BPGrad: Towards Global Optimality in Deep Learning via Branch and Pruning

Ziming Zhang*, Yuanwei Wu* and Guanghui Wang

zzhang@merl.com, y262w558@ku.edu, ghwang@ku.edu

MITSUBISHI ELECTRIC Changes for the Better | KU KANSAS | CVPR 2018 SALT LAKE CITY • JUNE 18-22

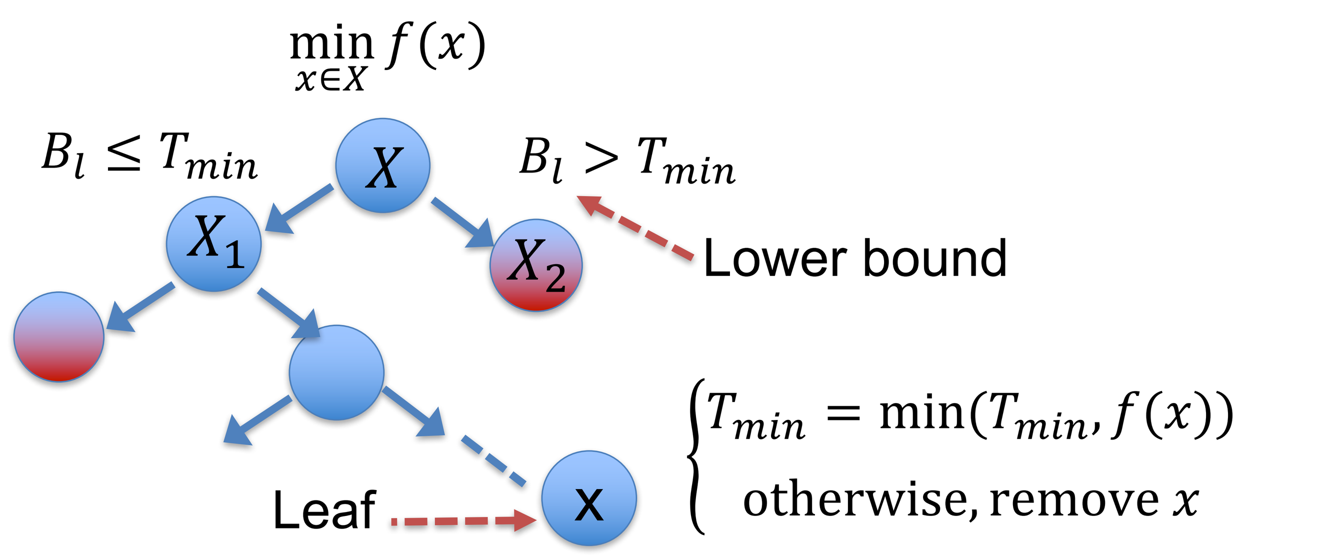
SETTING

Problem: Localizing global optima in deep learning.

Contributions:

1) Explored the possibility of locating global optimality in DL from the **algorithmic** perspective;
2) Proposed an efficient SGD-based solver, BPGrad, with **Branch & Pruning**;
3) Empirically good performance of BPGrad solver on object recognition, detection, and segmentation.

Branch & Pruning:



BPGRAD ALGORITHM

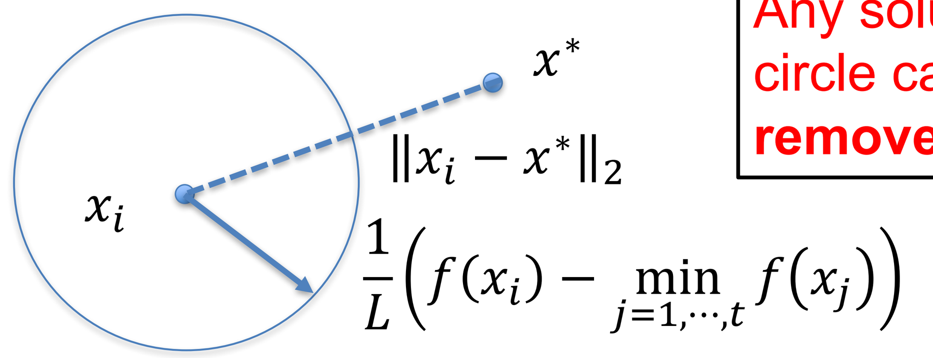
Assumptions:

1) The objective f has both lower and upper bounds;
2) f is differentiable in the parameter space;
3) f is Lipschitz continuous, or can be approximated by Lipschitz functions, with constant $L \geq 0$.

