Reinforcement and Supervised Learning in Medical Physics & Engineering

Lekan Molu

(Pronounced Lay-con Moh-lu)

November 17, 2025

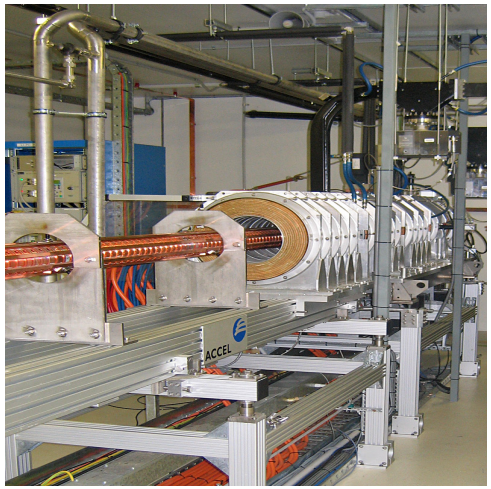
Talk outline: research ecosystem

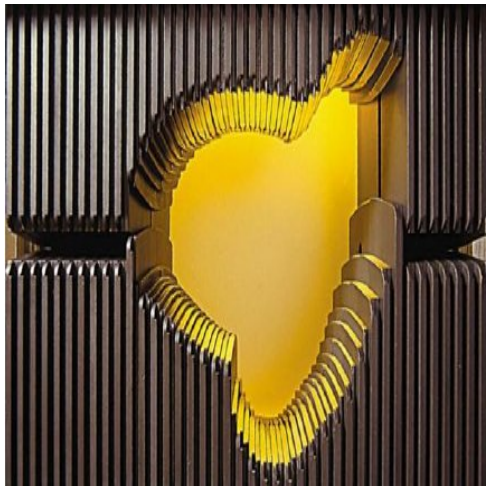
- Fictitious self-play with RL in beam angles selection and intensity modulated radiation therapy;
- Column-generation for beam angles selection in radiation therapy (RT);
- Representation in Reinforcement Learning (RL).

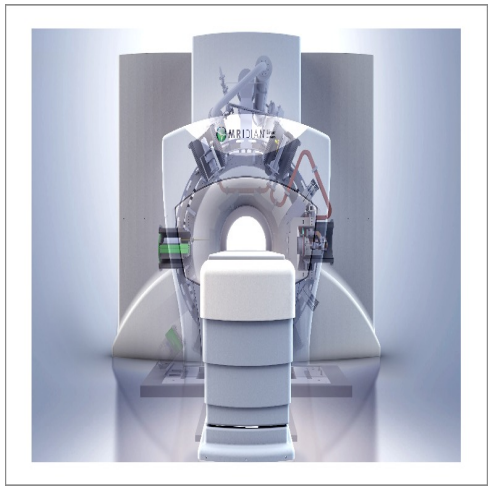
Technical overview

This page is left blank intentionally.

Beam orientation optimization







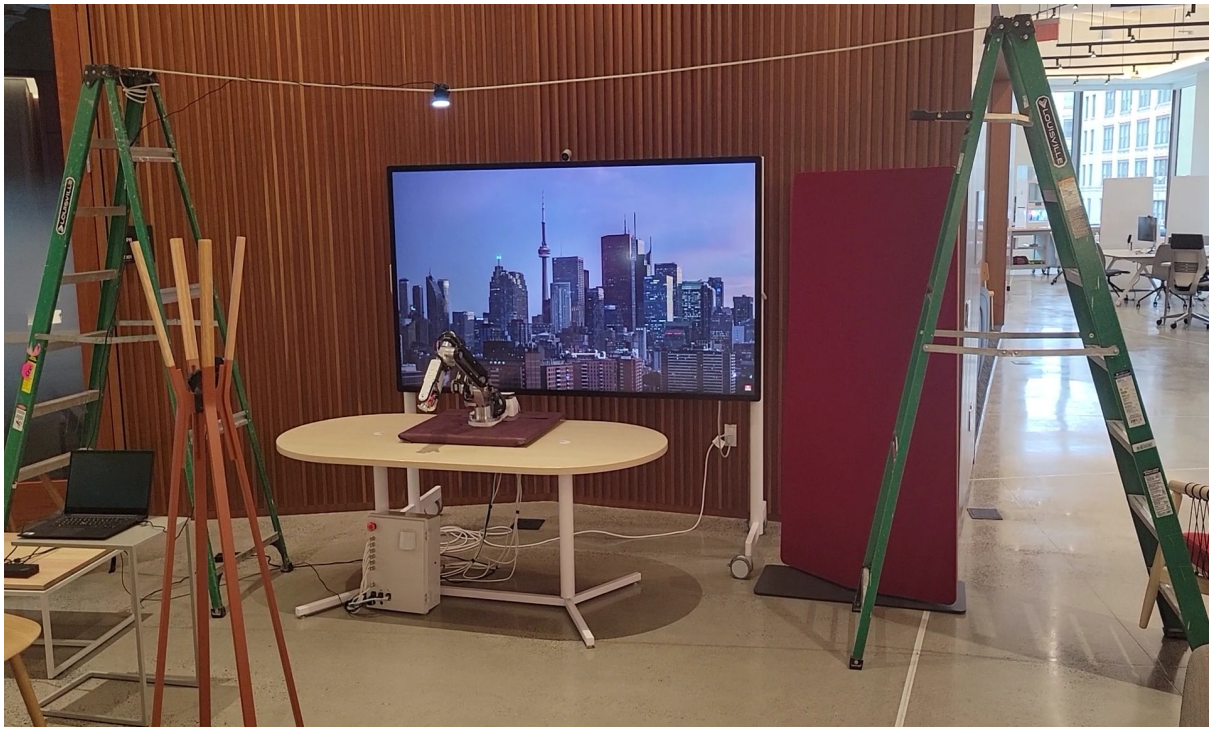
BOO relevant works

- Sadeghnejad Barkousaraie, Azar, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A fast deep learning approach for beam orientation optimization for prostate cancer treated with intensity-modulated radiation therapy." In *Medical physics: International Journal of Medical Physics Research and Practice,* 47, no. 3 (2020): 880-897.
- **Molu, Lekan**, Michael Folkerts, Dan Nguyen, Nicholas Gans, and Steve Jiang. "Deep BOO! Automating Beam Orientation Optimization in Radiation Therapy." In *Algorithm Foundations of Robotics XIII*, Merida, Mexico. Published in *Springer's Proceedings in Advanced Robotics (SPAR) Book*, 2020.
- Barkousaraie, Azar Sadeghnejad, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "Using Supervised Learning and Guided Monte Carlo Tree Search for Beam Orientation Optimization in Radiation Therapy." In *Workshop on Artificial Intelligence in Radiation Therapy*, pp. 1-9. Springer, Cham, 2019.
- Azar Sadeghnejad Barkousaraie, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A Fast Deep Learning Approach for Beam Orientation Selection Using Supervised Learning with Column Generation on IMRT Prostate Cancer Patients." *Medical Physics (AAPM)* 46 (6), E237-E237, San Antonio, TX, July 2019.
- **Lekan Molu**, Azar Sadeghnejad Barkousaraie, Nicholas Gans, Steve Jiang, and Dan Nguyen. "An Approximate Policy Iteration Scheme for Beam Orientation Selection in Radiation Therapy." *Medical Physics (AAPM)* 46 (6), E386-E386 San Antonio, TX, July 2019.
- Azar Sadeghnejad Barkousaraie, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A Reinforcement Learning Application of Guided Monte Carlo Tree Search Algorithm for Beam Orientation Selection in Radiation Therapy." *Medical Physics (AAPM)* 46 (6), E236-E236, San Antonio, TX, July 2019.

Transition slide

This page is left blank intentionally.

Controllable States Retrieval in RL



Representation learning selected works

- Anurag Koul, Shivakanth Sujit, Shaoru Chen, Ben Evans, Lili Wu, Byron Xu, Rajan Chari, Riashat Islam, Raihan Seraj, Yonathan Efroni, **Lekan Molu**, Miro Dudik, John Langford, Alex Lamb, 2023. PcLast: Discovering plannable continuous latent states. International Conference on Machine Learning (ICML).
- Alex Lamb, Riashat Islam, Yonathan Efroni, Aniket Didolkar, Dipendra Misra, Dylan Foster, **Lekan Molu**, Rajan Chari,kshay Krishnamurthy, and John Langford: Guaranteed Discovery of Controllable Latent States With Multi-step Inverse Models. Transactions on Machine Learning Research (2022)

Transition slide: External beam radiation therapy.

This page is left blank intentionally.

