Dynamic Game Theoretic Neural Optimizer

Guan-Horng Liu, Tianrong Chen, Evangelos A. Theodorou

Georgia Institute of Technology

{ghliu, tianrong.chen, evangelos.theodorou}@gatech.edu

ICML 2021 (Long Talk)

Dynamic Game Theoretic Neural Optimizer

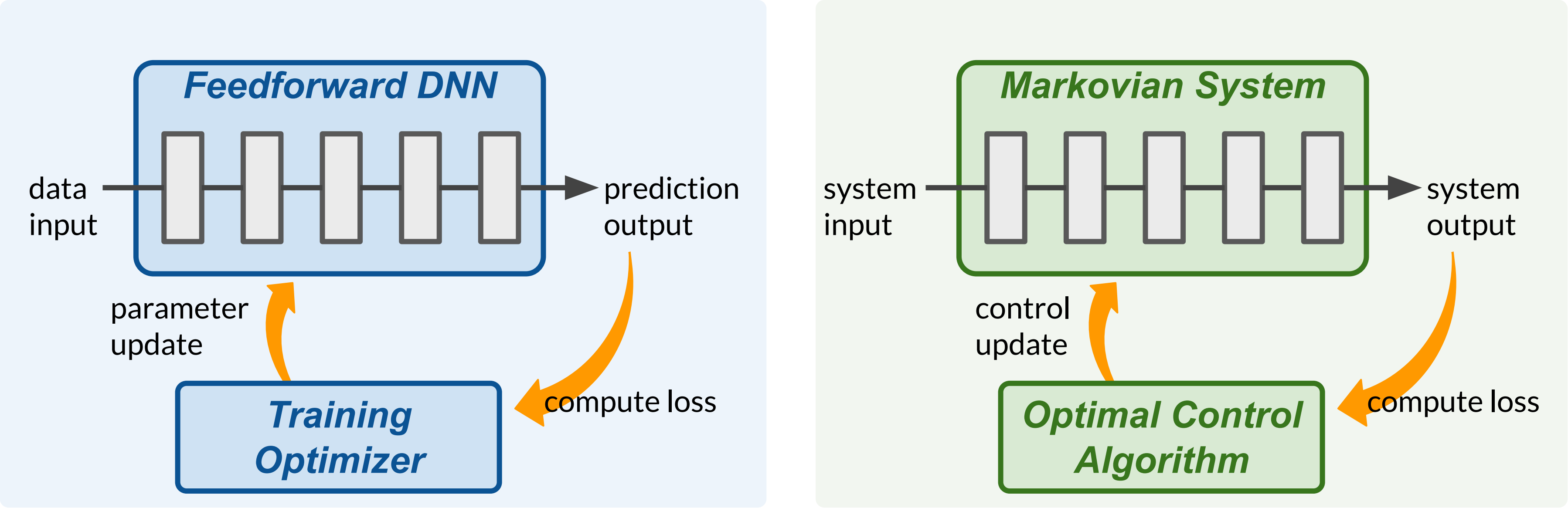
A new class of optimizers for training DNNs that features

- ***<u>Dynamic</u>****al system and optimal control perspective
  - Deep learning theory (*Weinan et al., 2018*; *Hu et al., 2019*; *Liu & Theodorou, 2019*)
  - Computational acceleration (*Gunther et al., 2020*; *Zhang et al., 2019*)
  - Optimal-control-inspired training methods (*Liu et al., 2021*; *Li & Hao, 2018*; *Li et al., 2017*)

- ***<u>Game Theoretic</u>*** interpretation
  - Generalizes OC-inspired methods to a larger network class (*e.g.* ResNet)
  - Novel algorithmic characterization from Nash equilibria standpoint
  - Enhance training with game-based applications (*e.g.* bandit analysis, robust control)

