Fast Distributed Coordinate Descent (Hydra^2)

Olivier Fercoq, Zheng Qu, Peter Richtárik and Martin Takáč

olivier.fercoq@ed.ac.uk, zheng.qu@ed.ac.uk, Peter.Richtarik@ed.ac.uk, Takac.MT@gmail.com

1. Problem Formulation

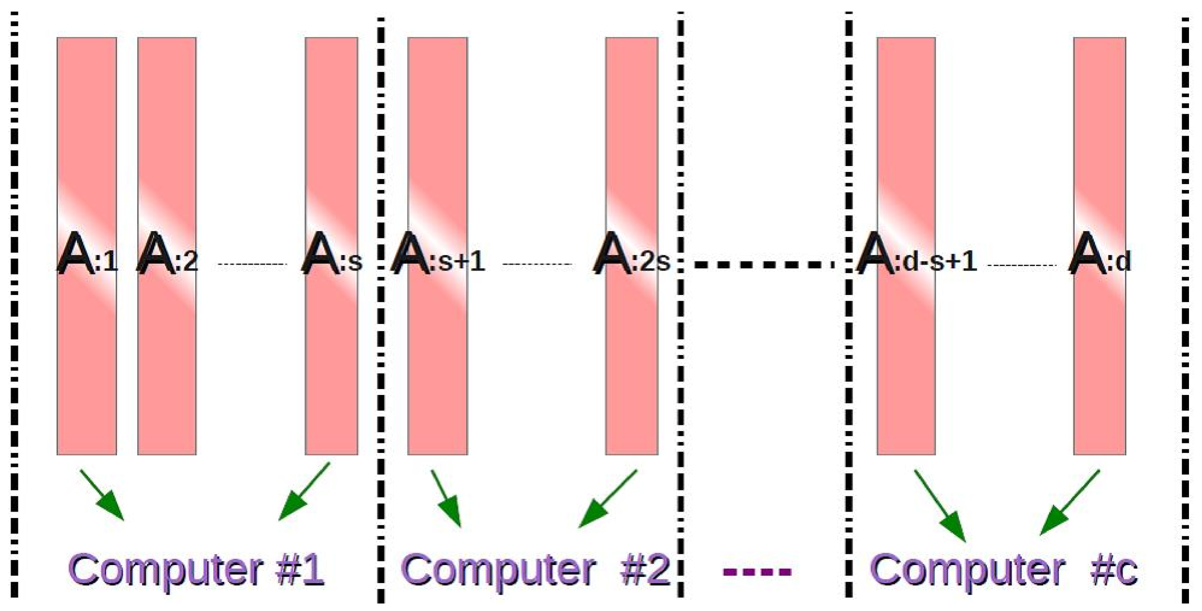
Consider the following optimization problem

$$\min_{x=(x^1,\dots,x^d)\in\mathbb{R}^d} L(x) \equiv f(x) + \sum_{i=1}^{d} \Psi_i(x^i)$$

- $f:\mathbb{R}^d\to\mathbb{R}$ is a convex differentiable loss function such that for all $x,h\in\mathbb{R}^d$
  $f(x+h)\leqslant f(x)+(\nabla f(x))^\top h+\frac{1}{2}h^\top\mathbf{A}^\top\mathbf{A}h,$
  where $\mathbf{A}$ is some available n-by-d matrix.
  Ex: $f(x)=\sum_{j=1}^n \ell(x,\mathbf{A}_{j:},y^j)$
  where $\mathbf{A}_{j:}$ denotes the j-th sample/example and $\ell:\mathbb{R}\to\mathbb{R}$ is some loss function:
  - square loss: $\frac{1}{2}(y^j-\mathbf{A}_{j:}x)^2$
  - logistic loss: $\log(1+\exp(-y^j\mathbf{A}_{j:}x))$
- $\Psi_i:\mathbb{R}\to\mathbb{R}\cup\{+\infty\}$ is convex and simple.
  Ex: - L_1 regularizer: $\lambda\|x\|_1$
  - SVM dual: $I_{[0,1]^d}(x)$

2. Data Distribution

Data distribution is crucial for problems whose size exceeds available memory of a single computer! We have c nodes available. The coordinates $\{1,\dots,d\}$ are partitioned into c sets $\{\mathcal{P}_l : l=1,\dots,c\}$, each of size $s:=d/c$. The columns of matrix $\mathbf{A}$ are partitioned accordingly, with those belonging to $\mathcal{P}_l$ stored on node l. Each processor selects uniformly random subset $\hat{S}_l\subseteq\mathcal{P}_l$ of cardinality τ, forming the distributed sampling $\hat{S}=\cup_l\hat{S}_l$.