External beam radiation therapy. Funding Sources

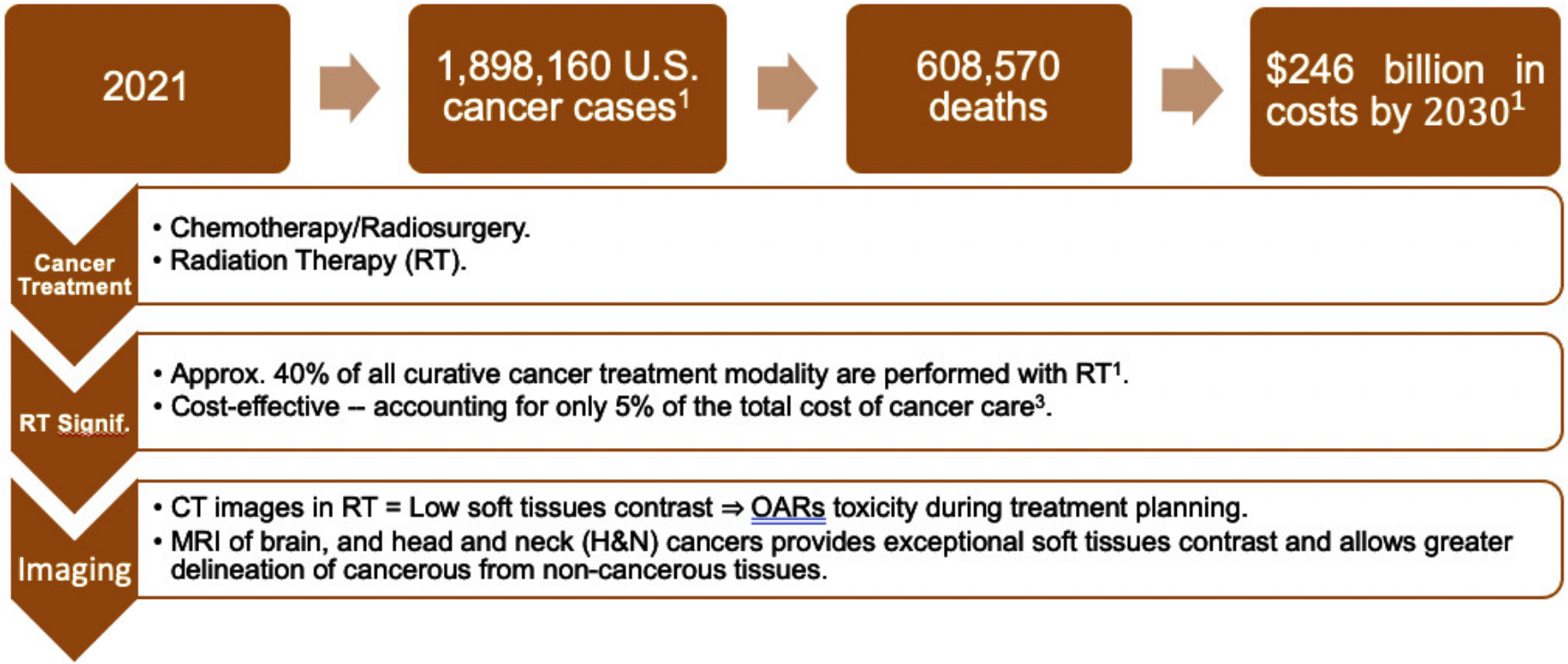
NATIONAL CANCER INSTITUTE

CANCER PREVENTION & RESEARCH INSTITUTE OF TEXAS

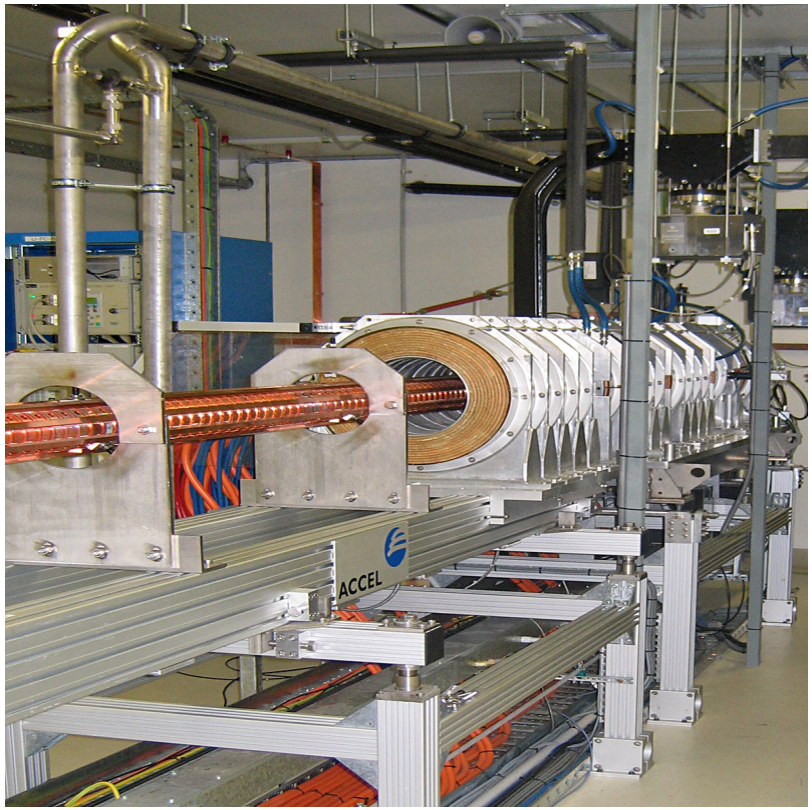
Talk outline: External beam radiation therapy

- Beam Orientation Optimization (BOO);
  - Column Generation as Pretraining for MCTS for BOO.
  - Monte Carlo Tree Search and Neuro-Dynamic Programming for BOO;

Research Significance

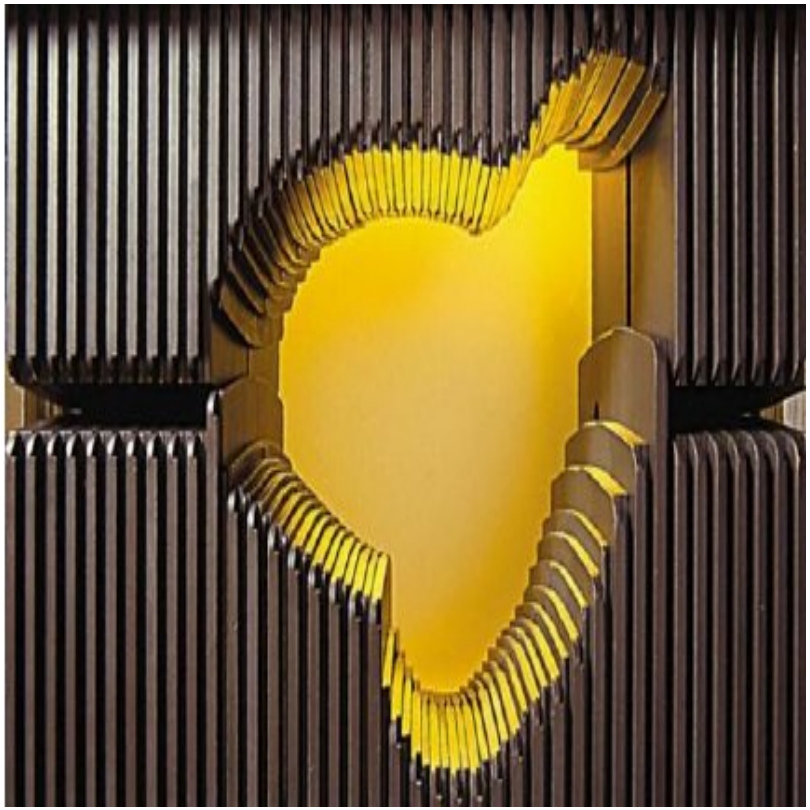


External beam radiation therapy (EBRT) Beam Delivery



©The Australian Synchrotron.

External beam radiation therapy (EBRT) Beam Delivery



A Multi-leaf collimator, ©Varian.

Radiation delivery couch and gantry

Varian's TrueBeam Radiotherapy System.

Transition slide: Column generation-guided supervised learning.

This page is left blank intentionally.

Part I.B: Supervised Column Generation Pretraining for BOO

- Beam Orientation Optimization (BOO)
  - Monte Carlo Tree Search and Neuro-Dynamic Programming for BOO
  - $\rightarrow$ Column Generation as Pretraining for Deep Neural Network BOO

Funding Agencies/Funds

- Cancer Prevention and Research Institute of Texas (CPRIT) (IIRA RP150485): $858,356. PI: Steve Jiang
- CPRIT MIRA RP160661: $4,103,894. PI: Steve Jiang
- NIH R-01 1R01CA237269-01: $490,133. PI: Steve Jiang

Relevant Publications

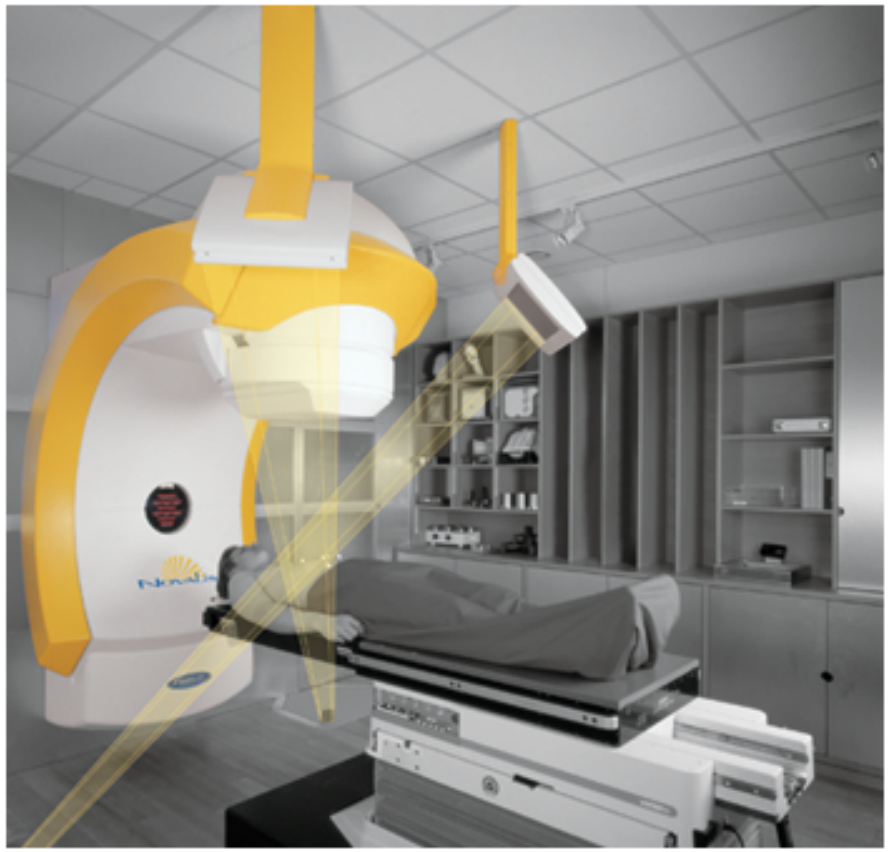
- Sadeghnejad Barkousaraie, Azar, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A fast deep learning approach for beam orientation optimization for prostate cancer treated with intensity-modulated radiation therapy." In *Medical physics: International Journal of Medical Physics Research and Practice,* 47, no. 3 (2020): 880-897.
- Barkousaraie, Azar Sadeghnejad, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "Using Supervised Learning and Guided Monte Carlo Tree Search for Beam Orientation Optimization in Radiation Therapy." In *Workshop on Artificial Intelligence in Radiation Therapy*, pp. 1-9. Springer, Cham, 2019.
- Azar Sadeghnejad Barkousaraie, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A Fast Deep Learning Approach for Beam Orientation Selection Using Supervised Learning with Column Generation on IMRT Prostate Cancer Patients." *Medical Physics (AAPM)* 46 (6), E237-E237, San Antonio, TX, July 2019.