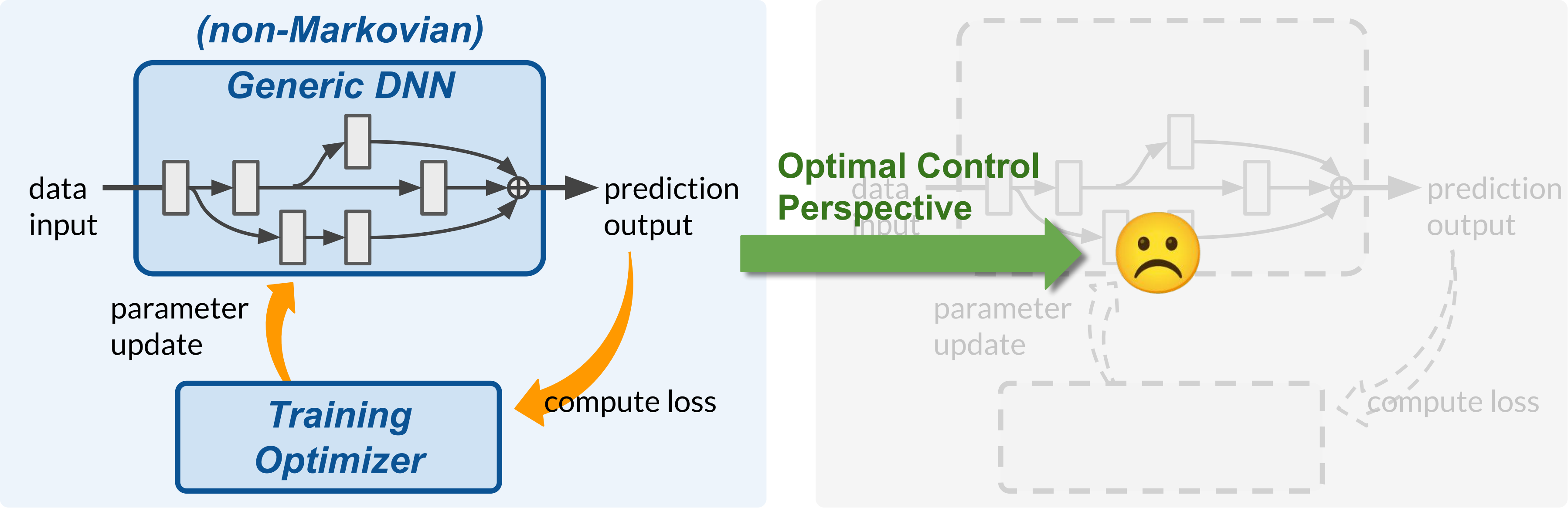
- Competitive performance on image classification while being computationally efficient and numerical stabler

Optimal Control Perspective



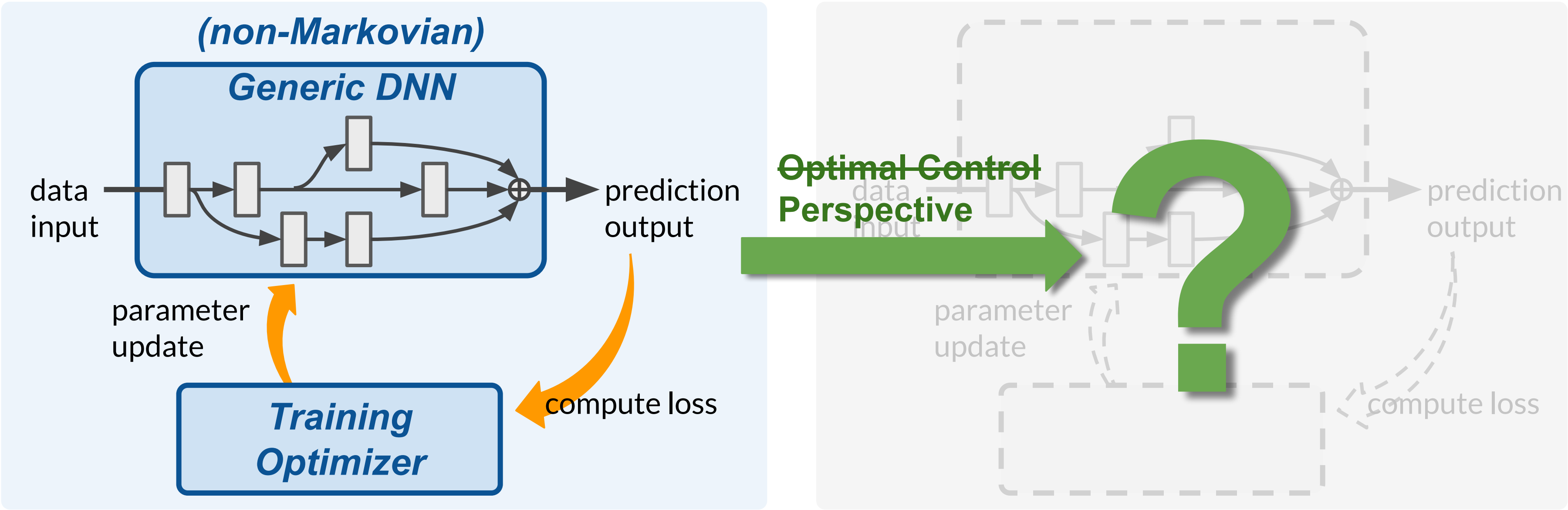
- Treat the *propagation of each layer* as a distinct *time step of a nonlinear dynamical system*.
- Interpret *layer parameter* as the *time-varying control* (*Weinan et al., 2018*; *Liu & Theodorou, 2019*).
- Rigorous optimization theory and new OC-inspired training method (*Liu et al., 2021*).

