Using Bifurcations for Diversity in Differentiable Games

Beyond First-order Methods in ML Workshop, ICML 2021

Jonathan Lorraine[1,2,3], Jack Parker-Holder[1,4], Paul Vicol[2,3], Aldo Pacchiano[5], Luke Metz[6], Tal Kachman[7], Jakob Foerster[1]

July 24th, 2021

Facebook AI Research[1], University of Toronto[2], Vector Institute[3], University of Oxford[4], UC Berkeley[5], Google Brain[6], Radboud University[7]

Introduction

- Finding different types of solutions has been useful in minimization - ex., shape vs. texture solutions for CNNs.

Introduction

- Finding different types of solutions has been useful in minimization - ex., shape vs. texture solutions for CNNs.
- Differentiable games generalize single-objective minimization:

$$\boldsymbol{\theta}_A^* \in \arg\min_{\boldsymbol{\theta}_A} \mathcal{L}_A(\boldsymbol{\theta}_A, \boldsymbol{\theta}_B^*),$$
$$\boldsymbol{\theta}_B^* \in \arg\min_{\boldsymbol{\theta}_B} \mathcal{L}_B(\boldsymbol{\theta}_A^*, \boldsymbol{\theta}_B)$$

Introduction

- Finding different types of solutions has been useful in minimization - ex., shape vs. texture solutions for CNNs.
- Differentiable games generalize single-objective minimization:

$$\boldsymbol{\theta}_A^* \in \arg\min_{\boldsymbol{\theta}_A} \mathcal{L}_A(\boldsymbol{\theta}_A, \boldsymbol{\theta}_B^*),$$
$$\boldsymbol{\theta}_B^* \in \arg\min_{\boldsymbol{\theta}_B} \mathcal{L}_B(\boldsymbol{\theta}_A^*, \boldsymbol{\theta}_B)$$

- Games are increasingly important in ML – ex., GANs, hyperparam opt., self-play, meta-learning, adversarial examples, numerous others.

Introduction

- Finding different types of solutions has been useful in minimization - ex., shape vs. texture solutions for CNNs.
- Differentiable games generalize single-objective minimization:

$$\boldsymbol{\theta}_A^* \in \arg\min_{\boldsymbol{\theta}_A} \mathcal{L}_A(\boldsymbol{\theta}_A, \boldsymbol{\theta}_B^*),$$
$$\boldsymbol{\theta}_B^* \in \arg\min_{\boldsymbol{\theta}_B} \mathcal{L}_B(\boldsymbol{\theta}_A^*, \boldsymbol{\theta}_B)$$

- Games are increasingly important in ML – ex., GANs, hyperparam opt., self-play, meta-learning, adversarial examples, numerous others.
- **Goal:** Want to find diverse solutions in differentiable games – ex., where players work together or battle each-other.

Background

- Ridge rider [1] finds diverse solutions in single-objective minimization by following Hessian's negative EVals at saddles.

Background

- Ridge rider [1] finds diverse solutions in single-objective minimization by following Hessian's negative EVals at saddles.
- Bifurcations are areas where small changes cause solution differences. Saddles are a key bifurcation in conservative systems from following gradients.

Background

- Ridge rider [1] finds diverse solutions in single-objective minimization by following Hessian's negative EVals at saddles.
- Bifurcations are areas where small changes cause solution differences. Saddles are a key bifurcation in conservative systems from following gradients.
- Following gradients in games is a non-conservative system, so more solution and bifurcation types.

Background

- Ridge rider [1] finds diverse solutions in single-objective minimization by following Hessian's negative EVals at saddles.
- Bifurcations are areas where small changes cause solution differences. Saddles are a key bifurcation in conservative systems from following gradients.
- Following gradients in games is a non-conservative system, so more solution and bifurcation types.
- Because, Hessian's generalization for games – i.e., the *Game Hessian* – may have complex EVals from lack of symmetry.

$$\overbrace{\hat{\mathcal{H}}}^{\text{Game Hessian}} = \begin{bmatrix} \text{Player } A \text{ Hessian} \nabla^2_{\boldsymbol{\theta}_A} \mathcal{L}_A & \nabla_{\boldsymbol{\theta}_A} \nabla_{\boldsymbol{\theta}_B} \mathcal{L}_A \\ \nabla_{\boldsymbol{\theta}_B} \nabla_{\boldsymbol{\theta}_A} \mathcal{L}_B{}^\top & \text{Player } B \text{ Hessian} \nabla^2_{\boldsymbol{\theta}_B} \mathcal{L}_B \end{bmatrix}$$

Our Method – Game Ridge Rider (GRR)

We generalize Ridge Rider to games with the following:

Our Method – Game Ridge Rider (GRR)

We generalize Ridge Rider to games with the following:

- Complex EVals may have EVecs with complex entries. We use an EVec selection for conjugate pairs that has all real entries, so we can follow it.

