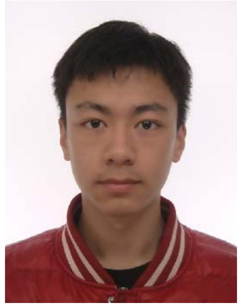
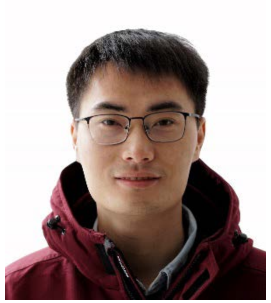
A Simple Online Algorithm for Competing with Dynamic Comparators





Yu-Jie Zhang Peng Zhao Zhi-Hua Zhou

LAMDA Group, Nanjing University

Outline

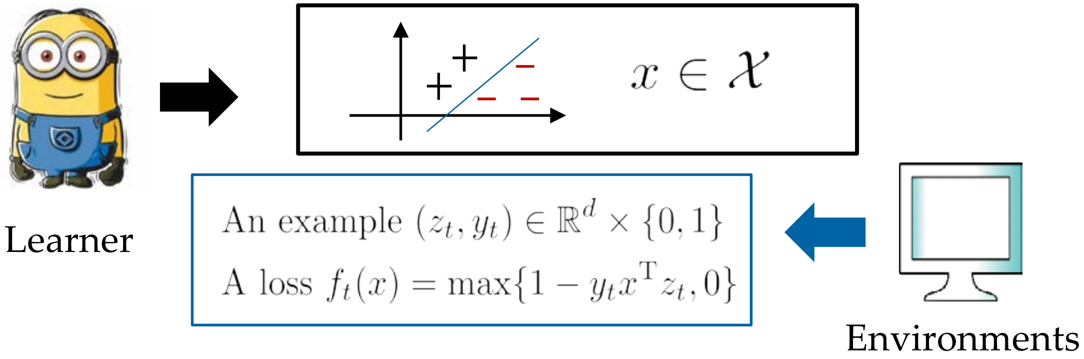
- Background Knowledge
  - Regret: Adaptivity / Non-stationarity
  - Fully Adaptive Bound
- Our Approach
- Experiments

OCO Framework and Regret

Online Convex Optimization

At each iteration $t \in T$,

- learner makes a decision $x_t \in \mathcal{X}$;
- environments reveal the convex online function $f_t(\cdot)$;
- learner suffers a loss $f_t(x_t)$ and observes $f_t(\cdot)$.



(Static) Regret

$$\mathbf{Reg}_T^s = \sum_{t=1}^{T} f_t(x_t) - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x) \leq O(\sqrt{T})$$

[Zinkevich, 2003]

Adaptivity and Non-stationarity

- *Adaptivity*: exploit benign environments while safeguard worst-case guarantee

Adaptivity	♠ variance bound	$\mathbf{Reg}_T^s \le O\left(\sqrt{\sum \|\nabla f_t(x_t) - \bar{\nabla}_T\|_*^2}\right)$
	♠ gradient variation bound	$\mathbf{Reg}_T^s \le O\left(\sqrt{\sum \sup \|\nabla f_t(x) - \nabla f_{t-1}(x)\|_*^2}\right)$
	♠ predictable sequences bound	$\mathbf{Reg}_T^s \le O\left(\sqrt{D_T}\right)$, $D_T = \sum \|\nabla f_t(x_t) - M_t\|_*^2$

- *Non-stationarity*: competing with dynamic comparators

Dynamic regret: $\mathbf{Reg}_T^d = \sum_{t=1}^T f_t(x_t) - \sum_{t=1}^T f_t(x_t^*), \quad x_t^* = \arg\min_{x \in \mathcal{X}} f_t(x)$

Non-stationarity	♠ path-length bound	$\mathbf{Reg}_T^d \le O(\sqrt{T(1+C_T)})$, $C_T = \sum_{t=2}^T \|x_t^* - x_{t-1}^*\|$
	♠ temporal variability bound	$\mathbf{Reg}_T^d \le O(T^{2/3} V_T^{1/3})$, $V_T = \sum_{t=2}^T \sup \|f_t(x) - f_{t-1}(x)\|$

Q: Can we exploit adaptivity in non-stationary environments?

Fully Adaptive Bound

- Fully adaptive bound [Jadbabaie et al., 2015]: combining adaptivity with non-stationarity

complexity measure	regret bound
path-length bound	$O(\sqrt{T(1+C_T)})$
temporal variability bound	$O(T^{2/3}V_T^{1/3})$
adaptive bound	$\mathrm{O}(\min\{\sqrt{T(1+C_T)}, T^{2/3}V_T^{1/3}\})$
fully adaptive bound	$O(\sqrt{D_T} + \min\{\sqrt{(1+D_T)(1+C_T)}, (1+D_T)^{1/3}T^{1/3}V_T^{1/3}\})$

$C_T = \sum_{t=2}^{T} \|x_t^* - x_{t-1}^*\|$

$V_T = \sum_{t=2}^{T} \sup |f_t(x) - f_{t-1}(x)|$

$D_T = \sum \|\nabla f_t(x_t) - M_t\|_*^2$

- AOMD [Jadbabaie et al., 2015]: a doubling trick based method

Algorithm 1 AOMD Jadbabaie et al. (2015)