Contributions

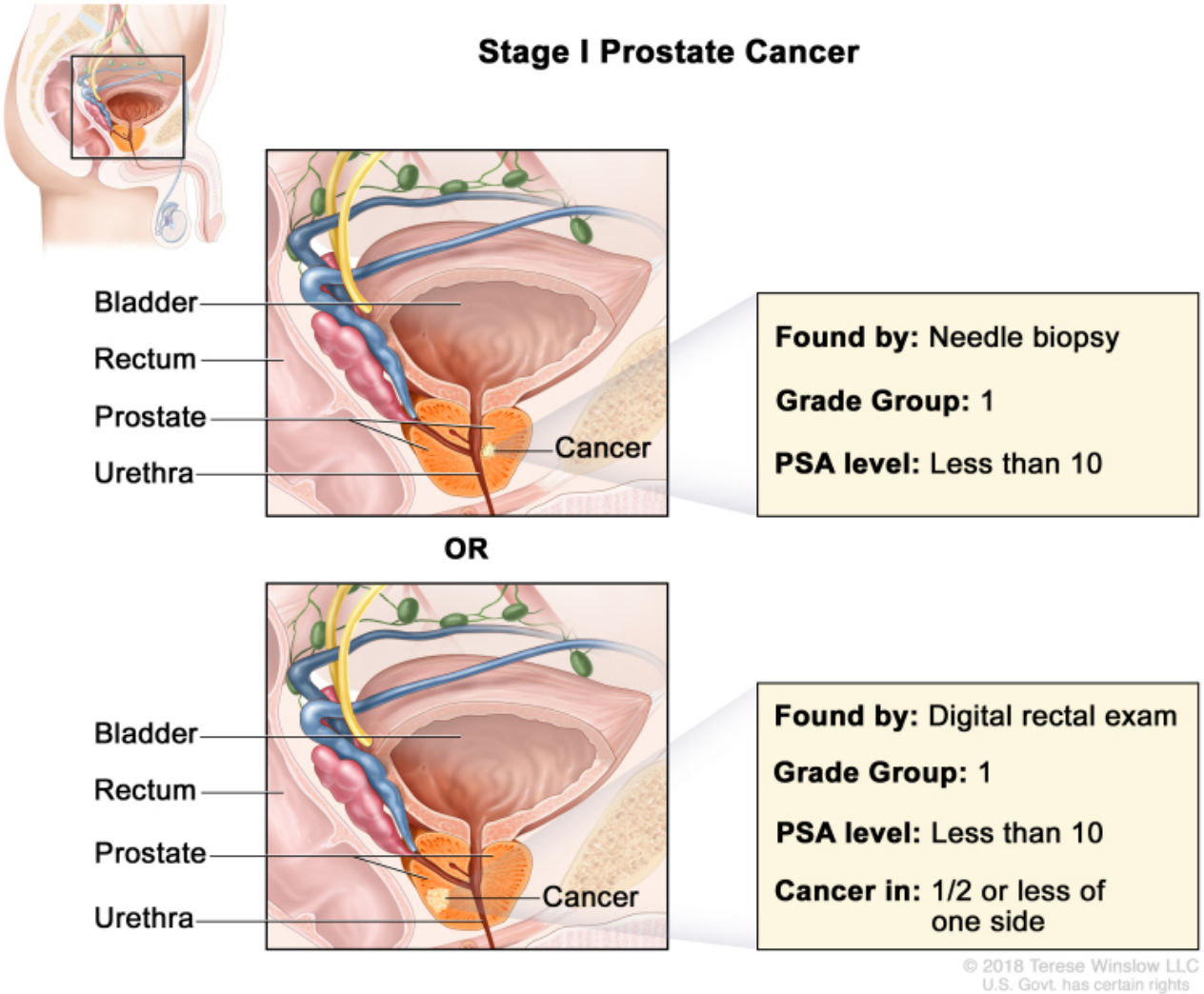
Relevant Publications

Azar Sadeghnejad Barkousaraie, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A fast deep learning approach for beam orientation optimization for prostate cancer treated with intensity-modulated radiation therapy." In *International Journal of Medical Physics Research and Practice*, https://doi.org/10.1002/mp.13986, 2019.

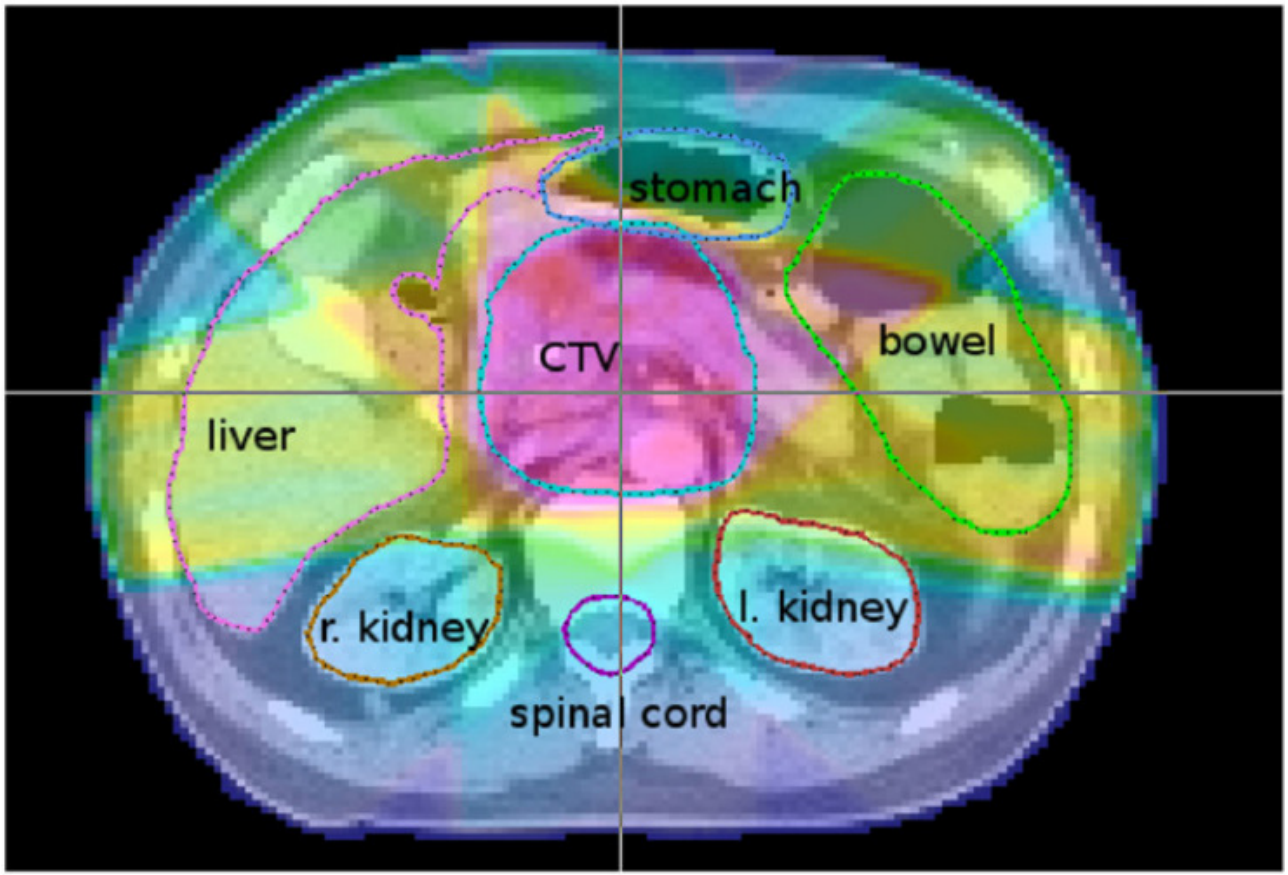
Beam angle optimization: A supervised learning approach



Protate cancer example



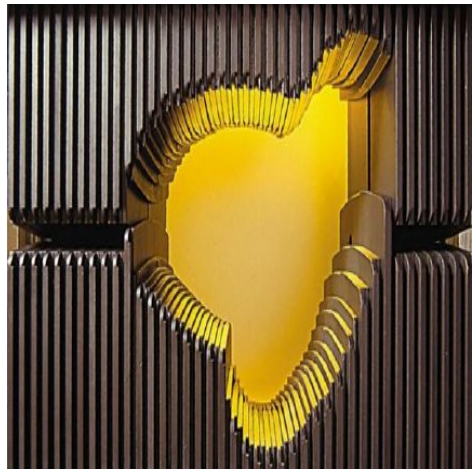
Clinical target volume example



Geometry of a 3D pancreatic case. Reprinte from Bertsimas et al. (2013).