3. Hydra^2, Approx and Hydra

- Hydra (HYbriD cooRdinAte descent) [2] is the first distributed coordinate descent method;
- APPROX (Accelerated Parallel PROXimal) [3] is the first accelerated coordinate descent method;
- Hydra [2]+APPROX [3]$\Longrightarrow$Hydra^2 [1].

4. Hydra^2

Algorithm 1: Hydra^2

pick $z_0\in\mathrm{dom}(\Psi)$, set $\theta_0=\tau/s$ and $u_0=0$

for $k=0,1,\dots$ **do**

- $z_{k+1}\leftarrow z_k$, $u_{k+1}\leftarrow u_k$
- **for** *each computer* $l\in\{1,\dots,c\}$ **do**
  - Randomly choose $\hat{S}_l\subseteq\mathcal{P}_l$
  - **for** *each* $i\in\hat{S}_l$ **do**
    - $t_k^i=\arg\min_t \nabla_i f(\theta_k^2 u_k+z_k)t+\frac{s\theta_k\mathbf{D}_i}{2\tau}t^2+\Psi_i(z_k^i+t)$
    - $z_{k+1}^i\leftarrow z_k^i+t_k^i$
    - $u_{k+1}^i\leftarrow u_k^i-(\frac{1}{\theta_k^2}-\frac{s}{\tau\theta_k})t_k^i$
- $\theta_{k+1}=\frac{1}{2}(\sqrt{{\theta_k}^4+4{\theta_k}^2}-{\theta_k}^2)$

OUTPUT: $x_{k+1}:=\theta_k^2 u_{k+1}+z_{k+1}$

- If $\theta_k\equiv\theta_0$, then Hydra^2 reduces to Hydra [2].
- The parameters $\{\mathbf{D}_i\}_i$ should be chosen such that $(f,\hat{S})\sim ESO(\mathbf{D})$, namely, the following **ESO** (Expected Separable Overapproximation) inequality holds for all $x,h\in\mathbb{R}^d$:

$$\mathbf{E}[f(x+h^{\hat{S}})]\leqslant f(x)+\frac{\mathbf{E}[|\hat{S}|]}{d}\left((\nabla f(x))^\top h+\frac{1}{2}\|h\|^2_{\mathbf{D}}\right).$$

5. Accelerated Convergence

Theorem. If $(f,\hat{S})\sim ESO(\mathbf{D})$, then,

$$\mathbf{E}[L(x_k)-L(x^*)]\leqslant\frac{C_1+C_2}{((k-1)\tau/s+2)^2},\quad\forall k\geqslant 1.$$

where

$$C_1=\left(1-\tfrac{\tau}{s}\right)(L(x_0)-L(x_*)),$$

$$C_2=\sum_{i=1}^d \mathbf{D}_i(x_0^i-x_*^i)^2.$$

6. Important Quantities

$$\begin{aligned}
\omega_j &:= \max\{x^\top\mathbf{A}_{j:}^\top\mathbf{A}_{j:}x \;:\; x^\top D^{\mathbf{A}_{j:}^\top\mathbf{A}_{j:}}x\leqslant 1\},\\
\omega'_j &:= \max\{x^\top\mathbf{A}_{j:}^\top\mathbf{A}_{j:}x \;:\; x^\top B^{\mathbf{A}_{j:}^\top\mathbf{A}_{j:}}x\leqslant 1\}\\
\sigma &:= \max\{x^\top\mathbf{A}^\top\mathbf{A}x : x^\top D^{\mathbf{A}^\top\mathbf{A}}x\leqslant 1\},\\
\sigma' &:= \max\{x^\top\mathbf{A}^\top\mathbf{A}x : x^\top B^{\mathbf{A}^\top\mathbf{A}}x\leqslant 1\}.
\end{aligned}\tag{1}$$

For any matrix $\mathbf{G}$, $D^{\mathbf{G}}$ denotes the diagonal matrix of $\mathbf{G}$ and $B^{\mathbf{G}}$ the block diagonal matrix of $\mathbf{G}$ associated to the partition $\{\mathcal{P}_1,\dots,\mathcal{P}_c\}$.