Lipschitz Continuity:

$$|f(x_1) - f(x_2)| \leq L\|x_1 - x_2\|_2, \forall x_1, x_2 \in X$$

$$f(x_i) - L\|x_i - x^*\|_2 \leq f^* \leq \min_{j=1,\cdots,t} f(x_j), \forall i \in [t]$$



Any solution inside the circle can be safely **removed** as pruning

BPGrad Algorithm for Lipschitz functions:

$$\min_{x_{t+1} \in X, \eta_t \geq 0} \left\| x_{t+1} - \left(x_t - \eta_t \nabla \tilde{f}(x_t)\right) \right\|_2^2 + \gamma \eta_t^2, (5)$$

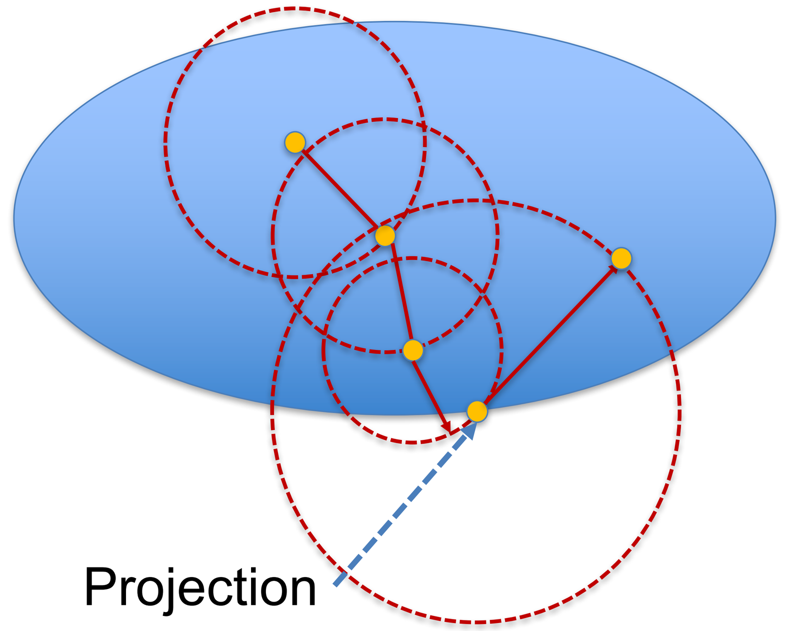
$$s.t. \max_{i=1,\cdots,t} \{f(x_i) - L\|x_i - x_{t+1}\|_2\} \leq \rho \min_{i=1,\cdots,t} f(x_i), (4)$$

Algorithm 1 BPGrad Algorithm for Deep Learning

Input : objective function f with Lipschitz constant $L \geq 0$, precision $\epsilon \geq 0$

Output : minimizer $\mathbf{x}^*$

Randomly initialize $\mathbf{x}_1$, $t \leftarrow 1$, $\rho \leftarrow 0$;
while $\min_{i=1,\cdots,t} f(\mathbf{x}_i) \leq \frac{\epsilon}{1-\rho}$ **do**
  while $\exists \mathbf{x}_{t+1} \in \mathcal{X}$ *satisfies Eq. (4)* **do**
    Compute $\mathbf{x}_{t+1}$ by solving Eq. (5);
    $t \leftarrow t+1$;
  end
  Increase ρ such that $0 \leq \rho < 1$ still holds;
end
return $\mathbf{x}^* = \mathbf{x}_{i^*}$ where $i^* \in \arg\min_{i=1,\cdots,t} f(\mathbf{x}_i)$;



- Global optimality
- Tightness bound guarantee
- Provable convergence

- Tracking of infeasible solutions
- Exponential samples

BPGRAD FOR DEEP LEARNING

Approximate DL solver based on BPGrad:

A1. Minimizing distortion is more important than minimizing step sizes, i.e. $\gamma \ll 1$;
A2. X is sufficiently large where $\exists \eta_t \geq 0$ so that $x_{t+1} = x_t - \eta_t \nabla \tilde{f}(x_t) \in X \setminus X_{\mathbb{R}}(t)$ always holds;
A3. $\eta_t \geq 0$ is always sufficiently small for local update;
A4. x_{t+1} can be sampled only based on x_t and $\nabla \tilde{f}(x_t)$.