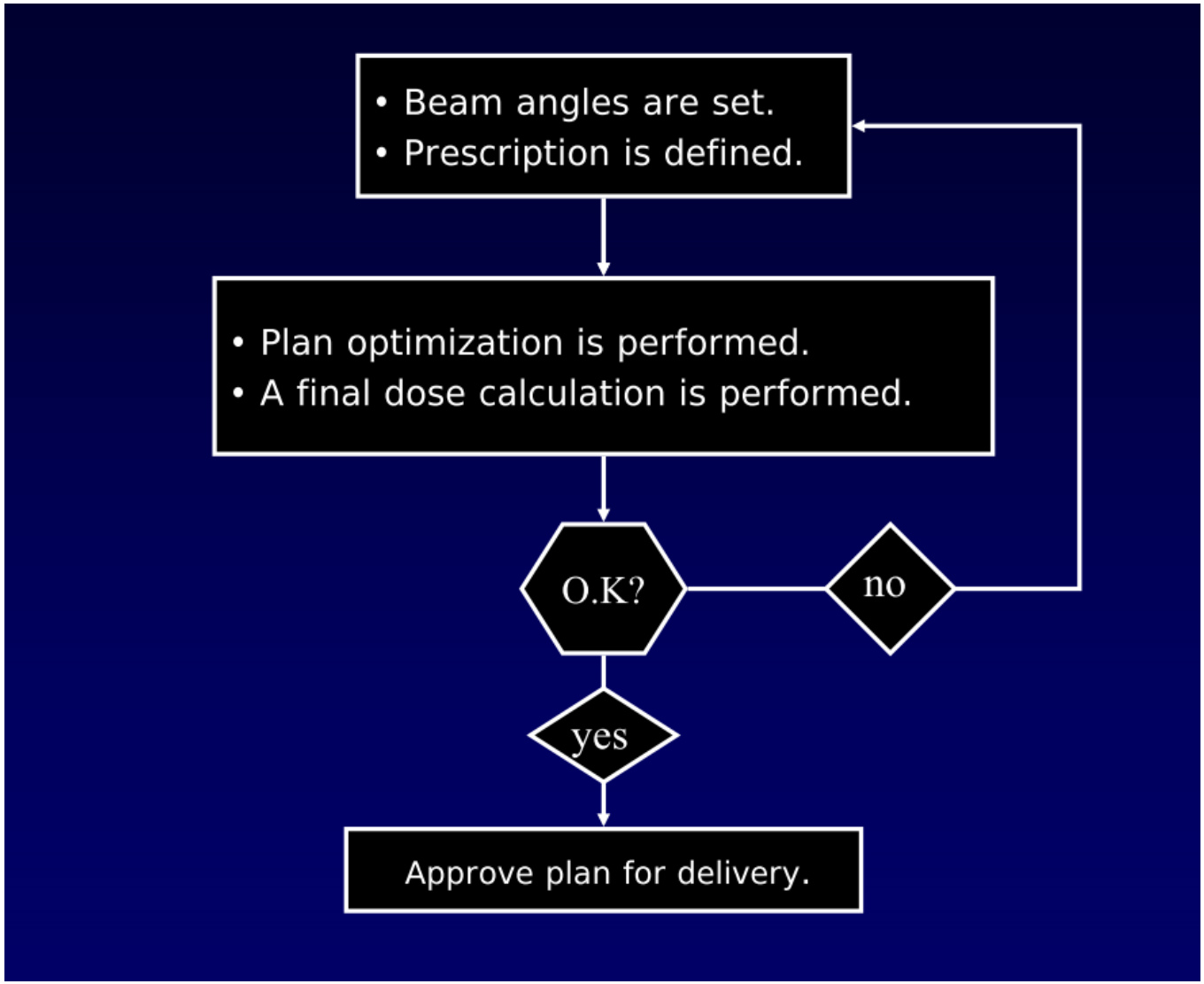
Building a treatment plan

- Selecting the number of beams;
- Selecting the directions from which to deliver the radiation;
- Optimizing the intensities of the beamlets in each beam (IMRT);
- Selection of delivery sequence (multileaf sequencing).



A Multi-leaf collimator, ©Varian.

Treatment plan flowchart



Reprinted from "IMRT Optimization Algorithms. David Shepard. Swedish Cancer Institute. AAPM 2007."

BOO workflow

Manually Selection/Protocols Adoption

Laborious process; could take up to 5 days for head and neck cancer treatment.

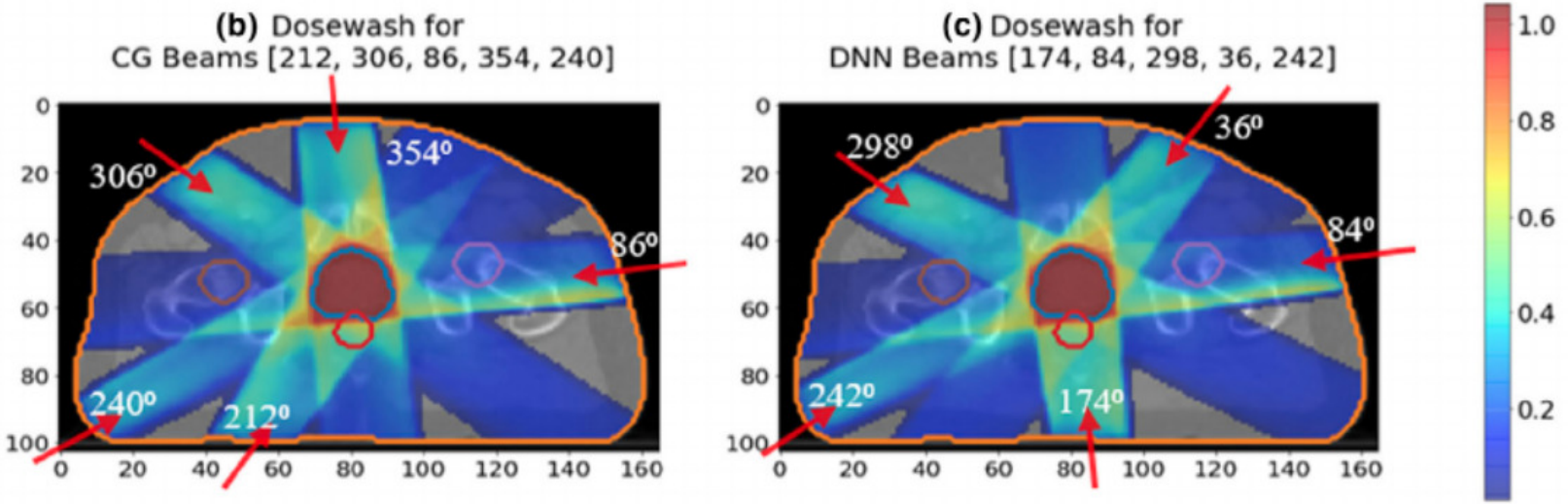
Pre-solve Large Sparse Dose Influence Matrix

Takes hours to solve for a single patient. Days/months for multiple patients.

Solve Fluence Map Optimization

Time-consuming: Often takes minutes.

Optimized example result and challenges



Delivered radiation dose after heavy optimization to a planning target volume example.

Dose influence matrix (Fluence)

- S volumes of interest, (VOIs);
  - Comprising organs-at-risk (OARs);
  - Normal tissues (all body voxels excluding specified target structure);
  - (Planning or clinical) target volume;
- Suppose V_h are the set of all voxels in a discretized grid $VOI_h\,(h = 1, \cdots, S)$;
- N_h: Number of voxels in VOI_h;
- B : set of beamlets, partitioned into subsets $B_1 \cup \cdots, B_n$ for n beams;

Dose influence matrix (Fluence)

- Set x_j, $j \in B$ as the beam intensity;
- θ_k, $k = 1, \cdots, n$ continuous beam angles;
- d_i dose delivered to voxel i, $(i \in V_h,\ h = 1, \cdots, S)$;
- Δ discretization resolution for all continuous angles $1, \cdots, n$.

- D matrix of dose influence;
  - Characterized by $D_{ij}(\theta_k)$;
  - For dose $i \in V_h$;
  - From beam $j \in B_k$ delivered at angle θ_k;
- D is computed at a resolution of $360^\circ/\Delta$ degrees.

Dosing objectives and constraints

- Three objectives: $O_p \& C_p$, $p = \{1, 2, 3\}$;
  - Type 1: (O_1 and C_1) for OARs: Penalize weighted convex combination (given by parameter $0 \leq \alpha_h \leq 1$) of the max. dose y_h to OAR $h \in O_1$ and of the mean dose $1/N_h(\sum_{i \in V_h} d_i)$ to the OAR;
  - Type 2: (O_2 and C_2) for target volumes: Penalize (negated) weighted convex combination (given by parameter $0 \leq \alpha_h \leq 1$) of the min. dose y_h to target and the mean dose $1/N_h(\sum_{i \in V_h} d_i)$ to the tumor;

Dosing objectives and constraints

- Three objectives: $O_p \& C_p$, $p = \{1, 2, 3\}$;
  - C_1 for upper bound on the max, and mean dose to OAR.
  - C_2; for lower bound on convex combination of the min and mean dose to target, h
- Type 3: (O_3 and C_3): A ramp function for to limiting underdosing to a target and over- dosing to an OAR.

