Escaping *Undesired* Stationary Points in Local Saddle Point Optimization
A Curvature Exploitation Approach

Leonard Adolphs Hadi Daneshmand Aurelien Lucchi Thomas Hofmann

Institute of Machine Learning, ETH Zürich, Switzerland

Saddle Point Problem

$$\min_{\mathbf{x}\in\mathbb{R}^k}\max_{\mathbf{y}\in\mathbb{R}^d} f(\mathbf{x},\mathbf{y})$$

Assumptions

- f smooth but non-convex (non-concave) in $\mathbf{x}$ ($\mathbf{y}$)
- $\nabla^2_{\mathbf{x}} f(\mathbf{x}^*,\mathbf{y}^*)$, $\nabla^2_{\mathbf{y}} f(\mathbf{x}^*,\mathbf{y}^*)$ non-degenerate

Relaxed Objective

Finding a **global** saddle point of the above form is generally **infeasible**. Therefore, we aim for a solution in a local neighbourhood, i.e., a point $(\mathbf{x}^*,\mathbf{y}^*)$ s.t.

$$f(\mathbf{x}^*,\mathbf{y}) \le f(\mathbf{x}^*,\mathbf{y}^*) \le f(\mathbf{x},\mathbf{y}^*) \quad \forall(\mathbf{x},\mathbf{y}) \in \mathcal{K}_\gamma$$

where $\mathcal{K}_\gamma$ is a local neighbourhood around the saddle.

Local Saddle Point Conditions

1. $\nabla f(\mathbf{x}^*,\mathbf{y}^*) = 0$
2. $\nabla^2_{\mathbf{x}} f(\mathbf{x}^*,\mathbf{y}^*) \succ 0$
3. $\nabla^2_{\mathbf{y}} f(\mathbf{x}^*,\mathbf{y}^*) \prec 0$

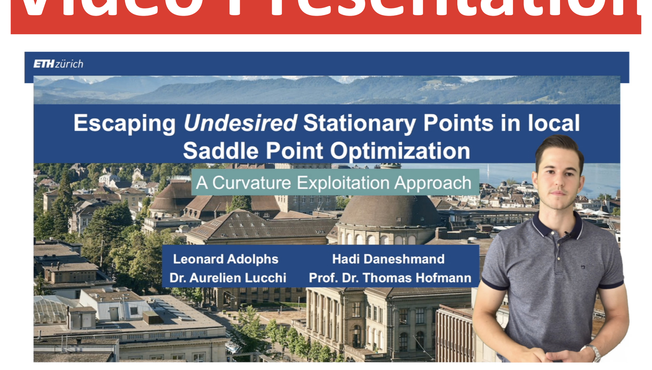
Gradient-Based Optimization

Simultaneously applying Gradient Descent on $\mathbf{x}$ and Gradient Ascent on $\mathbf{y}$:

$$\begin{pmatrix}\mathbf{x}^+\\ \mathbf{y}^+\end{pmatrix} = \begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix} + \eta\begin{pmatrix}-\nabla_{\mathbf{x}} f(\mathbf{x},\mathbf{y})\\ \nabla_{\mathbf{y}} f(\mathbf{x},\mathbf{y})\end{pmatrix}$$

If convergent, it almost surely finds a **stable** stationary point of the gradient dynamics. But not necessarily a solution to the local saddle point problem ...

Video Presentation



Gradient-Based Optimization Does not Solve for Local Saddles

Even if gradient-based optimization converges, we have no (approximate) guarantee of obtaining a solution to the local saddle point problem.

Stability versus Optimality

	local optimality condition	stability condition
Minimization	$\nabla^2_{\mathbf{x}} f(\mathbf{x},\mathbf{y}) \succ 0$ $=$	$\nabla^2_{\mathbf{x}} f(\mathbf{x},\mathbf{y}) \succ 0$
Saddle Point Optimization	$\nabla^2_{\mathbf{x}} f(\mathbf{x},\mathbf{y}) \succ 0$, $\nabla^2_{\mathbf{y}} f(\mathbf{x},\mathbf{y}) \prec 0$ $\neq$	$\lambda\left(\begin{bmatrix}-\nabla^2_{\mathbf{x}} f & -\nabla_{\mathbf{xy}} f\\ \nabla_{\mathbf{yx}} f & \nabla^2_{\mathbf{y}} f\end{bmatrix}\right)$ negative real part

Table 1: Stability versus optimality condition in minimization and saddle point optimization.

$$\min_{x\in\mathbb{R}}\max_{y\in\mathbb{R}}\left[f(x,y) = 2x^2 + y^2 + 4xy + \frac{4}{3}y^3 - \frac{1}{4}y^4\right]$$