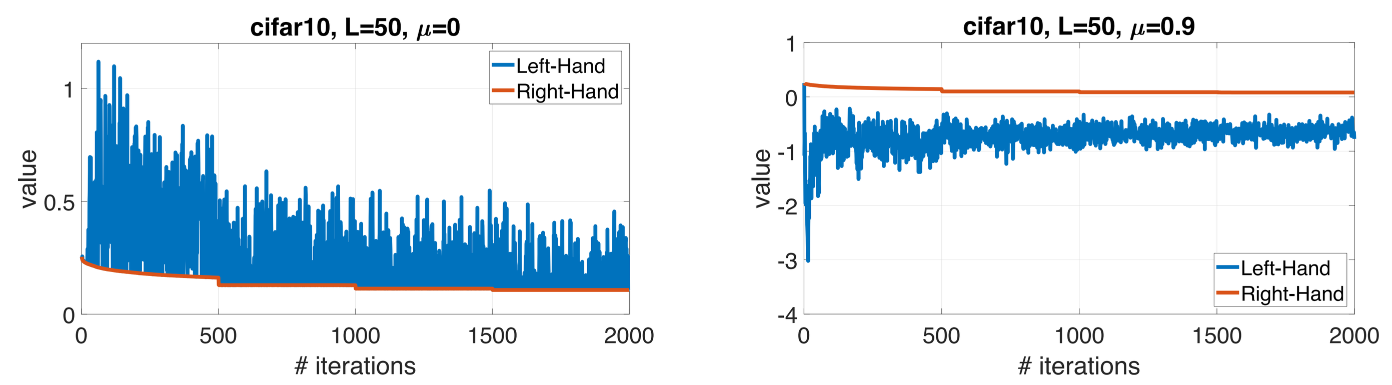
Algorithm 2 BPGrad based Solver for Deep Learning

Input : number of evaluations n repeating N times at most, objective function f with Lipschitz constant $L \geq 0$, momentum $0 \leq \mu \leq 1$