7. Four different stepsizes

The following four parameters all satisfy the **ESO** inequality.

$$\mathbf{D}_i^1=\sum_{j=1}^n[\underbrace{1+\frac{(\tau-1)(\omega_j-1)}{s_1}}_{\alpha_{j,1}}+\underbrace{(\frac{\tau}{s}-\frac{\tau-1}{s_1})\frac{\omega'_j-1}{\omega'_j}\omega_j}_{\alpha_{j,2}}]\mathbf{A}_{ji}^2$$

$$\mathbf{D}_i^2=[\underbrace{1+\frac{(\tau-1)(\sigma-1)}{s_1}}_{\beta_1}+\underbrace{(\frac{\tau}{s}-\frac{\tau-1}{s_1})\frac{\sigma'-1}{\sigma'}\sigma}_{\beta_2}]\sum_{j=1}^n\mathbf{A}_{ji}^2$$

$$\mathbf{D}_i^3=2\left(1+\frac{\tau-1}{s_1}(\max_j\omega_j-1)\right)\sum_{j=1}^n\mathbf{A}_{ji}^2$$

$$\mathbf{D}_i^4=\left(\frac{\tau}{\tau-1}1+\frac{\tau}{s_1}(\max_i\frac{\sum_{j=1}^n\omega_j\mathbf{A}_{ji}^2}{\sum_{j=1}^n\mathbf{A}_{ji}^2}-1)\right)\sum_{j=1}^n\mathbf{A}_{ji}^2$$

where $s_1=\max(s-1,1)$.

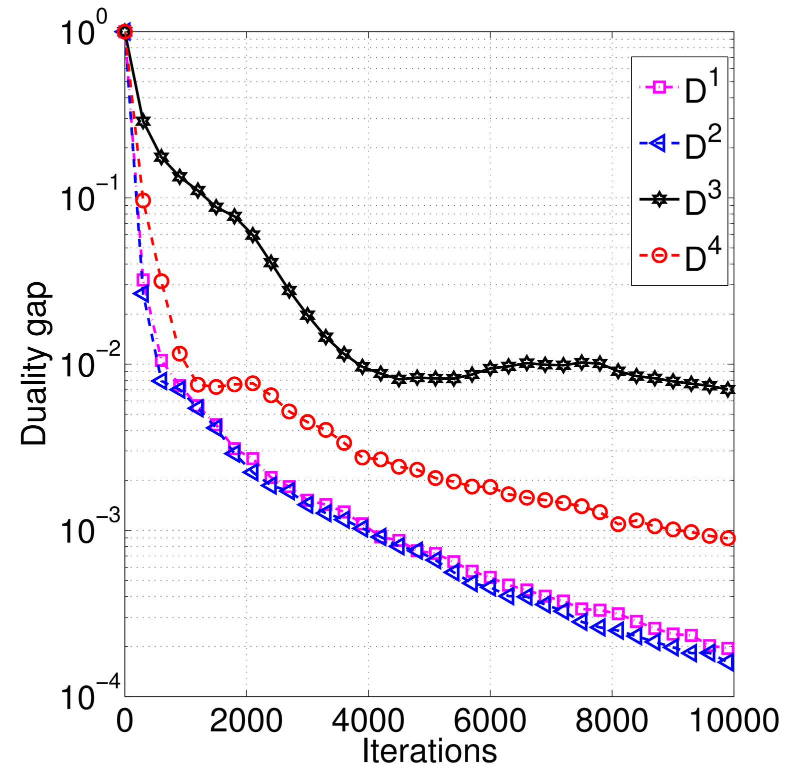
Remark. $\mathbf{D}_i^3$ was proposed in [2] as an easily computable upper bound of $\mathbf{D}_i^2$.

