Adaptive Gradient-Informed Lévy Exploration (AGILE): A Heavy-Tailed Adaptive Stochastic Optimizer

Abstract

We propose **Adaptive Gradient-Informed Lévy Exploration (AGILE)**, an optimization algorithm that combines gradient descent, Lévy flight-inspired exploration, and an adaptive annealing schedule. AGILE operates in two phases: a stochastic exploratory phase that allows for non-local jumps guided by gradient information and a deterministic refinement / exploitation phase via classical gradient descent. The method dynamically narrows its search radius and step budget upon improvements, balancing exploration and exploitation across the loss landscape.

1 Algorithm Description

Let $\mathcal{L}(\theta)$ be a differentiable loss function over parameters $\theta \in \mathbb{R}^d$. The goal is to minimize $\mathcal{L}$ using a hybrid strategy:

1.1 Initialization

Initialize model parameters θ_0, a maximum search budget P_0, a damping factor $\alpha \in (0, 1)$, and a Lévy exponent $\mu \in (1, 3]$. Set the best-known solution $\theta_{\text{best}} = \theta_0$ and $\mathcal{L}_{\text{best}} = \mathcal{L}(\theta_0)$.

1.2 Stochastic Exploration Phase

Repeat for at most P_k steps (where k indexes each improvement phase):

1. Compute the gradient: $\mathbf{g} = \nabla_\theta \mathcal{L}(\theta)$.

2. Sample a step size $\eta \sim \text{PowerLaw}(\mu, \eta_{\text{min}}, \eta_{\text{max}})$, where $\mu \in (1, 3]$ and $\eta_{\text{min}}, \eta_{\text{max}}$ define the truncation bounds.

3. Determine the direction:

$$\mathbf{d} = \begin{cases} -\dfrac{\mathbf{g}}{\|\mathbf{g}\|_2 + \epsilon}, & \text{if } \|\mathbf{g}\|_2 > \epsilon \\ \mathbf{u}, & \text{otherwise} \end{cases}$$

   where $\epsilon > 0$ is a small constant and $\mathbf{u} \sim \mathbb{S}^{d-1}$ is a random unit vector sampled uniformly from the unit sphere.

4. Update parameters:

$$\theta \leftarrow \theta + \eta \cdot \mathbf{d}$$

5. Evaluate new loss: $\mathcal{L}_{\text{new}} = \mathcal{L}(\theta)$.

6. If $\mathcal{L}_{\text{new}} < \mathcal{L}_{\text{best}}$, then:

$$\theta_{\text{best}} \leftarrow \theta, \quad \mathcal{L}_{\text{best}} \leftarrow \mathcal{L}_{\text{new}}, \quad P_{k+1} = \alpha \cdot P_k, \quad k \leftarrow k + 1$$

   and reset the local step counter to 0.

This loop continues until no improvement occurs within the current search budget P_k.

1.3 Deterministic Refinement Phase

Once the stochastic phase terminates, the algorithm switches to standard gradient descent from θ_{best} to ensure convergence:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla_\theta \mathcal{L}(\theta_t)$$

with fixed step size η until a stopping condition is met (e.g., gradient norm threshold, maximum iterations, or convergence in loss).

2 Key Features

- **Lévy-Guided Exploration:** Power-law distributed step sizes introduce rare large steps, enhancing the ability to escape poor local minima.
- **Gradient Biasing:** Search steps are guided by the gradient when informative, ensuring a general descent direction while maintaining stochasticity.
- **Adaptive Annealing:** The exploration phase uses a search patience or budget P_k, which decays after each improvement, refining the search space over time.
- **Refinement for Convergence:** A final gradient descent phase further refines the result and guarantees local convergence.

3 Discussion

The algorithm balances exploration and exploitation through a two-phase approach. The exploratory phase benefits from non-local transitions while maintaining directional guidance, enabling wide landscape traversal. The adaptive budget reduction mimics cooling schedules in simulated annealing but is tied to empirical improvement rather than time. Upon stabilization, gradient descent refines the solution. This method is particularly well-suited for nonconvex landscapes with many local minima or rugged geometry.