Mixed-integer nonconvex optimization

$$\text{minimize} \sum_{h \in O_1 \cup O_2} w_h \left(\alpha_h y_h + (1-\alpha_h) 1/N_h \left(\sum_{i \in V_h} d_i \right) \right) + \sum_{h \in O_3} w_h \left(1/N_h \sum_{i \in V_h} z_i^h \right) \tag{1}$$

$$\sum_{k=1}^{n} \sum_{j \in B_k} D_{ij}(\theta_k) x_j = d_i, \quad i \in V_h, \ h = 1, \ldots, S \tag{2}$$

$$d_i \geq LB_i, \quad i \in V_h, \ h = 1, \ldots, S \tag{3}$$

$$d_i \leq UB_i, \quad i \in V_h, \ h = 1, \ldots, S \tag{4}$$

$$x_j \geq 0, \quad j \in B_k, \ k = 1, \ldots, n \tag{5}$$

$$0 \leq \theta_k \leq 360, \quad k = 1, \ldots, n \tag{6}$$

$$y_h \geq d_i, \quad i \in V_h, \ h \in O_1 \cup C_1 \tag{7}$$

$$y_h \leq d_i, \quad i \in V_h, \ h \in O_2 \cup C_2 \tag{8}$$

$$\alpha_h y_h + (1-\alpha_h) 1/N_h \sum_{i \in V_h} d_i \leq g_h, \quad h \in C_1 \tag{9}$$

$$\alpha_h y_h + (1-\alpha_h) 1/N_h \sum_{i \in V_h} d_i \geq g_h, \quad h \in C_2 \tag{10}$$

$$z_i^h \geq s_h^u (d_i - t_h), \quad h \in O_3 \cup C_3, \ i \in V_h \tag{11}$$

$$z_i^h \geq s_h^l (t_h - d_i), \quad h \in O_3 \cup C_3, \ i \in V_h \tag{12}$$

Limited BOO Optimization

$$\underset{x,y,B_{limit}}{minimize} \quad F(y)$$

$$subject\,to \quad y = \sum_{b \in B_{limit}} \begin{bmatrix} D_{b,s=s_1} \\ \vdots \\ D_{b,s=s_T} \end{bmatrix} x_b$$

$$x_b \geq 0 \quad for\, b \in B_{all}$$

$$|B_{limit}| \leq n \quad B_{limit} \in B_{all}$$

- where B_{limit} is a set of limited ($n \approx 5 - 10$) feasible for treatment delivery to the patient.
- Kicker: solving B_{limit} is computationally prohibitive;
- Kicker: exhaustive beam angle search infeasible either for fast treatment planning.

Iterative greedy-based beam selection

- Column generation approximates limited BOO problem
  - Iteratively add a beam with the greatest likelihood to improve the current FMO solution;
  - FMO leverages Chambolle-Pock first-order primal-dual proximal operator on GPU;
  - DNN then trained to learn beam orientation reasoning of CG
- DNN essentially internalizes the FMO solution via CG.s

Minimizing computational expense of CG

Algorithm 1. *Brief Column Generation Structure for Beam Orientation Selection*

1. Initialize B_{limit} as an empty set: $k = 0, B^0_{limit} = \emptyset$
2. While $|B_{limit}| < n$:
   a. $b^{k+1} = \underset{\tilde{b} \in B_{all} \setminus B^k_{limit}}{argmin} \{P(\hat{B}): \hat{B} = B^k_{limit} + \tilde{b},\ |\hat{B}| = |B^k_{limit}| + 1\}$
   b. $B^{k+1}_{limit} = B^k_{limit} + b^{k+1}$

Column generation scheme

- Column generation approximates limited BOO problem
  - expensive since FMO for each beam in $B_{all} \setminus B^k_{limit}$ at each iteration k;
  - KKT conditions for master problem fast transitions!
  - If KKT conditions not met, check which variables are furthest from optimality, and would, therefore, improve the objective value the quickest if corrected!
- KKT conditions reveal which single beam would best improve the objective value at the next iteration

KKT trick

TABLE II. Karush-Kuhn-Tucke (KKT) CONDITIONS for problem P(B_all).

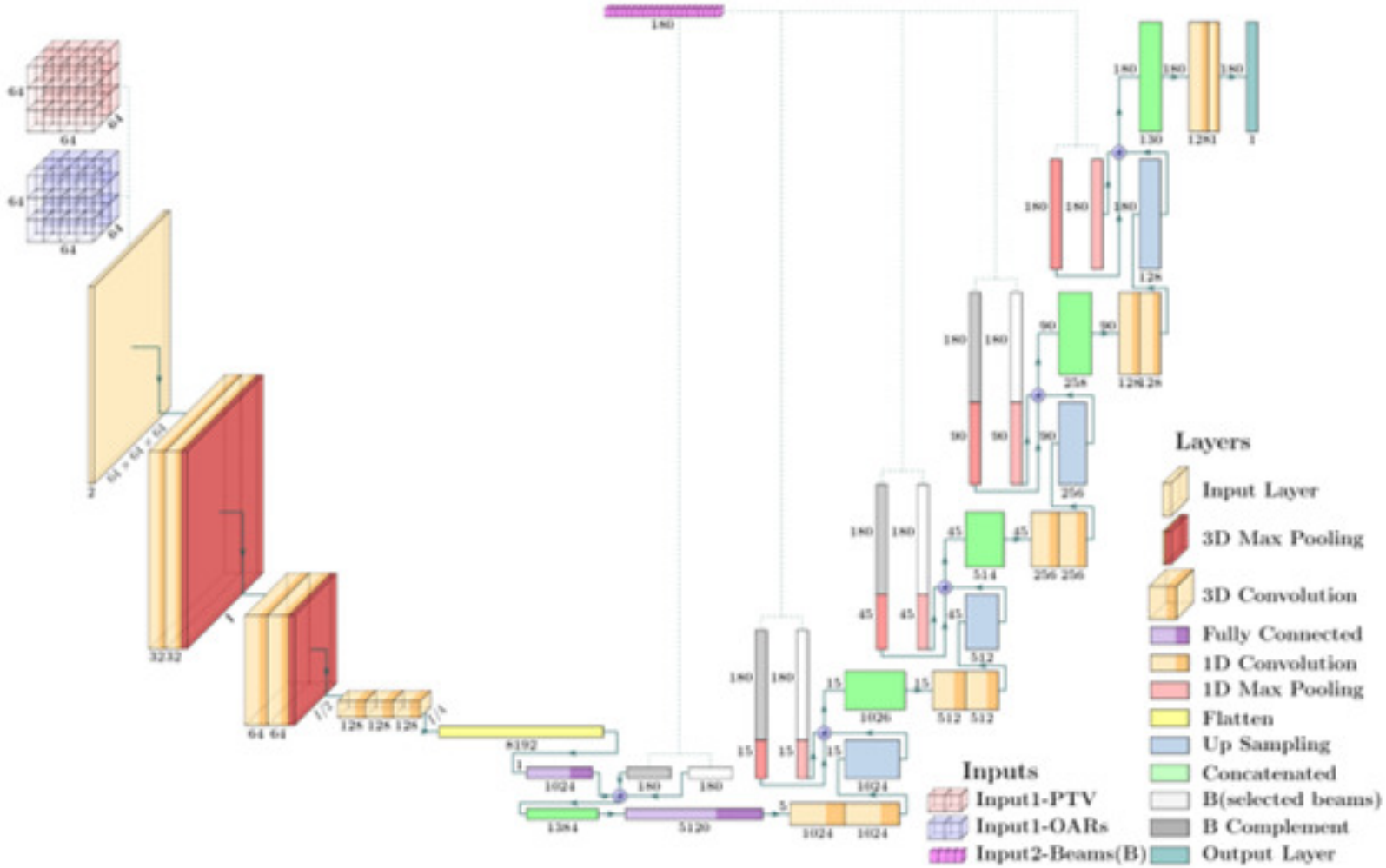
KKT conditions:	
Stationarity	$v_b = \begin{bmatrix} D_{b,s=s_1} \\ \vdots \\ D_{b,s=s_T} \end{bmatrix}^T z \quad for\, b \in B_{all}$ (1)
	$z \in \partial F(y)$ (2)
Primal feasibility	$y = \sum_{b \in B} \begin{bmatrix} D_{b,s=s_1} \\ \vdots \\ D_{b,s=s_T} \end{bmatrix} x_b$ (3)
	$x_b \geq 0 \quad for\, b \in B_{all}$ (4)
Dual feasibility	$v_b \geq 0 \quad for\, b \in B_{all}$ (5)
Complementary slackness	$v_{b,i} x_{b,i} = 0 \quad \forall b, i$ (6)