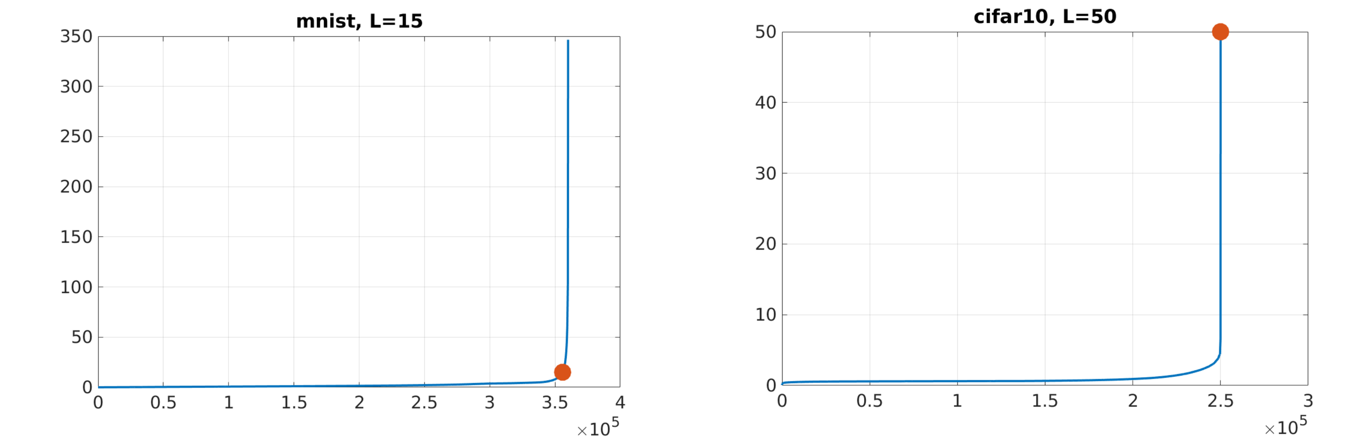
Output : minimizer $\mathbf{x}^*$

$t \leftarrow 1, \mathbf{v}_1 \leftarrow \mathbf{0}$, and randomly initialize $\mathbf{x}_1$;
for $m \leftarrow 1$ **to** N **do**
  $\rho \leftarrow 1 - \frac{1}{m}$;
  while $t < mn$ **do**
    $\mathbf{v}_{t+1} \leftarrow \mu \mathbf{v}_t - \frac{f(\mathbf{x}_t) - \rho \min_{i=1,\cdots,t} f(\mathbf{x}_i)}{L} \cdot \frac{\nabla f(\mathbf{x}_t)}{\|\nabla f(\mathbf{x}_t)\|_2}$;
    $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t + \mathbf{v}_{t+1}$;
    $t \leftarrow t+1$;
  end
  if $\min_{i=1,\cdots,t} f(\mathbf{x}_i) \leq \frac{\epsilon}{1-\rho}$ *holds* **then** Break ;
end
return $\mathbf{x}^* = \mathbf{x}_{i^*}$ where $i^* \in \arg\min_{i=1,\cdots,n} f(\mathbf{x}_i)$;

Momentum helps evolve toward global optima



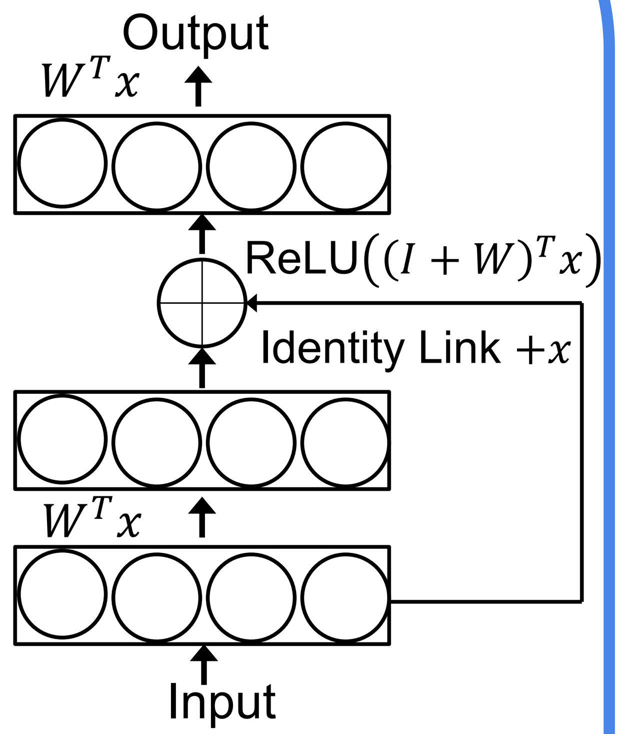
Estimation of Lipschitz Constant L:



