# Optimal Distributed Online Prediction using Mini-Batches

Ofer Dekel, Ran Gilad-Bachrach,
Ohad Shamir, and Lin Xiao

Microsoft Research

NIPS Workshop on Learning on Cores, Clusters and Clouds

December 11, 2010

# Motivation

- online algorithms often studied in serial setting
  - fast, simple, good generalization, ...
  - but *sequential* in nature
- web-scale online prediction (e.g., search engines)
  - inputs arrive at *high rate*
  - need to provide *real-time* service

  critical to use parallel/distributed computing
- how well can online algorithms (old or new) perform in distributed setting?

# Stochastic online prediction

- repeat for each $i = 1, 2, 3, \ldots$
  - predict $w_i \in W$ (e.g., based on $\nabla f(w_{i-1}, z_{i-1})$)
  - receive $z_i$ drawn i.i.d. from fixed distribution
  - suffer loss $f(w_i, z_i)$

- measure quality of predictions using *regret*

$$R(m) = \sum_{i=1}^{m} \left( f(w_i, z_i) - f(w^\star, z_i) \right)$$

  - $w^\star = \arg\min_{w \in W} \mathbb{E}_z[f(w, z)]$
  - assume $f(\cdot, z)$ convex, $W$ closed and convex

# Stochastic optimization

- find approximate solution to

$$\underset{w \in W}{\text{minimize}} \quad F(w) \triangleq \mathbb{E}_z[f(w,z)]$$

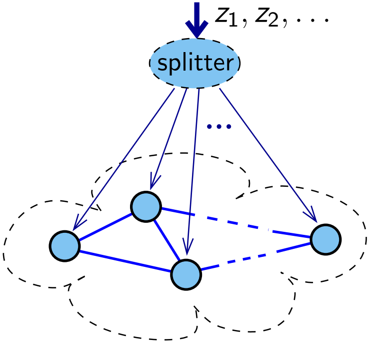
- success measured by *optimality gap*

$$G(m) = F(w_m) - F(w^\star)$$

- different motivations
  - often used to solve large-scale batch problem
  - usually no real-time requirement
- how can parallel computing speed up solution?

# Distributed online prediction

- system has $k$ nodes
- network model
  - limited bandwidth
  - latency
  - *non-blocking*
- measure same regret




$$R(m) = \sum_{i=1}^{m} \big( f(w_i, z_i) - f(w^\star, z_i) \big)$$

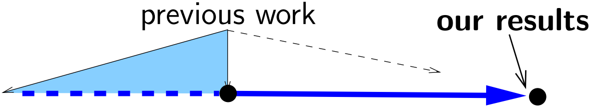
# Limits of performance

- an ideal (but unrealistic) solution
  - run serial algorithm on a "super" computer that is $k$ times faster
  - optimal regret bound: $\mathbb{E}[R(m)] \leq O(\sqrt{m})$
- a trivial (no-communication) solution
  - each node operates in isolation
  - regret bound scales poorly with network size $k$

$$\mathbb{E}[R(m)] \ \leq \ k \cdot O\left(\sqrt{m/k}\right) = O\left(\sqrt{km}\right)$$

# Related work and contribution

- previous work on distributed optimization
  - Tsitsiklis, Bertsekas and Athans (1986); Tsitsiklis and Bertsekas (1989); Nedić, Bertsekas and Bokar (2001); Nedić and Ozdaglar (2009); ...
  - Langford, Smola and Zinkevich (2009); Duchi, Agarwal and Wainwright (2010); Zinkevich, Weimar, Smola and Li (2010); ...
- when applied to problems considered here




| | trivial | ideal |
|---|---|---|
| online prediction | $O(\sqrt{km})$ | $O(\sqrt{m})$ |
| stochastic optimization | $O\left(\frac{1}{\sqrt{T}}\right)$ | $O\left(\frac{1}{\sqrt{kT}}\right)$ |

# Outline

- motivation and introduction
- variance bounds for serial algorithms
- DMB algorithm and regret bounds
- parallel stochastic optimization
- experiments on a web-scale problem