Limitation of Optimal Control Perspective



- OC-inspired methods, by construction, rely on Markovian interpretation between DNNs and dynamical systems. This poses difficulties for training modern networks (*e.g.* ResNet, Inception) that heavily rely on *non-Markovian dependencies* between layers.

Limitation of Optimal Control Perspective



- How should the Optimal Control perspective be modified in these cases?
- Do we gain any new optimization insight from such a generalization (if any)?
- Can efficient computation be made possible?

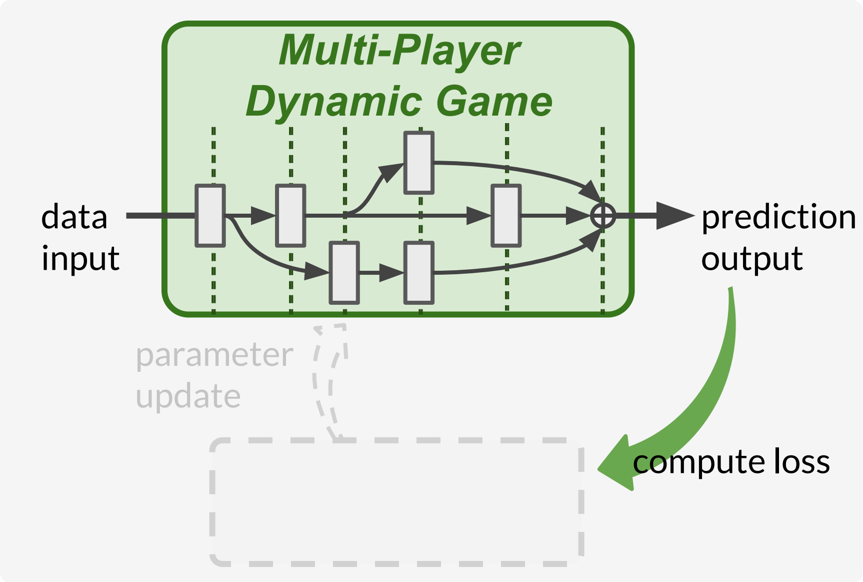
Multi-Player Dynamic Game

In a discrete-time N-player T-stage dynamic game, Player n commits to the action $\theta_{t,n}$ at each stage t and seeks to minimize

$$\boxed{L_n(\bar{\theta}_n; \bar{\theta}_{\neg n}) := \left[\phi_n(\boldsymbol{x}_T) + \sum_{t=0}^{T-1} \ell_{t,n}(\theta_{t,1}, \cdots, \theta_{t,N}) \right]}$$

$$s.t.\ \boldsymbol{x}_{t+1} = F_t(\boldsymbol{x}_t, \theta_{t,1}, \cdots, \theta_{t,N})$$

where $\bar{\theta}_n := \{\theta_{t,n} : t \in [T]\}$ and $\neg n := \{i \in [N] : i \neq n\}$.



Terminology Mapping

DNN		Multi-Player Game
Layer		Player
Computation Order		Stage Sequence (t)
Parameter of layer indexed by (t, n)	$\theta_{t,n}$	Action at stage t for Player n
Training Loss	L_n	Accumulated Cost (Payoff)

Multi-Player Dynamic Game

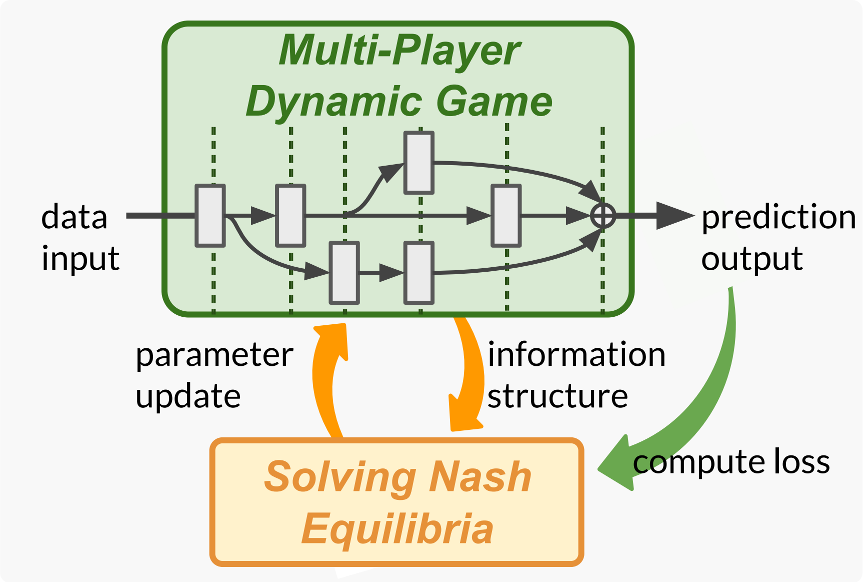
- *Nash Equilibria* $\{(\bar{\theta}_1^*, \cdots, \bar{\theta}_N^*)\}$ is a set of stationary points where no players has the incentive to deviate, *i.e.*