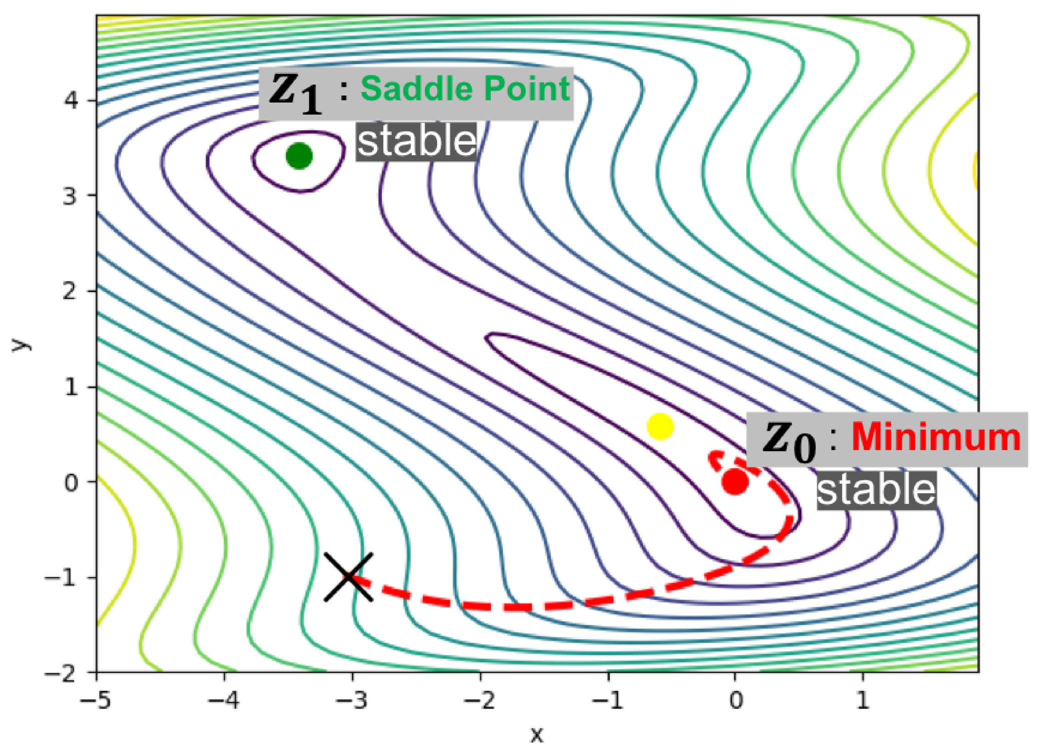


Figure 1: Gradient-based optimization converges to the minimum z_0 rather than the saddle point z_1.

CESP - Curvature Exploitation for the Saddle Point Problem

Intuition

Simple observation: If there is positive (negative) curvature in $\mathbf{x}$-direction ($\mathbf{y}$-direction) then the local saddle point conditions are not met, because $\nabla^2_{\mathbf{x}} f(\mathbf{x},\mathbf{y}) \not\succ 0$ ($\nabla^2_{\mathbf{y}} f(\mathbf{x},\mathbf{y}) \not\prec 0$).

Following negative curvature in $\mathbf{x}$ and positive curvature in $\mathbf{y}$ helps us escape from undesired stable stationary points.

Algorithm

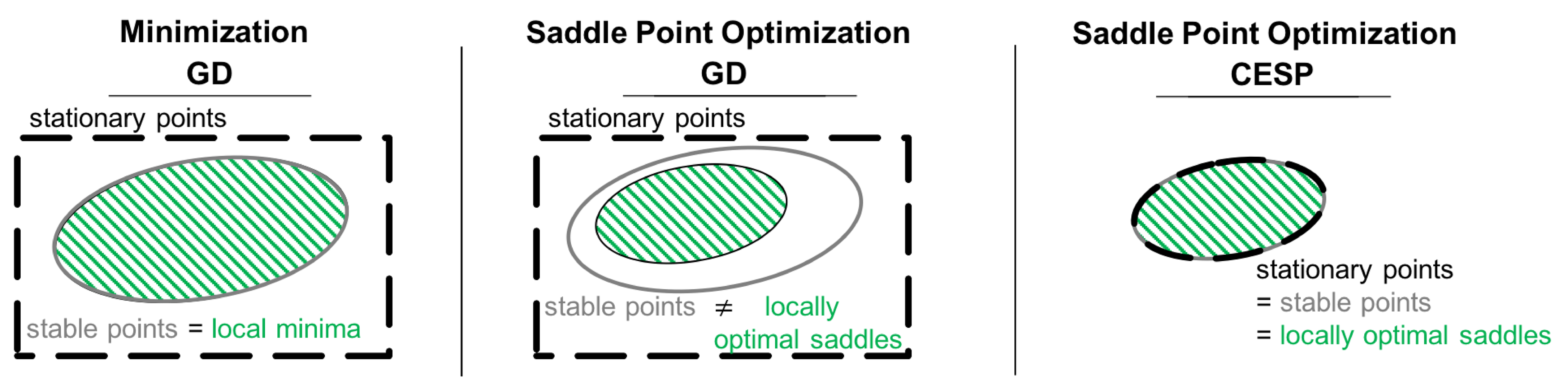
Let $\lambda_{\mathbf{x}}$ ($\lambda_{\mathbf{y}}$) be the minimum (maximum) eigenvalue of $\nabla^2_{\mathbf{x}} f$ ($\nabla^2_{\mathbf{y}} f$) with its associated eigenvector $\mathbf{v}_{\mathbf{x}}$ ($\mathbf{v}_{\mathbf{y}}$) and $\rho_{\mathbf{x}}, \rho_{\mathbf{y}} > 0$ some smoothness parameters.

