algorithm 1 satisfies*

$$\text{Regret}(\textit{Algorithm 1}) \leq \frac{1}{\rho} + \frac{1}{2\eta\rho^2} + \frac{\eta(\rho+1)^2 T}{2}.$$

*Setting $\eta = O(1/\sqrt{T})$, we have*

$$\text{Regret}(\textit{Algorithm 1}) = O(\sqrt{T}).$$

*Proof.* We define the stopping time $\tau$ as

$$\tau = \min\left\{ s \in [T] : \; T\rho \leq 1 + \sum_{t=1}^{s} g_t(x_t) \right\}.$$

Note that the stopping time $\tau$ is a random variable depending on the requests $\vec{\gamma}$ and the actions $\{x_t\}_{t=1}^T$. By the definition of $\tau$, we have

$$\sum_{t=1}^{\tau} g_t(x_t) = \sum_{t=1}^{\tau-1} g_t(x_t) + g_\tau(x_\tau) \leq (T\rho - 1) + 1 = T\rho.$$

This implies that for $t \leq \tau$, $x_t \in \operatorname{argmax}_{x \in \mathcal{X}_t} \{f_t(x) - \lambda_t g_t(x)\}$, and therefore,

$$f_t^*(\lambda_t) = f_t(x_t) - \lambda_t g_t(x_t).$$

Let $\mathcal{F}_{t-1} = \sigma(\gamma_1, \ldots, \gamma_{t-1})$ be the filtration generated by the requests up to time $t-1$. Since $\lambda_t$ is $\mathcal{F}_{t-1}$-measurable and $\gamma_t$ is independent of $\mathcal{F}_{t-1}$ under Assumption 26.1, we have

$$\begin{aligned}
\mathbb{E}\left[f_t(x_t) \mid \mathcal{F}_{t-1}\right] &= \mathbb{E}_{\gamma_t \sim \mathcal{D}}\left[f_t^*(\lambda_t)\right] + \lambda_t \mathbb{E}\left[g_t(x_t) \mid \mathcal{F}_{t-1}\right] \\
&= \bar{D}(\lambda_t) - \lambda_t \rho + \lambda_t \mathbb{E}\left[g_t(x_t) \mid \mathcal{F}_{t-1}\right] \\
&= \bar{D}(\lambda_t) - \mathbb{E}\left[\lambda_t \left(\rho - g_t(x_t)\right) \mid \mathcal{F}_{t-1}\right]
\end{aligned}$$

where the second equality holds by the definition of $\bar{D}(\lambda)$. Now, we consider

$$Z_t = \sum_{s=1}^{t} \lambda_s \left(\rho - g_s(x_s)\right) - \mathbb{E}\left[\sum_{s=1}^{t} \lambda_s \left(\rho - g_s(x_s)\right) \mid \mathcal{F}_{s-1}\right].$$

Then $\{Z_t, \mathcal{F}_t\}_{t=1}^T$ is a martingale. Moreover, the optional stopping theorem implies that

$$\mathbb{E}\left[Z_\tau\right] = 0,$$

implying in turn that

$$\mathbb{E}\left[\sum_{t=1}^{\tau} \lambda_t \left(\rho - g_t(x_t)\right)\right] = \mathbb{E}\left[\mathbb{E}\left[\sum_{t=1}^{\tau} \lambda_t \left(\rho - g_t(x_t)\right) \mid \mathcal{F}_{t-1}\right]\right].$$

Similarly, we may deduce that

$$\mathbb{E}\left[\sum_{t=1}^{\tau} f_t(x_t)\right] = \mathbb{E}\left[\sum_{t=1}^{\tau} \bar{D}(\lambda_t)\right] - \mathbb{E}\left[\sum_{t=1}^{\tau} \lambda_t \left(\rho - g_t(x_t)\right)\right]. \tag{26.3}$$

Here, since $\bar{D}(\lambda)$ is convex in $\lambda$, we have

$$\mathbb{E}\left[\sum_{t=1}^{\tau} \bar{D}(\lambda_t)\right] \geq \mathbb{E}\left[\tau \cdot \bar{D}\left(\frac{1}{\tau}\sum_{t=1}^{\tau} \lambda_t\right)\right]. \tag{26.4}$$

Note that

$$\begin{aligned}
\mathbb{E}_{\vec{\gamma} \sim \mathcal{D}^T}\left[\mathrm{OPT}(\vec{\gamma})\right] &= \frac{\tau}{T}\mathbb{E}_{\vec{\gamma} \sim \mathcal{D}^T}\left[\mathrm{OPT}(\vec{\gamma})\right] + \frac{T-\tau}{T}\mathbb{E}_{\vec{\gamma} \sim \mathcal{D}^T}\left[\mathrm{OPT}(\vec{\gamma})\right] \\
&\leq \tau \cdot \bar{D}\left(\frac{1}{\tau}\sum_{t=1}^{\tau} \lambda_t\right) + (T - \tau)
\end{aligned} \tag{26.5}$$

where the inequality follows from (26.2) and the fact that $\mathrm{OPT}(\vec{\gamma}) \leq T$ since $f_t(x) \leq 1$ for all $t \in [T]$ and $x \in \mathcal{X}_t$. Combining the above inequalities, we obtain

$$\begin{aligned}\text{Regret(Algorithm 1)} &\leq \mathbb{E}_{\vec{\gamma}\sim\mathcal{D}^T}\left[\mathrm{OPT}(\vec{\gamma}) - \sum_{t=1}^{\tau} f_t(x_t)\right] \\ &\leq \mathbb{E}_{\vec{\gamma}\sim\mathcal{D}^T}\left[(T-\tau) + \sum_{t=1}^{\tau} \lambda_t\left(\rho - g_t(x_t)\right)\right]\end{aligned} \tag{26.6}$$

where the first inequality holds because $f_t \geq 0$ for any $t$ while the second inequality comes from (26.3)–(26.5).

