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

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:
\[ B_1 \leq T_{min} \]
\[ B_2 > T_{min} \]

Lipschitz Continuity:
\[ |f(x_i) - f(x_j)| \leq L|x_i - x_j|, \quad x_i, x_j \in X \]

BPGrad Algorithm for Lipschitz functions:
\[ \min_{x \in X} \| x_{t+1} - \left( x_t - \eta_t \nabla f(x_t) \right) \| \leq \eta_t L \]

Any solution inside the circle can be safely removed as pruning

Algorithm 1: BPGrad Algorithm for Deep Learning
Input : objective function f with Lipschitz constant L ≥ 0, precision ε ≥ 0
Output: minimizer x∗
Randomly initialize x₁, t ← 1, ρ ← 0;
while min_{inited} f(xₙ) ≤ ρ do
    while ∃xₙ₊₁ ∈ X satisfies Eq. (5) do
        Compute xₙ₊₁ by solving Eq. (5);
        t ← t + 1;
    end
    Increase ρ such that 0 ≤ ρ < 1 still holds;
    return x* = xₙ, where \( x^* = \arg \min_{inited} f(x) \);