$$L_n(\bar{\theta}_n^*; \bar{\theta}_{\neg n}^*) \le L_n(\bar{\theta}_n; \bar{\theta}_{\neg n}^*),$$

where $\bar{\theta}_n^* \equiv \bar{\theta}_n^*(\eta_{t,n})$.

- *Information structure* $\eta_{t,n}$ is a set of information available to Player n at stage t for making the action $\theta_{t,n}$.

- Different information structures
  - ➩ Different Nash equilibria & optimality conditions
  - ➩ Different classes of training methods



$\bar{\theta}_n$: Collective actions of Player n over all stages
$\neg n$: Indices of all players except Player n

Characterize Optimizers via Information Structure

3 information structures and their Nash equilibria

Open-Loop Nash Equilibria (OLNE)

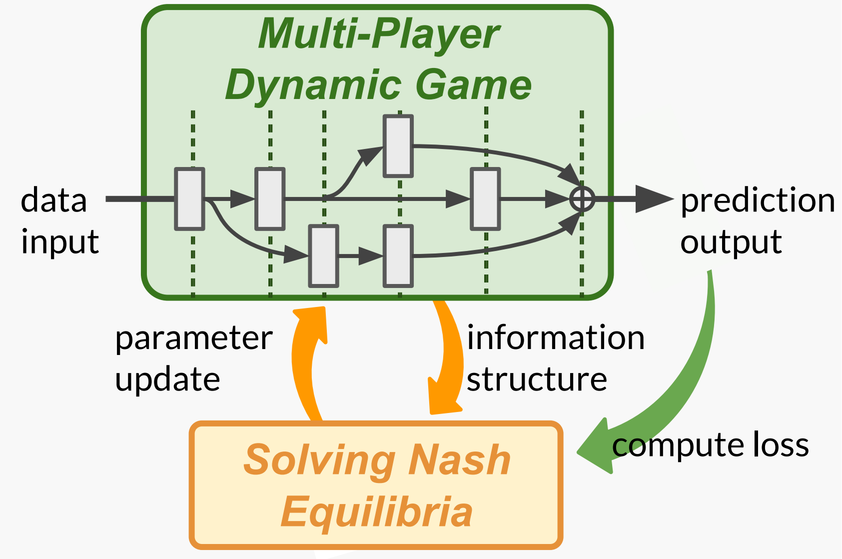
$$\eta_{t,n} = \{\boldsymbol{x}_0\}$$

Feedback Nash Equilibria (FNE)

$$\eta_{t,n} = \{\boldsymbol{x}_s : s \leq t\}$$

Cooperative/Group Rationality (GR)

$$\eta_{t,n} = \{\boldsymbol{x}_s, \theta^*_{t,\neg n} : s \leq t\}$$



$\bar{\theta}_n$: Collective actions of Player n over all stages

$\neg n$: Indices of all players except Player n

Characterize Optimizers via Information Structure

Connection between OLNE and baseline methods.

Open-Loop Nash Equilibria (OLNE)

$$\eta_{t,n} = \{\boldsymbol{x}_0\}$$

Feedback Nash Equilibria (FNE)

$$\eta_{t,n} = \{\boldsymbol{x}_s : s \le t\}$$

Cooperative/Group Rationality (GR)

$$\eta_{t,n} = \{\boldsymbol{x}_s, \theta^*_{t,\neg n} : s \le t\}$$

Proposition (informal; see our paper)

Iteratively solving the optimality condition of *OLNE* recovers the descent direction of standard training methods, *e.g. SGD, RMSprop, Adam, KFAC*, etc.

$\bar{\theta}_n$: Collective actions of Player n over all stages
$\neg n$: Indices of all players except Player n

Characterize Optimizers via Information Structure

DGNOpt iteratively solves FNE or GR.