CG-DNN Algorithm

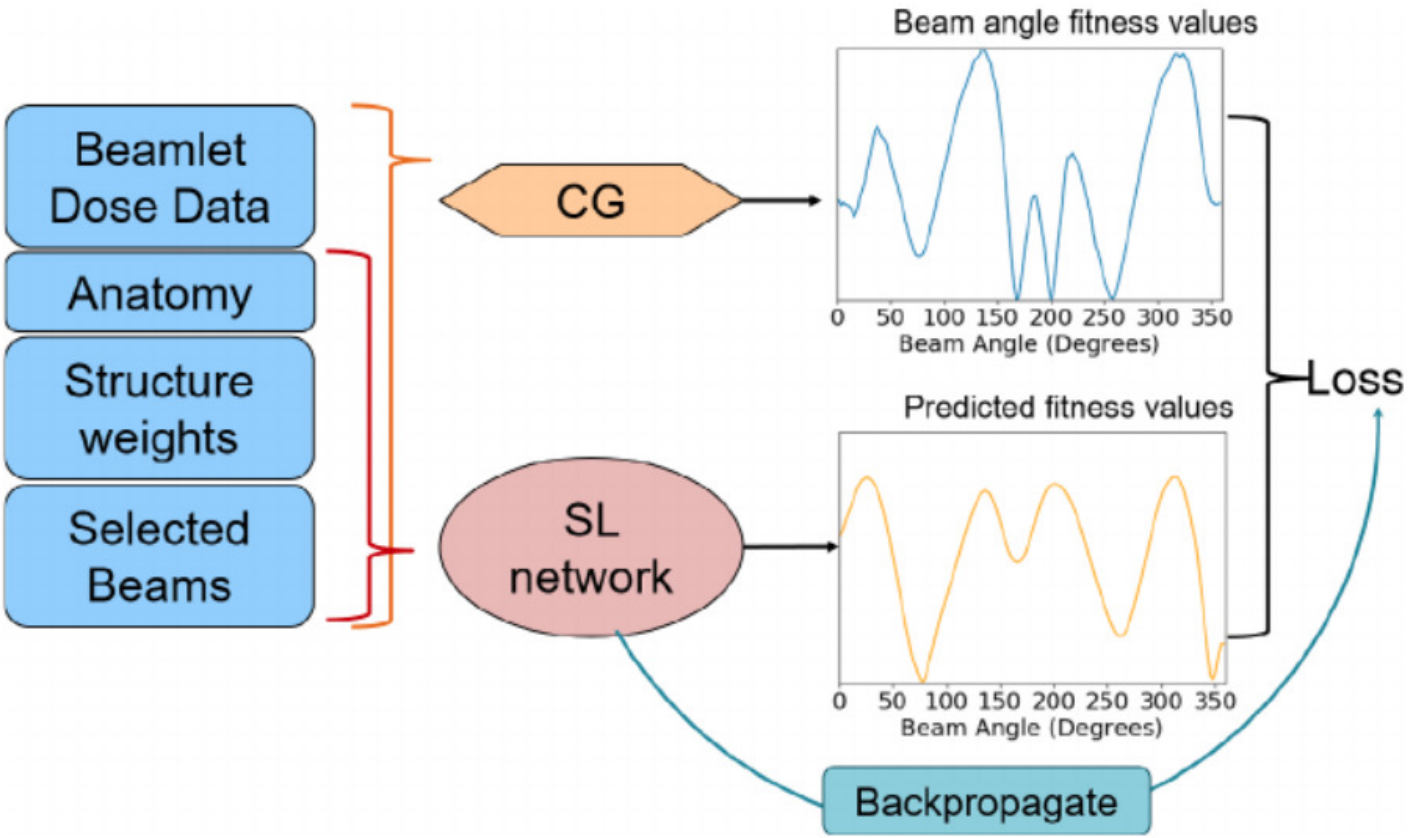
Algorithm 2. *Solving a Sequence of a Limited BOO Problem to Select n Beam Orientations*

1. Create a one-dimensional array (A) with the same size as J (in this work $|J| = 180$)
2. Set current number of selected beam orientation (B) as 0, $n_B = 0$
3. Initialize array A with zeros
4. Set the value of current objective function $F_{now} = \infty$
5. While $n_B < n$ and stopping criteria has not been met, do:
   a. Use the updated set of B to define $P_{limit}(B, n_B)$
   b. $P_{limit}(B, n_B)$ by Chambolle-Pock algorithm[56]
   c. For each structure $s_t \in S$, calculate the Lagrange multiplier as z_{s_t}, whose size is the number of voxels in s_t
   d. Define vector v_b for all beams ($b \in B_{all}$)
   e. For each beam $b_j \in B_{all} \backslash B$:
      i. For each structure $s_t \in S$:
         1. Calculate $D_{b_j s_t}$, the dose matrix of beam b_j for s_t
         2. $v_{b_j} \mathrel{+}= D_{b_j s_t} z_{s_t}$
      ii. Set $r_{b_j} = -v_{b_j}$
   f. If $r_b \leq 0 \quad \forall\, b \in B_{all}$
      i. Stop the algorithm. The solution is optimal.
   g. Otherwise:
      i. Normalize r values by Calculate fitness vector f for all beam with equation (9)
      ii. $\tilde{b} = argmax(f)$,
      iii. Update the associated element in array A: $(\alpha_{\tilde{b}})$ to 1, $A_{\alpha_{\tilde{b}}} = 1$
      iv. $n_B = n_B + 1$
6. Return A