7. Comparison of stepsizes

Lemma. Let $\tau\geqslant 2$. Then for all $i\in\{1,\dots,d\}$:

$$\mathbf{D}_i^1\leqslant\mathbf{D}_i^4\leqslant\mathbf{D}_i^3,\quad\mathbf{D}_i^2\leqslant\mathbf{D}_i^4\leqslant\mathbf{D}_i^3.$$

In order to investigate the benefit of the new stepsize parameters, we solved the SVM dual problem on the *astro-ph* dataset with $d=29{,}882$ samples and $n=99{,}757$ features for $(c,\tau)=(32,10)$. We plot the evolution of the duality gap, obtained by using the four different stepsize parameters. We see clearly that smaller stepsize parameters lead to faster convergence, as predicted by Theorem. Moreover, using our easily computable new stepsize parameters $\{\mathbf{D}_i^1\}_i$, we achieve comparable convergence speed with respect to the existing but not easily computable parameters $\{\mathbf{D}_i^2\}_i$.



8. Weak effect of partition

Lemma. If $\tau\geqslant 2$, then

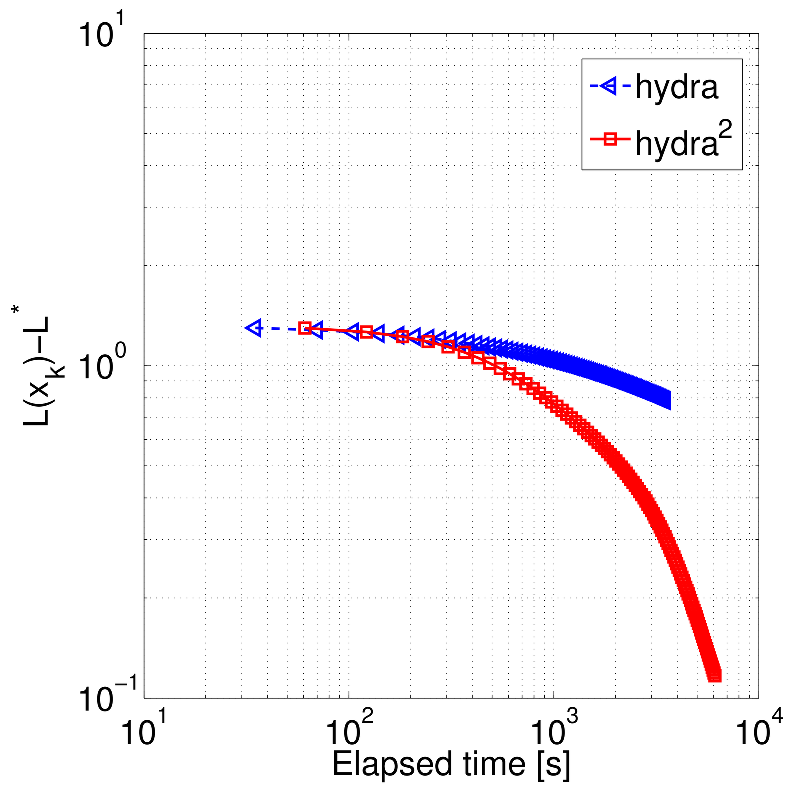
$$\beta_2\leqslant\frac{\beta_1}{\tau-1},\quad\alpha_{2,j}\leqslant\frac{\alpha_{1,j}}{\tau-1},\quad\forall j=1,\dots,n.$$

Insight: as long as $\tau\geqslant 2$, the effect of partitioning the data (across the nodes) on the iteration complexity of Hydra^2 is negligible, and vanishes as τ increases.

9. Big data Experiment

We compare Hydra with Hydra^2 on a synthetic big data LASSO problem. Dimension of matrix $\mathbf{A}$: $d=50$ billion, $n=5{,}000{,}000$. Dataset size: 5TB.

We have used 128 physical Cray XC30 compute nodes connected via Aries interconnect. On each physical node we have run two MPI processes (hence $c=256$ and $s=195{,}312{,}500$)– each process runs 24 OpenMP threads (Hyperthreads). In order to minimize communication we have chosen $\tau=s/1000$ (hence each thread computed an update for 8,138 coordinates during one iteration, on average).



10. References

[1] Fercoq O., Qu Z., Richtárik, P., Takáč, M. : Fast distributed coordinate descent for non-strongly convex losses, *IEEE workshop on Machine Learning for Signal Processing, 2014.*

[2] Richtárik, P., Takáč, M.: Distributed Coordinate Descent Method for learning with Big Data, *arXiv:1310.2059, 2013.*

[3] Fercoq O., Richtárik, P.: Accelerated, parallel and proximal coordinate descent, *arXiv:1312.5799, 2013.*