# Serial online algorithms

- projected gradient descent

$$w_{j+1} = \pi_W\left(w_j - \frac{1}{\alpha_j} g_j\right)$$

- dual averaging method

$$w_{j+1} = \operatorname*{arg\,min}_{w \in W} \left\{ \left\langle \sum_{i=1}^{j} g_i, w \right\rangle + \alpha_j \, h(w) \right\}$$

optimal regret bound (attained by $\alpha_j = \Theta(\sqrt{j})$):

$$\mathbb{E}[R(m)] = O(\sqrt{m})$$

# Variance bounds

- additional assumptions
  - smoothness: $\forall\, z \in Z,\ \forall\, w, w' \in W$,
  $$\|\nabla_w f(w,z) - \nabla_w f(w',z)\| \le L\|w - w'\|$$
  - bounded gradient variance: $\forall\, w \in W$,
  $$\mathbb{E}_z\left[\left\|\nabla_w f(w,z) - \nabla F(w)]\right\|^2\right] \le \sigma^2$$
- **Theorem:** refined bound using $\alpha_j = L + (\sigma/D)\sqrt{j}$

$$\mathbb{E}[R(m)] \ \le \ 2D^2L + 2D\sigma\sqrt{m} \ \triangleq \psi(\sigma^2, m)$$

# Variance reduction via mini-batching

- mini-batching
  - predict $b$ samples using same predictor
  - update predictor based on average gradients

  *not a new idea, but no theoretical support*

- our analysis: consider averaged cost function

$$\bar{f}\left(w,(z_1,\ldots,z_b)\right) \triangleq \frac{1}{b}\sum_{s=1}^{b} f(w,z_s)$$

  - $\nabla_w \bar{f}$ has variance $\frac{\sigma^2}{b}$; at most $\lceil \frac{m}{b} \rceil$ batches
  - serial regret bound:

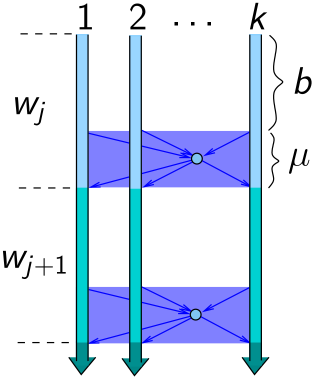
$$b \cdot \psi\left(\frac{\sigma^2}{b}, \left\lceil \frac{m}{b} \right\rceil\right) \leq 2bD^2L + 2D\sigma\sqrt{m+b}$$

# Distributed mini-batch (DMB)

- for each node
  - accumulate gradients of first $b/k$ inputs
  - vector-sum to compute $\bar{g}_j$ over $b$ gradients
  - update $w_{j+1}$ based on $\bar{g}_j$

- expected regret bound

$$(b+\mu)\,\psi\left(\frac{\sigma^2}{b}, \left\lceil \frac{m}{b+\mu} \right\rceil\right)$$



# Regret bound for DMB

- suppose $\psi(\sigma^2, m) = 2D^2L + 2D\sigma\sqrt{m}$
  - if $b = m^\rho$ for any $\rho \in (0, 1/2)$, then
    $$\mathbb{E}[R(m)] \;\leq\; 2D\sigma\sqrt{m} + o(\sqrt{m})$$
  - choose $b = m^{1/3}$, bound becomes
    $$2D\sigma\sqrt{m} + 2D\left(LD + \sigma\sqrt{\mu}\right)m^{1/3} + O(m^{1/6})$$
- asymptotically optimal: dominant term same as in ideal serial solution
- scale nicely with latency: often $\mu \propto \log(k)$

# Stochastic Optimization

- find approximate solution to

$$\underset{w \in W}{\text{minimize}} \quad F(w) \triangleq \mathbb{E}_z[f(w,z)]$$

- success measured by optimality gap

$$G(m) = F(\bar{w}_m) - F(w^\star)$$

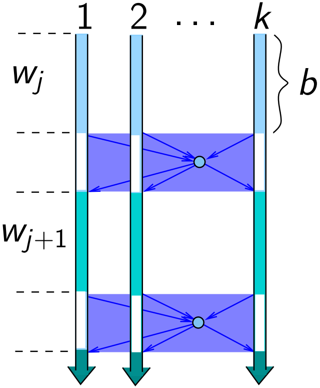
- for convex loss and i.i.d. inputs

$$\mathbb{E}[G(m)] \leq \frac{1}{m}\mathbb{E}[R(m)] \leq \frac{1}{m}\psi(\sigma^2, m) \triangleq \bar{\psi}(\sigma^2, m)$$