Open-Loop Nash Equilibria (OLNE)

$$\eta_{t,n} = \{\boldsymbol{x}_0\}$$

Feedback Nash Equilibria (FNE)

$$\eta_{t,n} = \{\boldsymbol{x}_s : s \leq t\}$$

Cooperative/Group Rationality (GR)

$$\eta_{t,n} = \{\boldsymbol{x}_s, \theta^*_{t,\neg n} : s \leq t\}$$

Standard training methods (SGD, Adam, KFAC...)

- Minimal information structure (OLNE)
- *Hamiltonian* optimality condition
- *Vector-form* parameter update

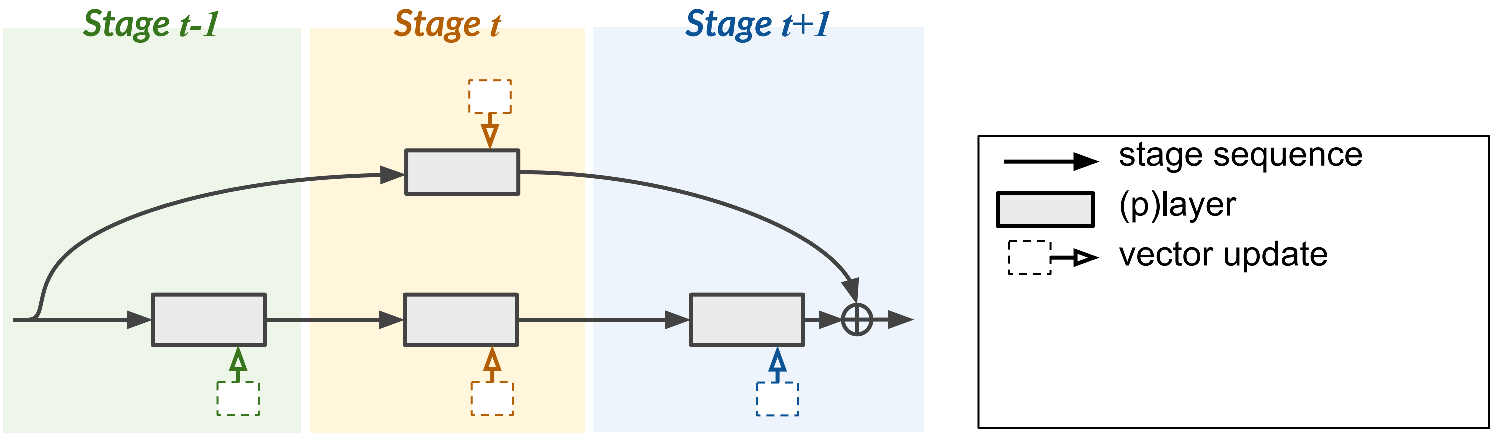
Dynamic Game-theoretic Neural Optimizer

- Richer information structures (FNE or GR)
- *Bellman* optimality condition
- *Feedback* parameter update

$\bar{\theta}_n$: Collective actions of Player n over all stages
$\neg n$: Indices of all players except Player n

Standard Vector *vs.* DGNOpt Feedback Update

- Computation graph of parameter update (using *OLNE*)



- Formula of parameter update of Player n at stage t (using *OLNE*)

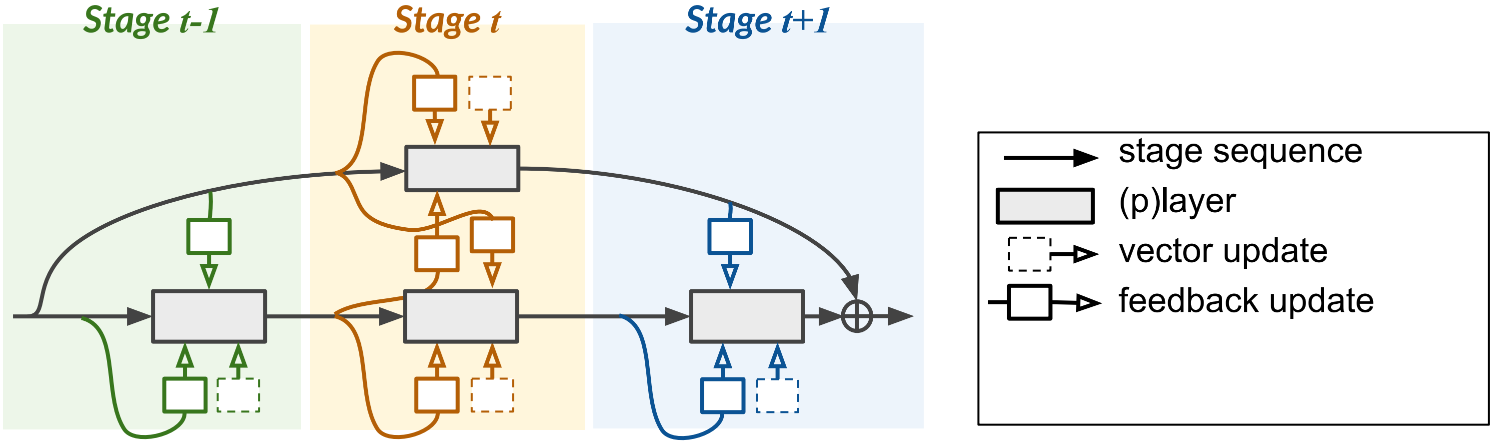
$$\theta_{t,n} \leftarrow \theta_{t,n} + \delta\theta_{t,n}$$