Network Structure



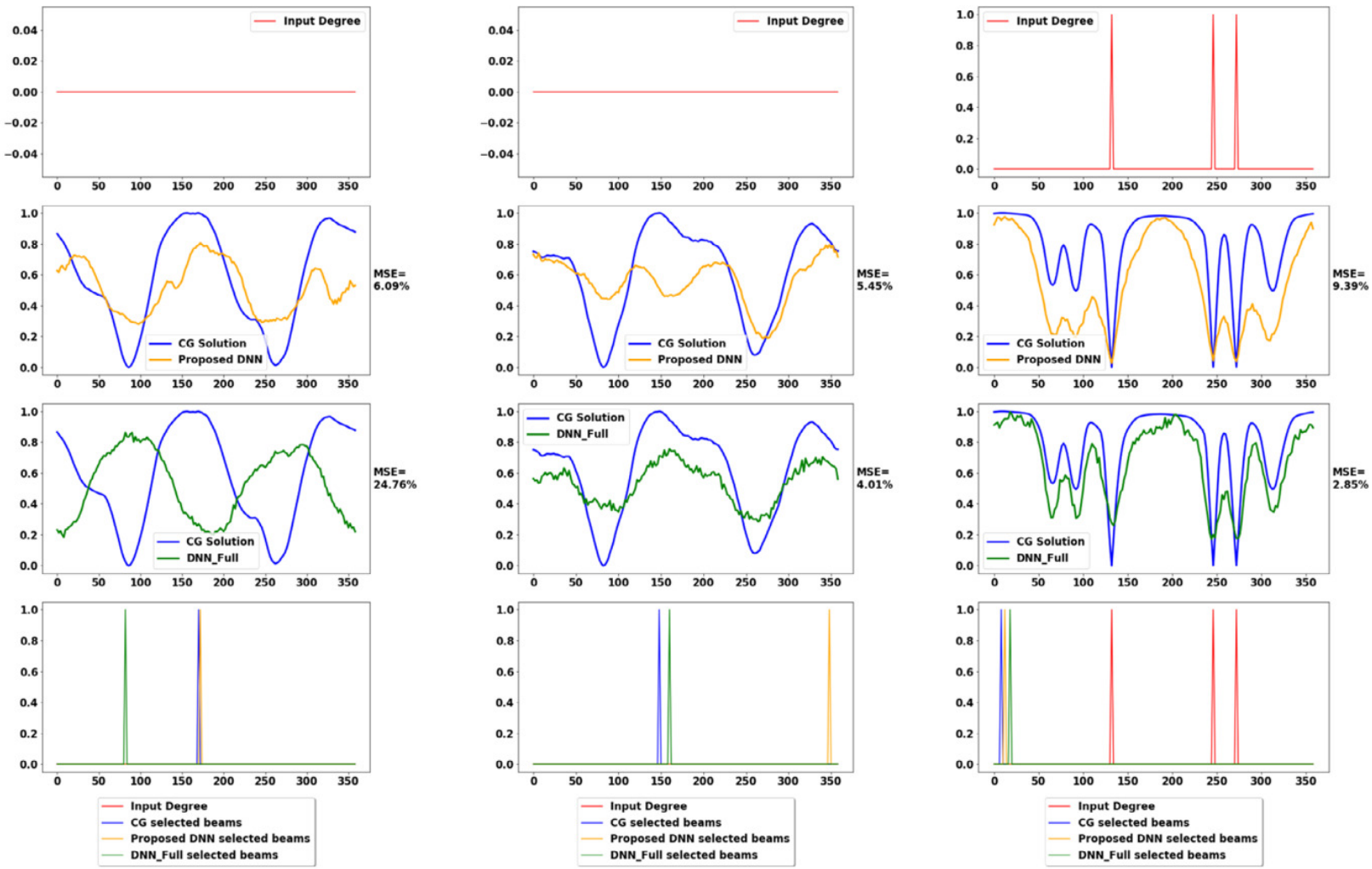
Training Schematic



Training and Validation Loss

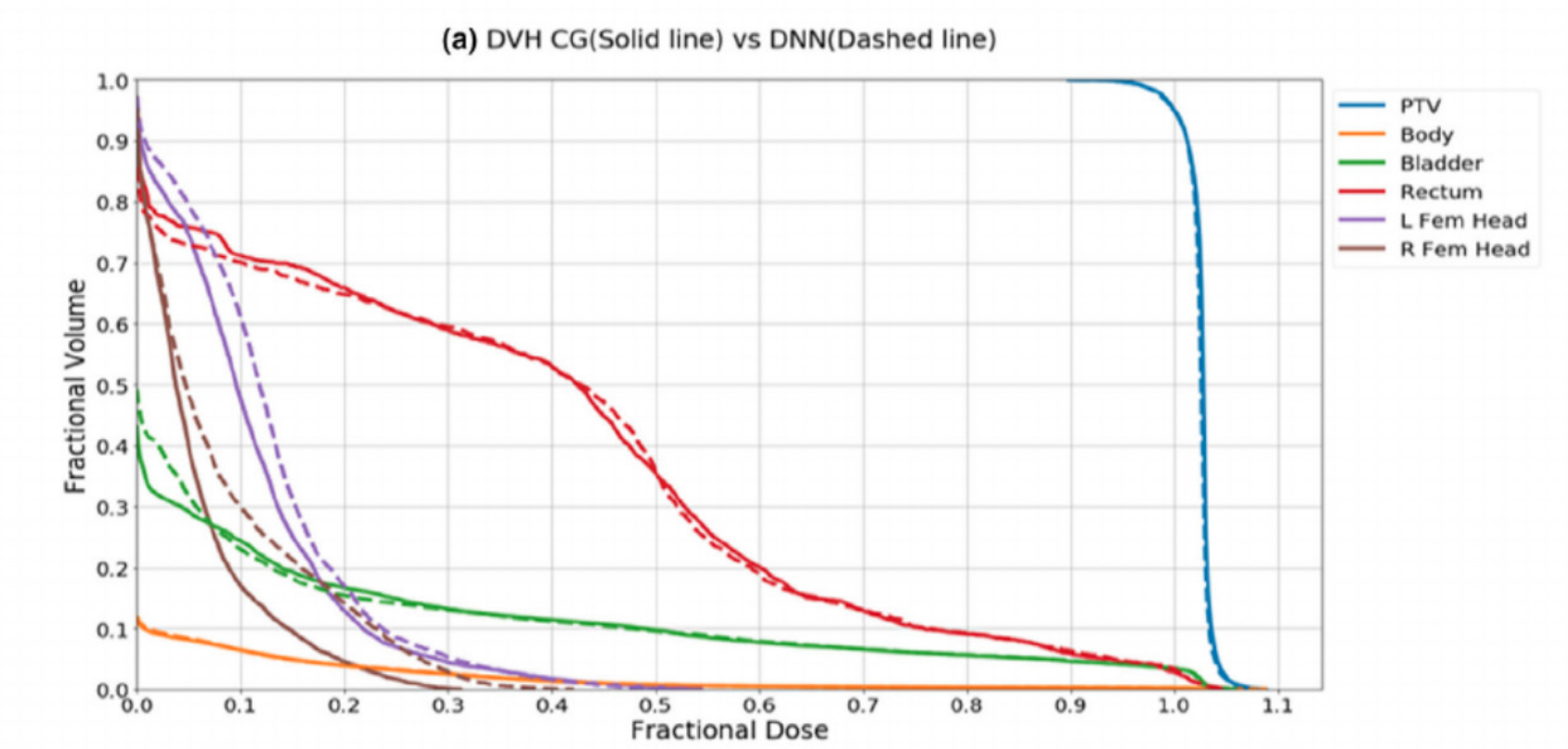
Average training (solid) and validation (dotted) loss function (MSE) values across six cross-validation folds for the network (blue) and full network.

Inference: Column Generation vs U-Net



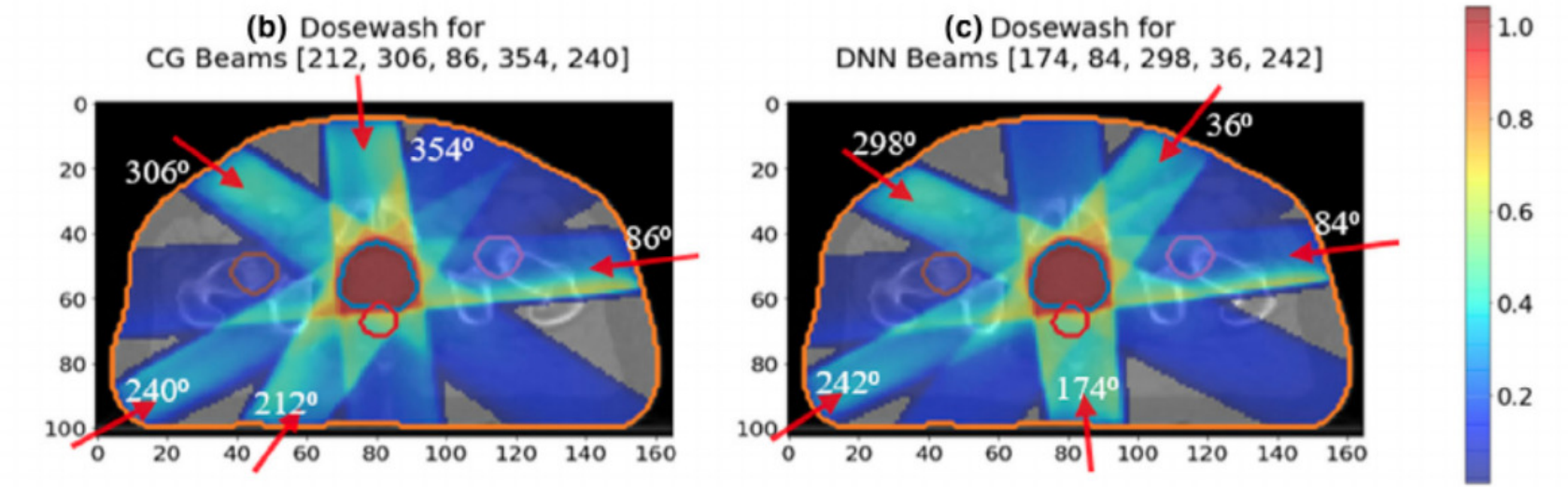
(a) Prediction of 1st beam orientation given no beam. (b) Prediction of 2nd beam orientation given 1 beam. (c) Prediction of 3rd beam orientation given 2 beams. (d) Prediction of 4th beam orientation given 3 beams. (e) Prediction of 5th beam orientation given 4 beams.

DVH of Column Generation vs Neural Network



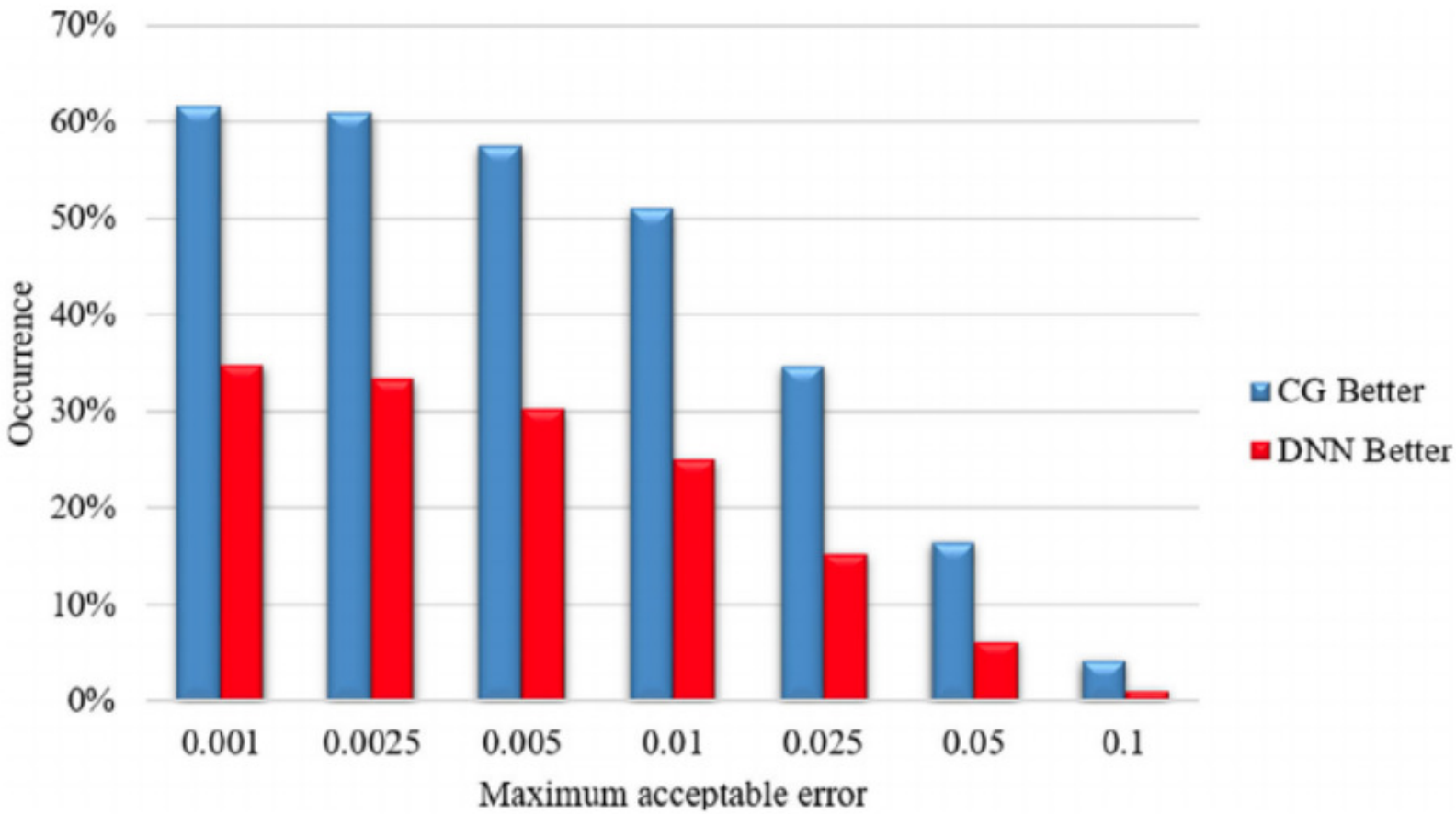
Dose-Volume Histogram of CG vs DNN architectures

Dose Washes of Column Generation vs Neural Network



Dose-Volume Histogram of CG vs DNN architectures

FMO Costs: Column Generation vs Neural Network



Dose-Volume Histogram of CG vs DNN architectures

Conclusions

- A Sparse Lookout Tree Strategy for guiding beam angle transitions while training a neural network
- A supervised deep neural network that learns from a CG algorithm
- Both methods leverage the convex FMO in learning the *optimal* set of beam angles
  - Alternating Direction Method of Multipliers
  - Chambolle-Pock algorithm
- Trade-offs in solutions generated by either approach allows flexibility for treatment planners
- Grants