# DMB for stochastic optimization

- for each node
  - accumulate gradients of $b/k$ inputs
  - vector-sum to compute $\bar{g}_j$ over $b$ gradients
  - update $w_{j+1}$ based on $\bar{g}_j$
- bound on optimality gap

$$\mathbb{E}[G(m)] \leq \bar{\psi}\left(\frac{\sigma^2}{b}, \frac{m}{b}\right)$$



# DMB for stochastic optimization

- if serial gap is $\bar{\psi}(\sigma^2, m) = \frac{2D^2L}{m} + \frac{2D\sigma}{\sqrt{m}}$, then

$$\mathbb{E}[G(m)] \leq \bar{\psi}\left(\frac{\sigma^2}{b}, \frac{m}{b}\right) = \frac{2bD^2L}{m} + \frac{2D\sigma}{\sqrt{m}}$$

- parallel speed-up

$$S = \frac{m}{\frac{m}{b}\left(\frac{b}{k} + \delta\right)} = \frac{k}{1 + \frac{\delta}{b}k}$$

  - asymptotic linear speed-up with $b \propto m^{1/3}$
  - similar result for reaching same optimality gap

# Web-scale experiments

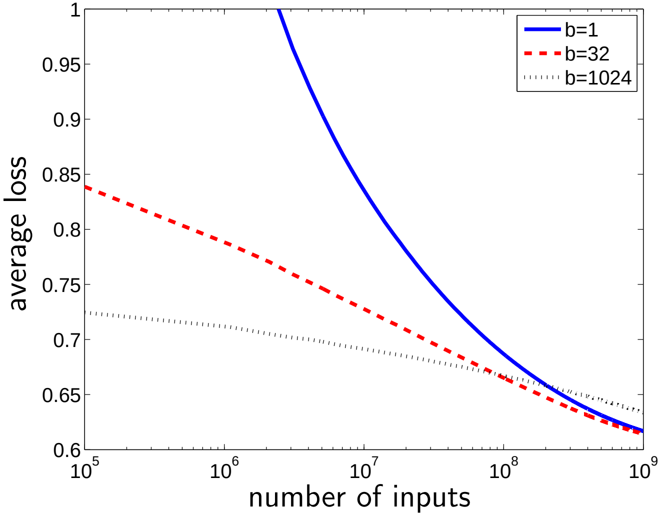
- an online binary prediction problem
  - predict *highly monetizable* queries
  - log of $10^9$ queries issued to a commercial search engine

- logistic loss function

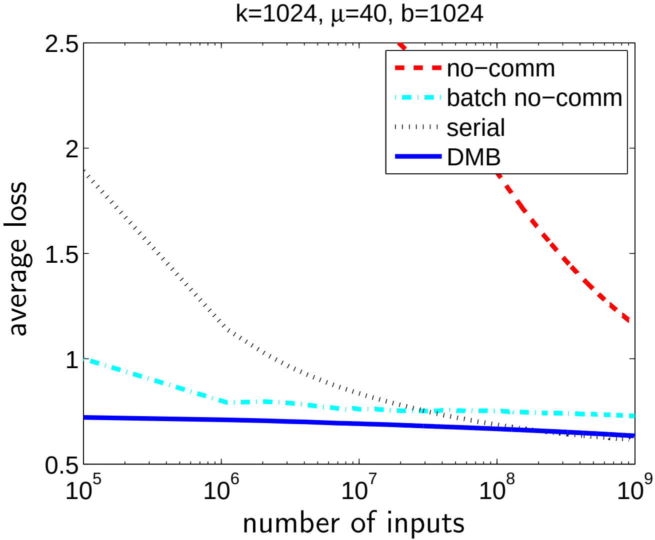
$$f(w, z) = \log\big(1 + \exp(-\langle w, z\rangle)\big)$$