Open-Loop Nash Equilibria (OLNE)

$$\eta_{t,n} = \{\boldsymbol{x}_0\}$$

Standard Vector *vs.* DGNOpt Feedback Update

- Computation graph of parameter update (using *FNE*)



- Formula of parameter update of Player n at stage t (using *FNE*)

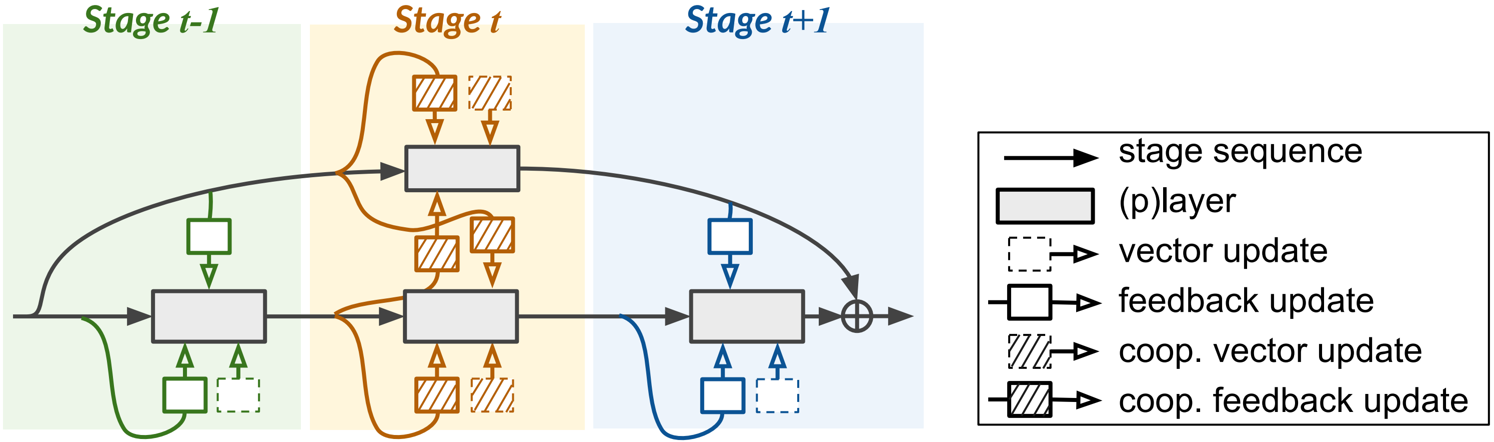
$$\theta_{t,n} \leftarrow \theta_{t,n} + \delta\theta_{t,n} + \mathbf{K}_{t,n}(\delta x_t)$$

Feedback Nash Equilibria (FNE)

$$\eta_{t,n} = \{\boldsymbol{x}_s : s \leq t\}$$

Standard Vector *vs.* DGNOpt Feedback Update

- Computation graph of parameter update (using <u>*GR*</u>)



- Formula of parameter update of Player n at stage t (using <u>*GR*</u>)

$$\theta_{t,n} \leftarrow \theta_{t,n} + \widetilde{\delta\theta}_{t,n}(\delta\theta_{t,\neg n}) + \widetilde{\mathbf{K}}_{t,n}(\delta x_t, \mathbf{K}_{t,\neg n})$$