$$\mathbf{v}_{\mathbf{z}}^{(-)} = 1_{\{\lambda_{\mathbf{x}}<0\}}\frac{\lambda_{\mathbf{x}}}{2\rho_{\mathbf{x}}}\operatorname{sgn}(\mathbf{v}_{\mathbf{x}}^\top\nabla_{\mathbf{x}} f(\mathbf{z}))\mathbf{v}_{\mathbf{x}}$$

$$\mathbf{v}_{\mathbf{z}}^{(+)} = 1_{\{\lambda_{\mathbf{y}}>0\}}\frac{\lambda_{\mathbf{y}}}{2\rho_{\mathbf{y}}}\operatorname{sgn}(\mathbf{v}_{\mathbf{y}}^\top\nabla_{\mathbf{y}} f(\mathbf{z}))\mathbf{v}_{\mathbf{y}}$$

$$\begin{pmatrix}\mathbf{x}^+\\ \mathbf{y}^+\end{pmatrix} = \begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix} + \eta\begin{pmatrix}-\nabla_{\mathbf{x}} f(\mathbf{x},\mathbf{y})\\ \nabla_{\mathbf{y}} f(\mathbf{x},\mathbf{y})\end{pmatrix} + \begin{pmatrix}\mathbf{v}_{\mathbf{z}}^{(-)}\\ \mathbf{v}_{\mathbf{z}}^{(+)}\end{pmatrix}$$