Our Method – Game Ridge Rider (GRR)

We generalize Ridge Rider to games with the following:

- Complex EVals may have EVecs with complex entries. We use an EVec selection for conjugate pairs that has all real entries, so we can follow it.
- We detect and allow for branching at new types of bifurcations – ex., Hopf where $\approx \Re$ EVal crosses 0.

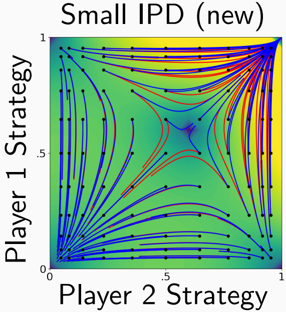
Our Method – Game Ridge Rider (GRR)

We generalize Ridge Rider to games with the following:

- Complex EVals may have EVecs with complex entries. We use an EVec selection for conjugate pairs that has all real entries, so we can follow it.
- We detect and allow for branching at new types of bifurcations – ex., Hopf where $\approx \Re$ EVal crosses 0.
- We apply an arbitrary optimization algorithm after branching.

New Toy Problems

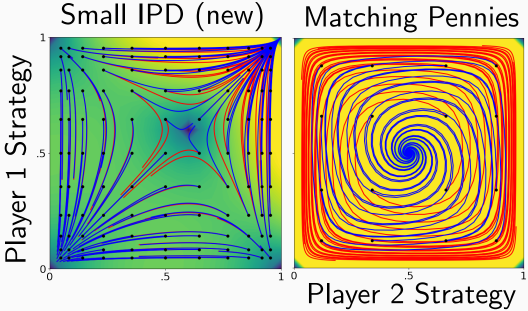
Following the gradient in red, LOLA [2] in blue



- Small IPD is a 2 param. Iterated Prisoner's Dilemma with TT and DD solutions, but only real EVals.

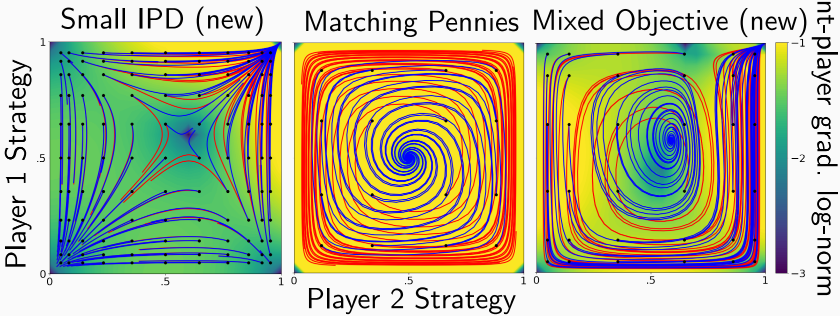
New Toy Problems

Following the gradient in red, LOLA [2] in blue



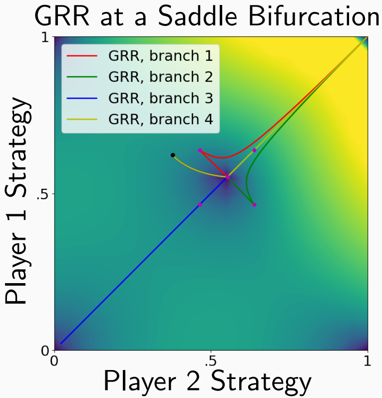
- Small IPD is a 2 param. Iterated Prisoner's Dilemma with TT and DD solutions, but only real EVals.
- Matching Pennies is a 2 param. rock-paper-scissors with imaginary EVals, but only 1 solution.

Following the gradient in red, LOLA [2] in blue



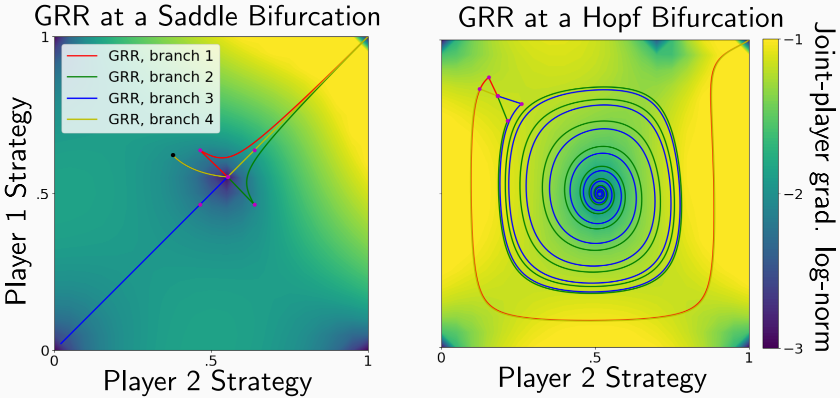
- Small IPD is a 2 param. Iterated Prisoner's Dilemma with TT and DD solutions, but only real EVals.
- Matching Pennies is a 2 param. rock-paper-scissors with imaginary EVals, but only 1 solution.
- Mixing these gives a 2 param. problem like the full IPD with multiple solutions, complex EVals, and a Hopf bifurcation.