...
for $t = 1$ to T **do**
  `% check doubling condition`
  if $L_N^2 < \gamma \min\left\{C_{(N)}\ ,\ V_{(N)}^{2/3}\Delta_N^{2/3}D_{(N)}^{-1/3}\right\} + 4R_{\max}^2$ **then**
    ...
  end if
  ...
  % update $D_{(N)}, C_{(N)}, V_{(N)}$ and Δ_N
  $D_{(N)} = D_{(N)} + \|\nabla_t - M_t\|_*^2$
  $C_{(N)} = C_{(N)} + \|x_t^* - x_{t-1}^*\|$
  $V_{(N)} = V_{(N)} + \sup_{x\in\mathcal{X}} |f_t(x) - f_{t-1}(x)|$
  $\Delta_N = \Delta_N + 1$
  ...
end for

non-convex!

A *simple* and *computationally efficient* online algorithm,
- enjoys fully adaptive bound;
- avoids non-convex program.

Optimistic Mirror Descent

- OptimisticMD: online mirror descent with an extra gradient step to exploit M_t

$$\begin{cases} \widehat{x}_{t+1} = \arg\min_{x\in\mathcal{X}} \eta\langle \nabla f_t(x_t), x\rangle + \mathcal{D}_{\mathcal{R}}(x, \widehat{x}_t) \\ x_{t+1} = \arg\min_{x\in\mathcal{X}} \eta\langle M_{t+1}, x\rangle + \mathcal{D}_{\mathcal{R}}(x, \widehat{x}_{t+1}) \end{cases} \Longrightarrow \mathbf{Reg}_T^s \le O\left(\sqrt{D_T}\right)$$

[Rakhlin and Sridharan, 2013]:

- Path-length Bound [Lemma 1]

D_T, C_T are unknown in advance!

$$\mathbf{Reg}_T^d \le \frac{\eta D_T}{2} + \frac{R_{\max}^2 + \gamma C_T}{2\eta}$$

$$\eta^*_{\texttt{path}} = \sqrt{\frac{R_{\max}^2 + \gamma C_T}{1 + D_T}} \Longrightarrow O\Big(\underbrace{\sqrt{(1+C_T)(1+D_T)}}_{\texttt{Path-Bound}}\Big)$$

- Temporal Variability Bound [Lemma 2]

$$\eta^*_{\texttt{var}} = \sqrt{\frac{C_1 + C_2\lceil T/\Delta_* \rceil}{1 + D_T}} \Longrightarrow O\big(\underbrace{(1+D_T)^{1/3}T^{1/3}V_T^{1/3}}_{\texttt{Var-Bound}}\big)$$

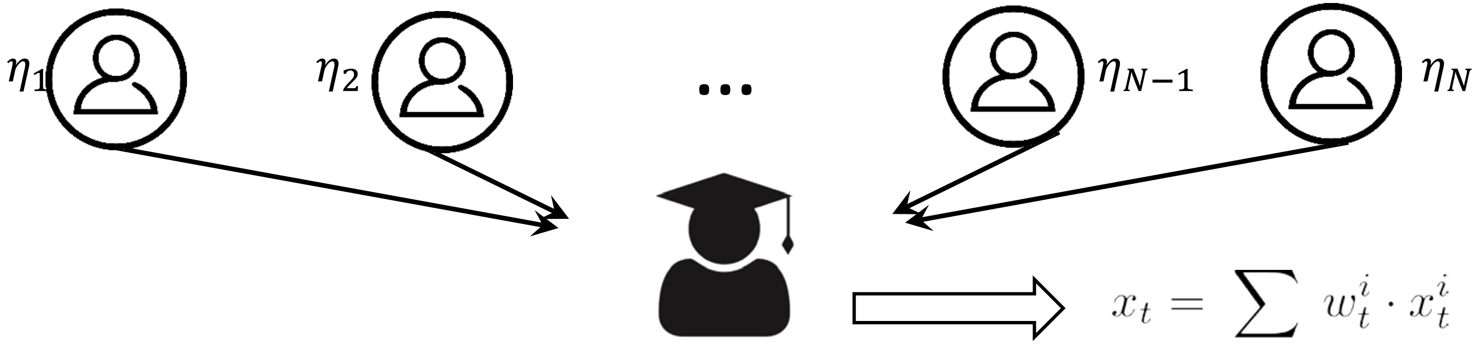
D_T, V_T are unknown in advance!

Δ_* is a constant relative to D_T, V_T and T.

Q: how to tune the step size?