Cooperative/Group Rationality (GR)

$$\eta_{t,n} = \{\boldsymbol{x}_s, \theta^*_{t,\neg n} : s \leq t\}$$

Benefit of Richer Information & Feedback Update

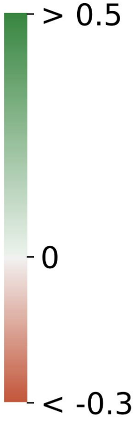
- Improve accuracy (%) of best-tuned baselines across image classification datasets.
(*i.e.* enlarge information structure from OLNE to FNE/GR)

residual nets

	SGD	RMSprop	Adam	EKFAC
MNIST	+0.05	-0.02	+0.05	-0.01
SVHN	+1.32	+0.06	+0.02	+1.16
CIFAR10	+0.35	+0.68	+0.19	+0.09
CIFAR100	+0.28	+0.26	+0.23	+0.29

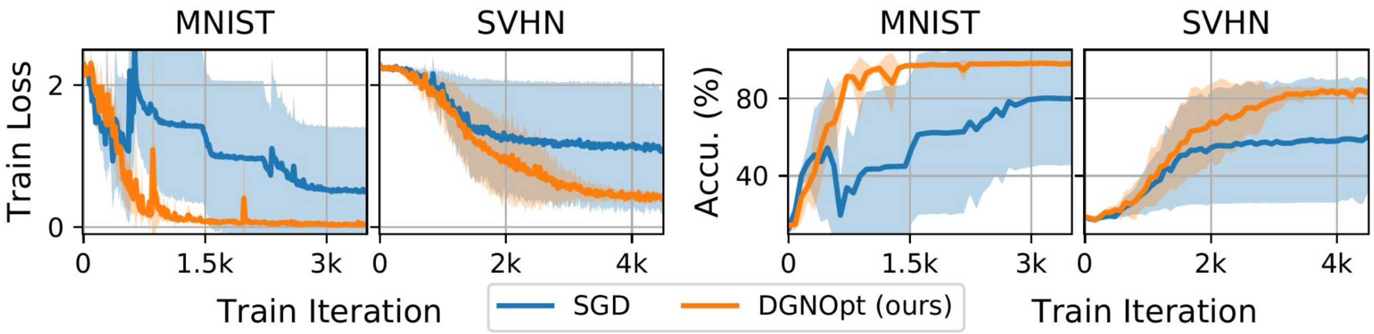
Inception nets

	SGD	RMSprop	Adam	EKFAC
MNIST	+0.07	+0.10	+0.05	+0.03
SVHN	+0.26	+0.10	+0.19	+0.05
CIFAR10	+0.42	+0.21	+0.25	+0.18



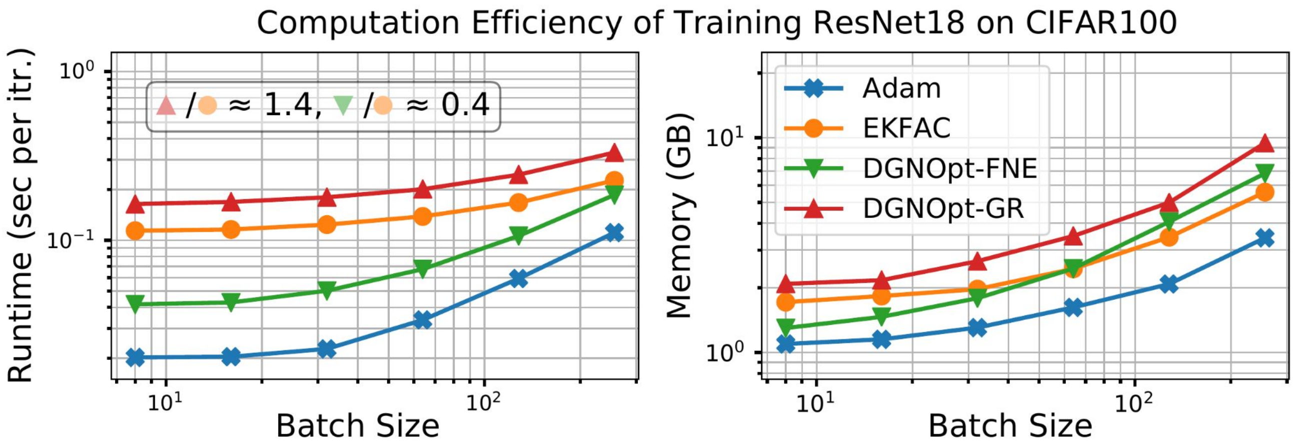
Benefit of Richer Information & Feedback Update

- Improve accuracy (%) of best-tuned baselines across image classification datasets.
- Superior numerical stability when using unstable hyper-parameter (*e.g.* large LR).
  - ➪ Feedback compensates internal disturbance and stabilizes propagation.