Recall that we update $\lambda_t$ based on running online gradient descent on $\lambda$. Therefore, for any $\lambda \geq 0$,

$$\sum_{t=1}^{\tau} \lambda_t\left(\rho - g_t(x_t)\right) - \sum_{t=1}^{\tau} \lambda\left(\rho - g_t(x_t)\right) \leq \frac{\lambda^2}{2\eta} + \frac{\eta(\rho+1)^2 T}{2}.$$

This implies that for any $\lambda \geq 0$,

$$\text{Regret(Algorithm 1)} \leq \mathbb{E}_{\vec{\gamma}\sim\mathcal{D}^T}\left[(T-\tau) + \sum_{t=1}^{\tau} \lambda\left(\rho - g_t(x_t)\right)\right] + \frac{\lambda^2}{2\eta} + \frac{\eta(\rho+1)^2 T}{2}.$$

If $\tau = T$, then we may take $\lambda = 0$ to show

$$\text{Regret(Algorithm 1)} \leq \frac{\eta(\rho+1)^2 T}{2}.$$

If $\tau < T$, then we take $\lambda = 1/\rho$, in which case,

$$\sum_{t=1}^{\tau} \lambda\left(\rho - g_t(x_t)\right) \leq \frac{1}{\rho}\left((\tau - T)\rho + 1\right) \leq \frac{1}{\rho} - (T-\tau).$$

This means that when $\tau < T$, we have

$$\text{Regret(Algorithm 1)} \leq \frac{1}{\rho} + \frac{1}{2\eta\rho^2} + \frac{\eta(\rho+1)^2 T}{2},$$

as required. □

## References


[AD14] Shipra Agrawal and Nikhil R. Devanur. Bandits with concave rewards and convex knapsacks. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, EC '14, page 989–1006, New York, NY, USA, 2014. Association for Computing Machinery. 2

[AD15] Shipra Agrawal and Nikhil R. Devanur. Fast algorithms for online stochastic convex programming. In *Proceedings of the 2015 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1405–1424, 2015. 2

[AWY14] Shipra Agrawal, Zizhuo Wang, and Yinyu Ye. A dynamic near-optimal algorithm for online linear programming. *Operations Research*, 62(4):876–890, 2014. 2

[BG19] Santiago R. Balseiro and Yonatan Gur. Learning in repeated auctions with budgets: Regret minimization and equilibrium. *Management Science*, 65(9):3952–3968, 2019. 2

[BJN07] Niv Buchbinder, Kamal Jain, and Joseph Seffi Naor. Online primal-dual algorithms for maximizing ad-auctions revenue. In *Proceedings of the 15th Annual European Conference on Algorithms*, ESA'07, page 253–264, Berlin, Heidelberg, 2007. Springer-Verlag. 2

[BKS13] A. Badanidiyuru, R. Kleinberg, and A. Slivkins. Bandits with knapsacks. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 207–216, Los Alamitos, CA, USA, oct 2013. IEEE Computer Society. 2

[BLM20] Santiago Balseiro, Haihao Lu, and Vahab Mirrokni. Dual mirror descent for online allocation problems. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 613–628. PMLR, 13–18 Jul 2020. 2, 2, 3, 3.2, 26.4

[BLM23] Santiago R. Balseiro, Haihao Lu, and Vahab Mirrokni. The best of many worlds: Dual mirror descent for online allocation problems. *Operations Research*, 71(1):101–119, 2023. 2, 2, 3, 3.2, 26.4

[DH09] Nikhil R. Devanur and Thomas P. Hayes. The adwords problem: Online keyword matching with budgeted bidders under random permutations. In *Proceedings of the 10th ACM Conference on Electronic Commerce*, EC '09, page 71–78, New York, NY, USA, 2009. Association for Computing Machinery. 2

[FMMM09] Jon Feldman, Aranyak Mehta, Vahab Mirrokni, and S. Muthukrishnan. Online stochastic matching: Beating 1-1/e. In *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '09, page 117–126, USA, 2009. IEEE Computer Society. 2

[GGLM82] Fred Glover, Randy Glover, Joe Lorenzo, and Claude McMillan. The passenger-mix problem in the scheduled airlines. *Interfaces*, 12(3):73–80, 1982. 2

[GM16] Anupam Gupta and Marco Molinaro. How the experts algorithm can help solve lps online. *Mathematics of Operations Research*, 41(4):1404–1431, 2016. 2

[GNR14] Negin Golrezaei, Hamid Nazerzadeh, and Paat Rusmevichientong. Real-time optimization of personalized assortments. *Management Science*, 60(6):1532–1551, 2014. 2

[ISSS19] Nicole Immorlica, Karthik Abinav Sankararaman, Robert Schapire, and Aleksandrs Slivkins. Adversarial bandits with knapsacks. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 202–219, 2019. 2

[JLZ22] Jiashuo Jiang, Xiaocheng Li, and Jiawei Zhang. Online stochastic optimization with wasserstein based non-stationarity, 2022. 2

[KVV90] R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing*, STOC '90, page 352–358, New York, NY, USA, 1990. Association for Computing Machinery. 2

[LSY20] Xiaocheng Li, Chunlin Sun, and Yinyu Ye. Simple and fast algorithm for binary integer and online linear programming. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9412–9421. Curran Associates, Inc., 2020. 2