Applying our Method on Toy Problems



- For both the small IPD (left) and mixed objective (right) our method – Game Ridge Rider (GRR) – finds all solutions.

Applying our Method on Toy Problems



- For both the small IPD (left) and mixed objective (right) our method – Game Ridge Rider (GRR) – finds all solutions.

Finding Diverse Solutions in the Iterated Prisoners Dilemma

	Solution Mode	
Search Strategy	Cooperate	Defect
$\times$20 Random init and LOLA [2]	✓	×
$\times$20 Random init and follow grad.	×	✓

- Randomly initializing then applying a training method only finds 1 solution mode.

Finding Diverse Solutions in the Iterated Prisoners Dilemma

	Solution Mode	
Search Strategy	Cooperate	Defect
$\times$20 Random init and LOLA [2]	✓	×
$\times$20 Random init and follow grad.	×	✓
Ours: Max entropy saddle and branch	✓	✓

- Randomly initializing then applying a training method only finds 1 solution mode.
- Our method finds both solution modes.

Finding Diverse Solutions in the Iterated Prisoners Dilemma

	Solution Mode	
Search Strategy	Cooperate	Defect
$\times$20 Random init and LOLA [2]	✓	×
$\times$20 Random init and follow grad.	×	✓
Ours: Max entropy saddle and branch	✓	✓
Random init and branch	×	✓

- Randomly initializing then applying a training method only finds 1 solution mode.
- Our method finds both solution modes.
- If we don't start at a saddle, then branching doesn't affect the solution.

Takeaways

- Differentiable games generalize single-objective minimization, but with non-conservative dynamics from complex EVals.

Takeaways

- Differentiable games generalize single-objective minimization, but with non-conservative dynamics from complex EVals.
- We can view methods for diverse solutions in single-objective minimization – i.e., Ridge Rider (RR) – as finding bifurcations in conservative systems and branching.

Takeaways

- Differentiable games generalize single-objective minimization, but with non-conservative dynamics from complex EVals.
- We can view methods for diverse solutions in single-objective minimization – i.e., Ridge Rider (RR) – as finding bifurcations in conservative systems and branching.
- This viewpoint allows usage of tools from dynamical systems to generalize RR to non-conservative systems.

Thanks!

Jonathan Lorraine

Jack Parker-Holder

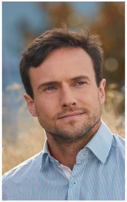
Paul Vicol

Aldo Pacchiano

Luke Metz

Tal Kachman

Jakob Foerster



References

[1] Jack Parker-Holder, Luke Metz, Cinjon Resnick, Hengyuan Hu, Adam Lerer, Alistair Letcher, Alexander Peysakhovich, Aldo Pacchiano, and Jakob Foerster. Ridge rider: Finding diverse solutions by following eigenvectors of the hessian. In *Advances in Neural Information Processing Systems*, volume 33, pages 753–765, 2020.

[2] Jakob Foerster, Richard Y Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness. In *International Conference on Autonomous Agents and MultiAgent Systems*, pages 122–130, 2018.

Extra Slides

Proposed Algorithm

- Check out the paper for more details.

Algorithm 1 Game Ridge Rider (GRR)–red modifications

Input: $\boldsymbol{\omega}^{\text{Saddle}}$, α, ChooseFromArchive, GetRidges,
EndRide, Optimize, UpdateRidge
$\mathcal{A} = \text{GetRidges}(\boldsymbol{\omega}^{\text{Saddle}})$ # Init. Archive
while Archive $\mathcal{A}$ non-empty **do**
  $j, \mathcal{A} = \text{ChooseFromArchive}(\mathcal{A})$
  $(\boldsymbol{\omega}^j, e_j, \lambda_j) = \mathcal{A}_j$
  while $\text{EndRide}(\boldsymbol{\omega}^j, e_j, \lambda_j)$ not True **do**
    $\boldsymbol{\omega}^i \leftarrow \boldsymbol{\omega}^j - \alpha e_j$ # Step along the ridge e_j
    $e_j, \lambda_j = \text{UpdateRidge}(\boldsymbol{\omega}^j, e_j, \lambda_j)$
  $\boldsymbol{\omega}^j = \text{Optimize}(\boldsymbol{\omega}^j)$
  $\mathcal{A} = \mathcal{A} \cup \text{GetRidges}(\boldsymbol{\omega}^j)$ # Add new ridges