Master-Base aggregation

- Running OptimisticMDs with multiple step sizes and aggregating the predictions



$$x_t = \sum_{i\in[N]} w_t^i \cdot x_t^i$$

- *Base algorithm (OptimisticMDs)*: there exists a base algorithm enjoys
  - path-length: $\texttt{Base-regret} \leq O(\texttt{Path-Bound})$
  - temporal variability: $\texttt{Base-regret} \leq O(\texttt{Var-Bound})$

  $\Big\}\ O(\min\{\texttt{Path-Bound}, \texttt{Var-Bound}\})$

- *Master algorithm*: we design a novel algorithm based on Optimistic Hedge
  - regret w.r.t. any base algorithm: $\texttt{Master-regret} \leq O(\sqrt{1 + D_T})$

Algorithm

Algorithm 1 `Master Algorithm`

Input: step size ϵ, parameter pools $\mathcal{P}$

Initiate N base algorithms $\mathcal{S} = \{S_i \mid i \in [N]\}$ by running Algorithm 2 with each step size $\eta_i \in \mathcal{P}$
Initialize: $L_0^i = 0$ for all $i \in [N]$, and receive M_1
for $t = 1$ **to** T **do**
    Receive x_t^i from base and update weights by (9)
    Play $x_t = \sum_{i \in [N]} w_t^i x_t^i$
    Observe the function $f_t(\cdot)$, query the gradient $\nabla f_t(x_t)$ and receive M_{t+1}
    Update $L_t^i = L_{t-1}^i + \langle \nabla f_t(x_t), x_t^i - x_t \rangle$
    Send $\nabla f_t(x_t)$ and M_{t+1} to base algorithms
end for

track

Algorithm 2 OMD `(Base Algorithm)`

Input: step size $\eta_i \in \mathcal{P}$

Let x_1^i be any point in $\mathcal{X}$
for $t = 1$ **to** T **do**
    Submit x_t^i to the master, then receive the gradient $\nabla f_t(x_t)$ and current predictable sequence M_{t+1}
    Prepare the prediction for the next iteration as,

$$\widehat{x}_{t+1}^i = \arg\min_{x \in \mathcal{X}} \eta_i \langle \nabla f_t(x_t), x \rangle + \mathcal{D}_{\mathcal{R}}(x, \widehat{x}_t^i),$$

$$x_{t+1}^i = \arg\min_{x \in \mathcal{X}} \eta_i \langle M_{t+1}, x \rangle + \mathcal{D}_{\mathcal{R}}(x, \widehat{x}_{t+1}^i)$$

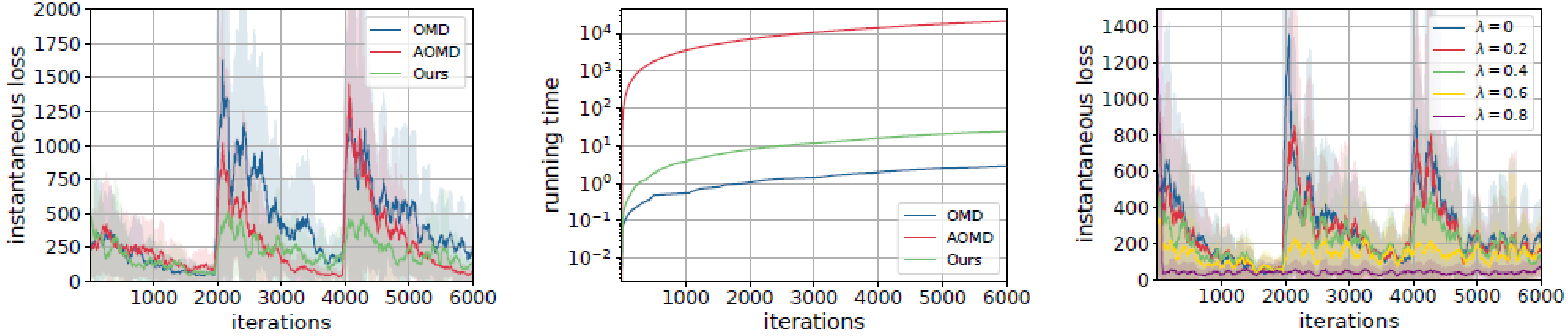
end for

$\texttt{Master-regret} \le O(\sqrt{1 + D_T})$ $+$ $\texttt{Base-regret} \le O(\texttt{Path-Bound}, \texttt{Var-Bound})$

$\Longrightarrow O(\sqrt{D_T} + \min\{\sqrt{(1 + D_T)(1 + C_T)}, (1 + D_T)^{1/3} T^{1/3} V_T^{1/3}\})$ Fully adaptive bound

- Our algorithm does not involve non-convex optimization problem solving
- We further accelerate the algorithm by reducing the gradient querying times to 1

Experiments



(a) Comparison on instantaneous loss (b) Comparison on running time (c) Different qualities of M_t

◻ Effectiveness:

- ➢ Non-stationarity (Figure a): Ours ≈ AOMD > OMD
- ➢ Adaptivity (Figure c):Better quality of M_t, greater performance.

◻ Efficiency

- ➢ Running time (Figure b): *24 seconds* for our algorithm while around *6 hours* for AOMD

Summary

- We design a computationally *efficient* method with the fully adaptive *dynamic regret* bound.
- Our method is based on the *master-base framework*, and a novel master algorithm is carefully designed to achieve the compatible bound.
- Empirical results validate the *effectiveness* and *efficiency*.

Thanks!