Benefit of Richer Information & Feedback Update

- Improve accuracy (%) of best-tuned baselines across image classification datasets.
- Superior numerical stability when using unstable hyper-parameter (*e.g.* large LR).
- Computational efficient compared to second-order baseline.

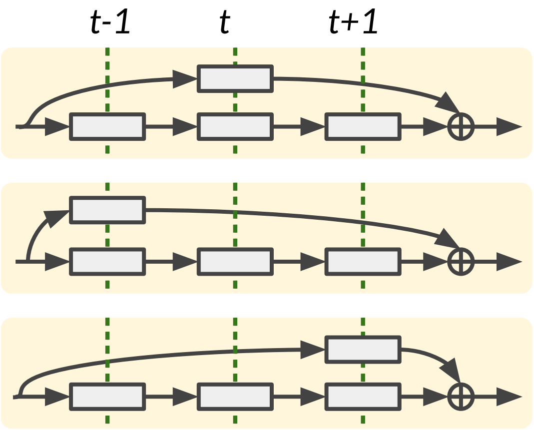


Game-Theoretic Applications

- Training process of DGNOpt exhibits ambiguity.

Algorithm DGNOpt

Convert DNN to multi-player game.
repeat
  Sample data, forward pass, compute loss.
  Solve Nash equilibria with DGDNpt.
until converges



- Different alignment strategy yields different game structure.
  - ➪ What is the *optimal alignment strategy*?
  - ➪ Can we adapt the best-estimated alignment throughout training?

Game-Theoretic Applications

- Integrate DGNOpt with multi-armed bandit (MAB)

Algorithm DGNOpt with MAB

Initialize MAB.
repeat
Select an alignment m from MAB. *(pull an arm from MAB)*
Convert DNN to multi-player game based on m.
Sample data, forward pass, compute loss.
Solve Nash equilibria with DGDNpt.
Compute accuracy and update MAB. *(observe reward of this round, update MAB)*
until converges

Game-Theoretic Applications

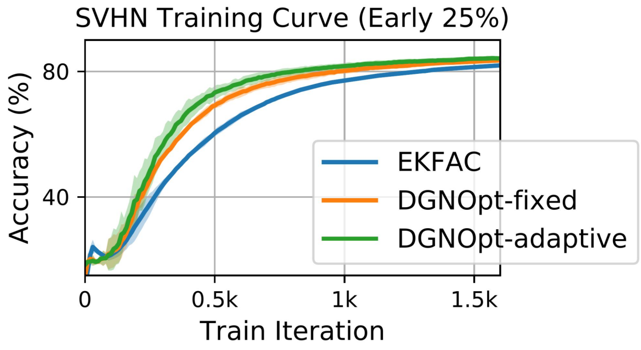
- Integrate DGNOpt with multi-armed bandit (MAB)

Algorithm DGNOpt with MAB

Initialize MAB.
repeat
  Select an alignment m from MAB.
  Convert DNN to multi-player game based on m.
  Sample data, forward pass, compute loss.
  Solve Nash equilibria with DGDNpt.
  Compute accuracy and update MAB.
until converges

Table. Training result (accuracy %)

Dataset	EKFAC	DGNOpt + Aligning Strategy fixed	random	adaptive
SVHN	87.49	88.20	88.12	**88.33**
CIFAR10	84.67	85.20	85.27	**85.65**



Conclusion

Dynamic **G**ame-theoretic **N**eural **Opt**imizer (**DGNOpt**) is a new class of training optimizers that

- advances the dynamical system methodology.
- introduces riguous game-theoretic analysis. (e.g. information structure, Nash equilibria)
- generalizes prior OC-inspired methods to accept generic (non-Markovian) DNNs.
- enables game-related applications. (e.g. bandit, robust control)
- strengthens Optimal Control as a principle tool of analyzing deep learning optimization.

Paper

Poster

Reference

Weinan et al., 2018, "A mean-field optimal control formulation of deep learning."
Hu et al., 2019, "Mean-field langevin system, optimal control and deep neural networks."
Liu & Theodorou, 2019, "Deep learning theory review: An optimal control and dynamical systems perspective."
Gunther et al., 2020, "Layer-parallel training of deep residual neural networks."
Zhang et al., 2019, "You only propagate once: Accelerating adversarial training via maximal principle."
Liu et al., 2021, "DDPNOpt: Differential dynamic programming neural optimizer."
Li & Hao, 2018, "An optimal control approach to deep learning and applications to discrete-weight neural networks."
Li et al., 2017, "Maximum principle based algorithms for deep learning."