Theoretical Guarantees



Toy Example

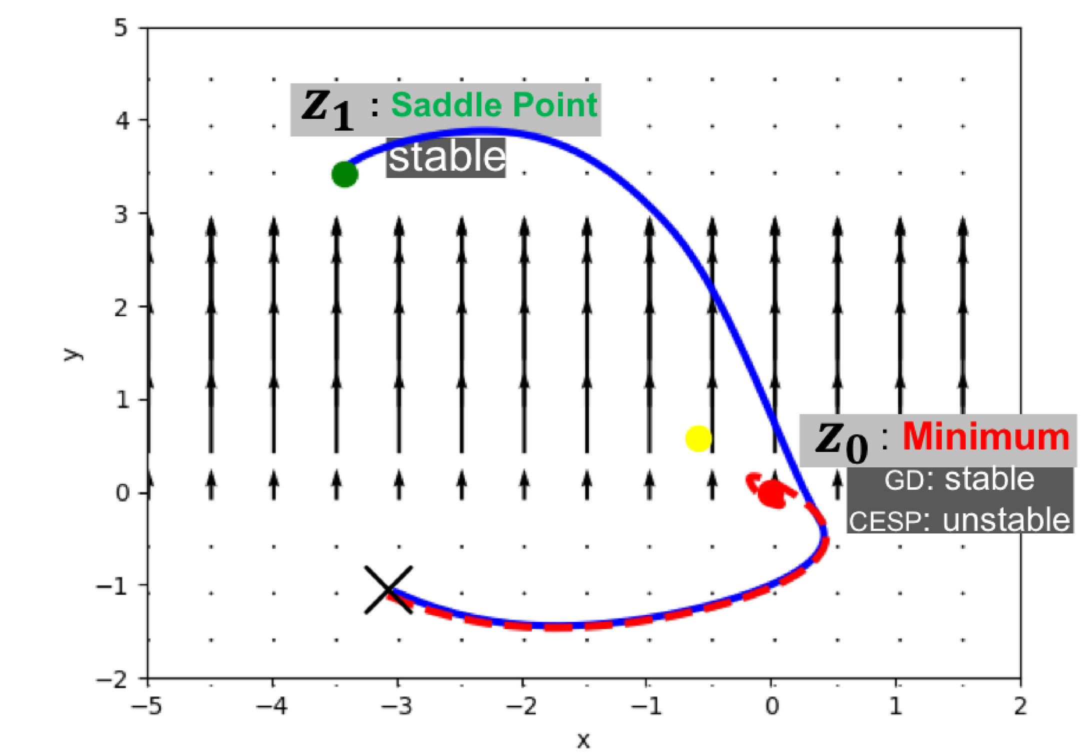


Figure 2: CESP (blue) converges to the saddle point as opposed to gradient-based optimization (red). The vector field shows the *extreme curvature vector* $(\mathbf{v}_{\mathbf{z}}^{(-)}, \mathbf{v}_{\mathbf{z}}^{(+)})$.

CESP in the Real World

- Theoretical guarantees hold also for *transformed* gradient steps, e.g. **ADAGRAD**.
- Cheap implementation with Power Iterations using only **Hessian-vector Products**.
- Tested on (small) **Generative Adversarial Nets**

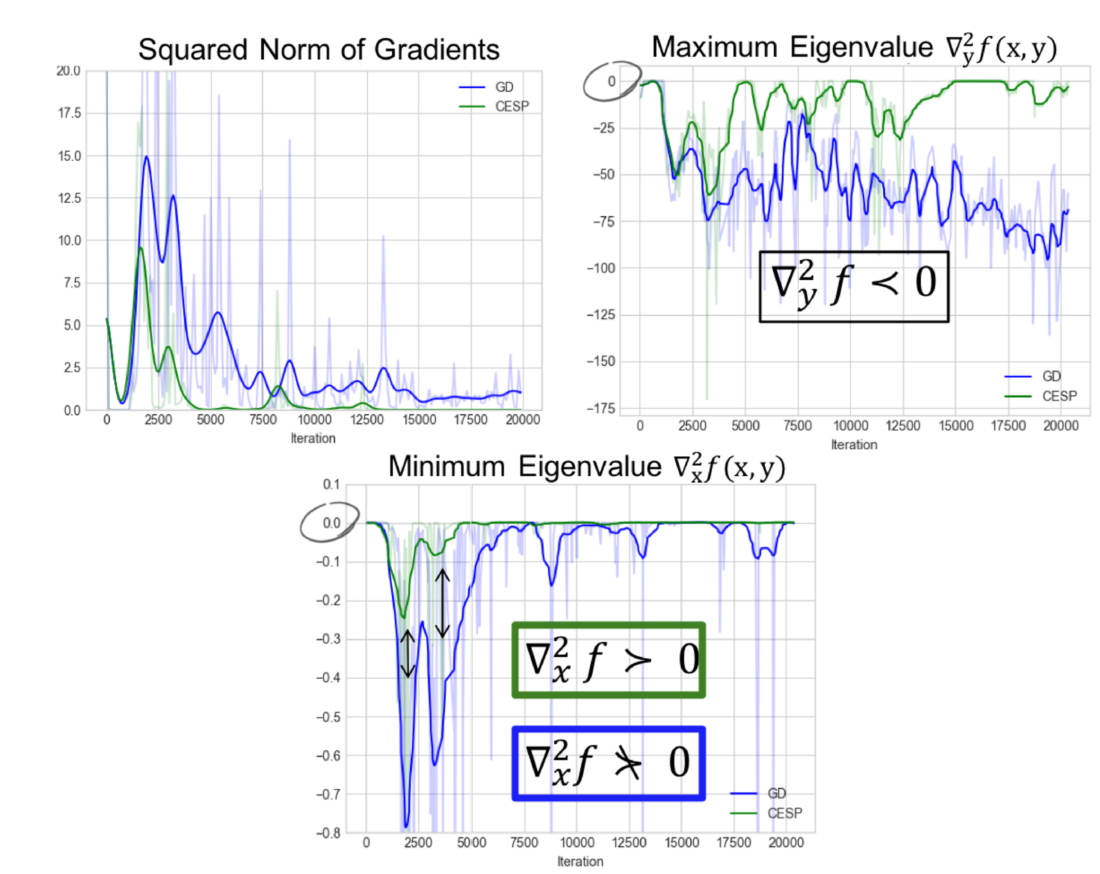


Figure 3: CESP drives convergent solution to the desired *min-max* structure.

- Many more possible applications, e.g. **Robust Optimization** for empirical risk minimization