- algorithm: stochastic dual averaging method (separate $5x10^8$ queries for parameter tuning)

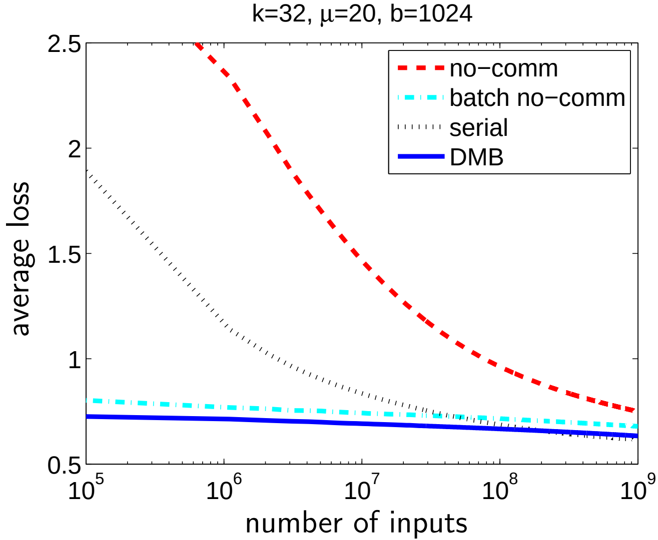
# Experiments: serial mini-batching



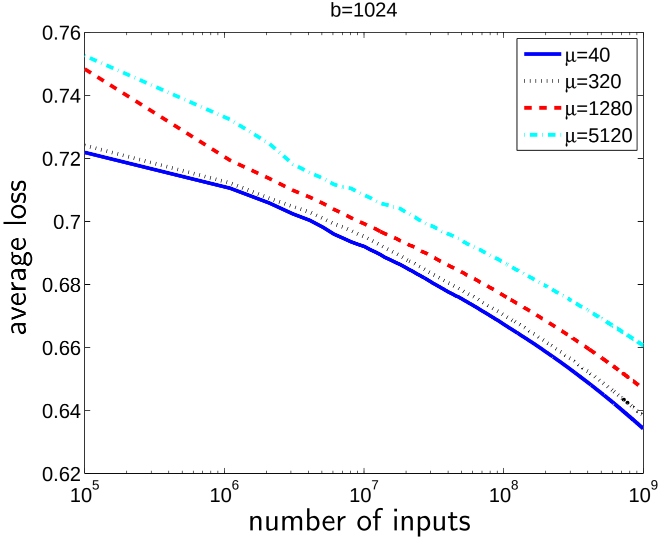
# Experiments: DMB vs. others



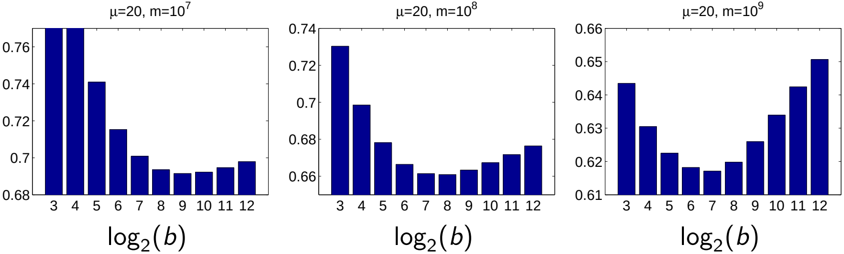
# Experiments: DMB vs. others



# Experiments: effects of latency



# Experiments: optimal batch size




- fixed cluster size $k = 32$ (latency $\mu = 20$)
- empirical observations
  - large batch size ($b = 512$) beneficial at first
  - small batch size ($b = 128$) better in the end

# Summary

- distributed stochastic online prediction
  - DMB turns serial algorithms into parallel ones
  - optimal $O(\sqrt{m})$ regret bound for smooth loss
- stochastic optimization: near linear speed-up
- *first* provable demonstration that distributed computing worthwhile for these two problems

**future directions**

- DMB in asynchronous distributed environment (progress made, report available on arXiv)
- non-smooth functions? non-stochastic inputs?