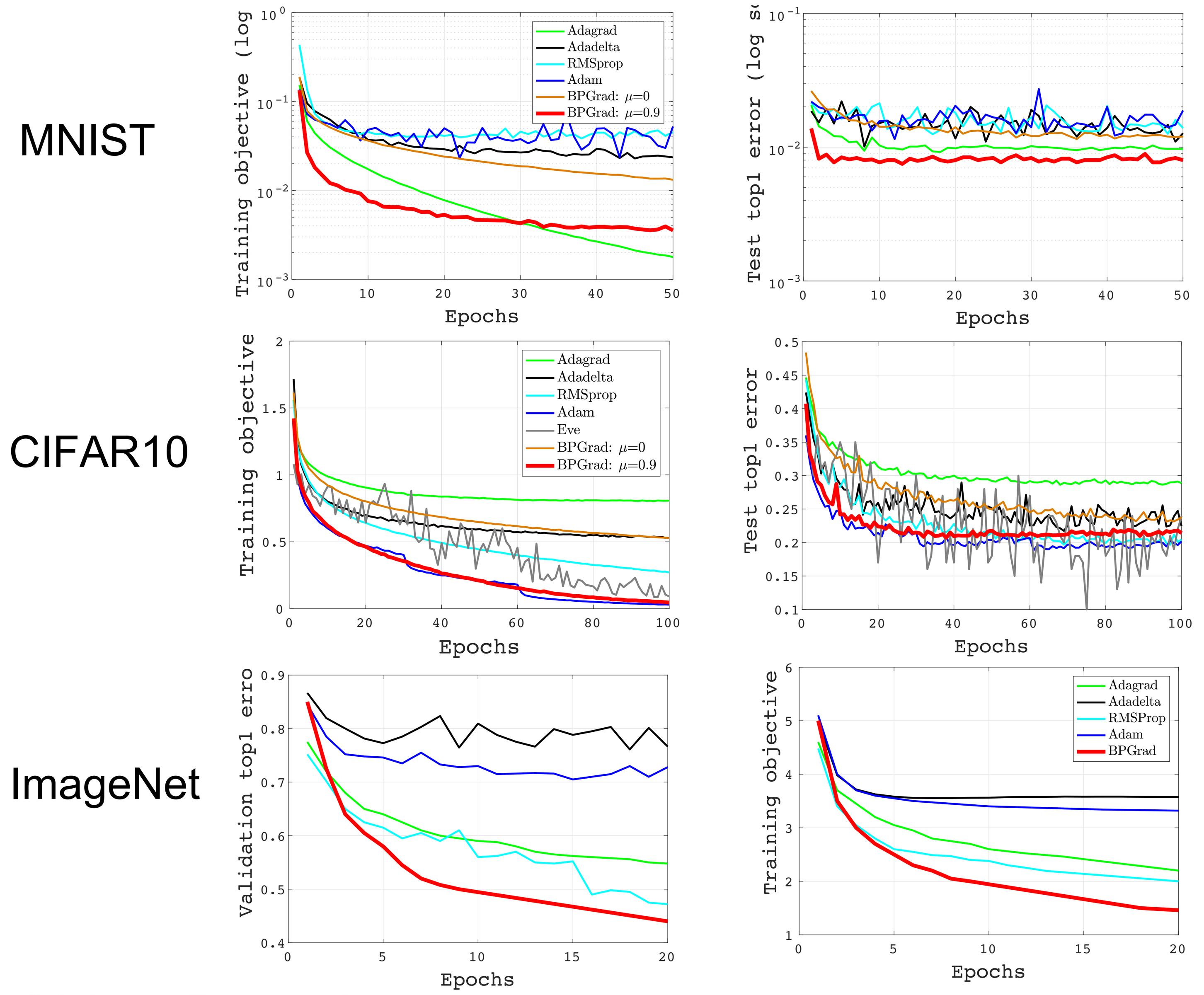
EXPERIMENTS

Global Convergence of BPGrad Solver:

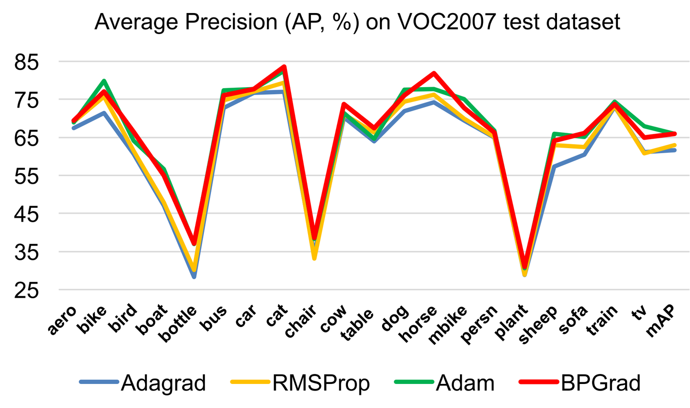
- Numerical test (Li&Yuan, NIPS'17)
  - Two-layer neural network
  - 10,302 parameters
  - On MNIST with 20 epochs and batch size of 200
- SGD vs. BPGrad
  - Momentum coefficient 0.9
  - Best performance
  - Euclidean distance between the solutions is 0.6



Object Recognition:



Object Detection:



Segmentation:

	mean IU	pixel accuracy	mean accuracy	average
Adagrad	60.8	89.5	77.4	75.9
Adadelta	46.6	86.0	54.4	62.3
RMSProp	60.5	**90.2**	71.0	73.9
Adam	50.9	87.2	66.4	68.2
BPGrad	**62.4**	89.8	**79.6**	**77.3**

Table 3. Numerical comparison on semantic segmentation performance (%) using VOC2011 test dataset at the 50-th epoch.

(* denotes equal contributions)