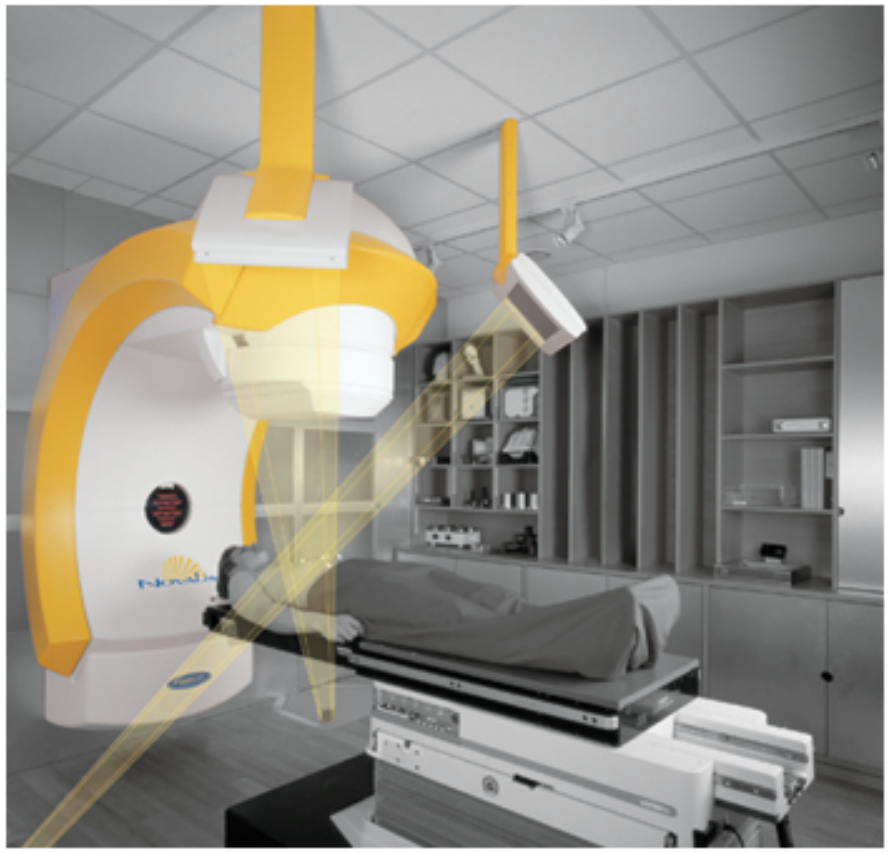
Beam Orientation Optimization (BOO)

- Beam Orientation Optimization (BOO)
  - $\rightarrow$ Monte Carlo Tree Search and Neuro-Dynamic Programming

Beam Orientation Optimization (BOO)

- Beam Orientation Optimization (BOO)
  - $\rightarrow$ Monte Carlo Tree Search and Neuro-Dynamic Programming
  - Column Generation as Pretraining for Deep Neural Network BOO

Beam angle optimization: An RL approach



BOO relevant works

- Sadeghnejad Barkousaraie, Azar, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A fast deep learning approach for beam orientation optimization for prostate cancer treated with intensity-modulated radiation therapy." In *Medical physics: International Journal of Medical Physics Research and Practice,* 47, no. 3 (2020): 880-897.
- **Molu, Lekan**, Michael Folkerts, Dan Nguyen, Nicholas Gans, and Steve Jiang. "Deep BOO! Automating Beam Orientation Optimization in Radiation Therapy." In *Algorithm Foundations of Robotics XIII*, Merida, Mexico. Published in *Springer's Proceedings in Advanced Robotics (SPAR) Book*, 2020.
- Barkousaraie, Azar Sadeghnejad, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "Using Supervised Learning and Guided Monte Carlo Tree Search for Beam Orientation Optimization in Radiation Therapy." In *Workshop on Artificial Intelligence in Radiation Therapy*, pp. 1-9. Springer, Cham, 2019.
- Azar Sadeghnejad Barkousaraie, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A Fast Deep Learning Approach for Beam Orientation Selection Using Supervised Learning with Column Generation on IMRT Prostate Cancer Patients." *Medical Physics (AAPM)* 46 (6), E237-E237, San Antonio, TX, July 2019.
- **Lekan Molu**, Azar Sadeghnejad Barkousaraie, Nicholas Gans, Steve Jiang, and Dan Nguyen. "An Approximate Policy Iteration Scheme for Beam Orientation Selection in Radiation Therapy." *Medical Physics (AAPM)* 46 (6), E386-E386 San Antonio, TX, July 2019.
- Azar Sadeghnejad Barkousaraie, **Lekan Molu**, Steve Jiang, and Dan Nguyen. "A Reinforcement Learning Application of Guided Monte Carlo Tree Search Algorithm for Beam Orientation Selection in Radiation Therapy." *Medical Physics (AAPM)* 46 (6), E236-E236, San Antonio, TX, July 2019.

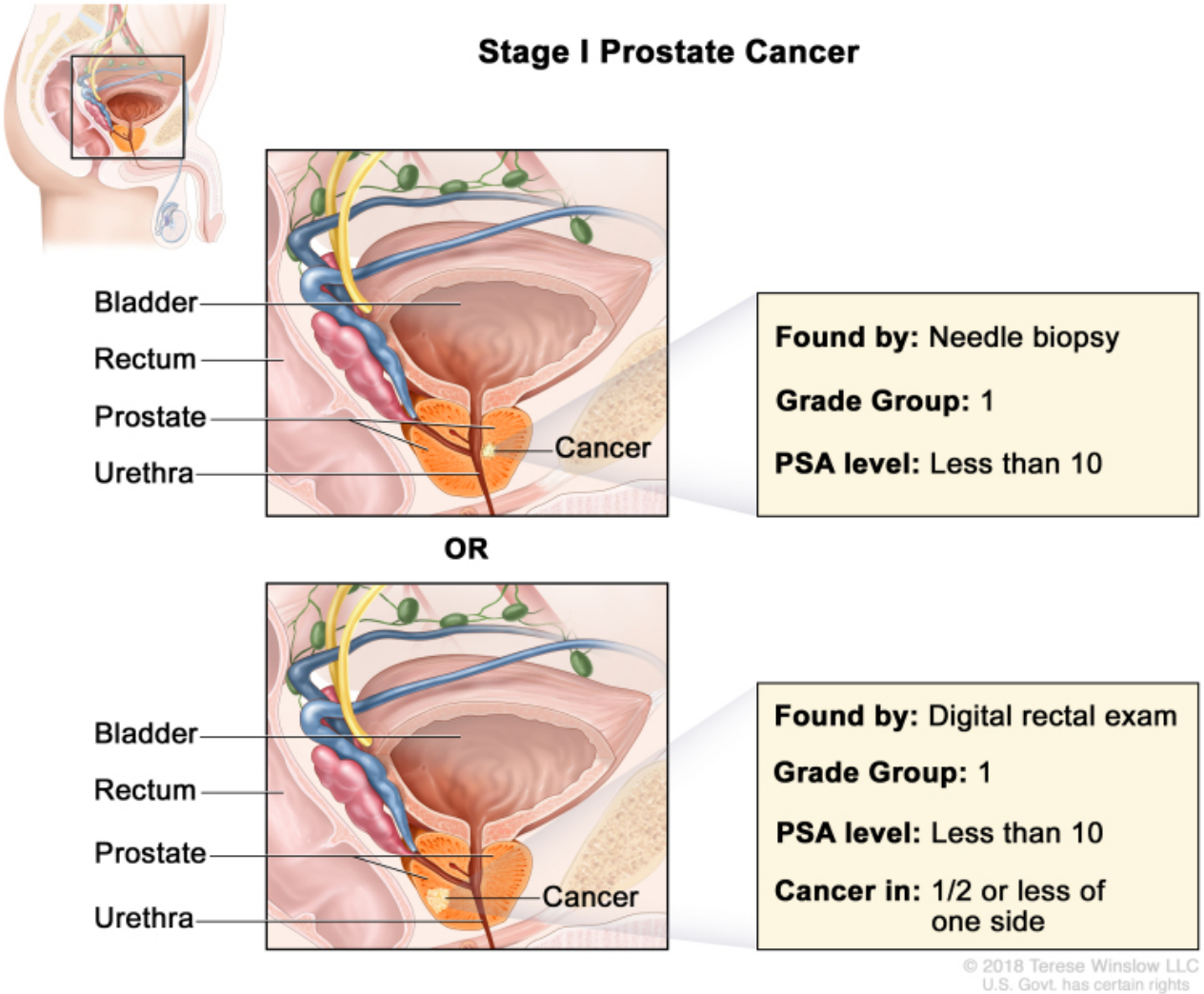
Contributions

Relevant Publications

Molu, Lekan, Michael Folkerts, Dan Nguyen, Nicholas Gans, and Steve Jiang. "Deep BOO! Automating Beam Orientation Optimization in Radiation Therapy." In *Algorithm Foundations of Robotics XIII*, Merida, Mexico. Published in *Springer's Proceedings in Advanced Robotics (SPAR) Book*, 2020.

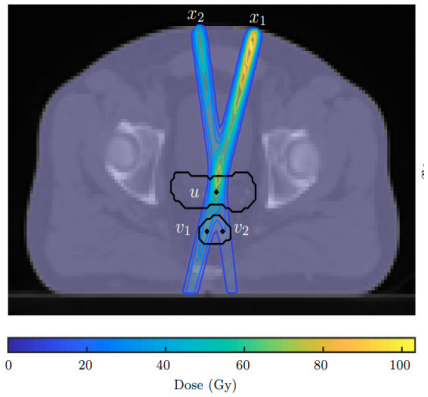
- A sparse tree lookout strategy for games with large state spaces guides transition between beam angle sets
- Tree lookout strategy guided by a deep neural network policy

Protate cancer example

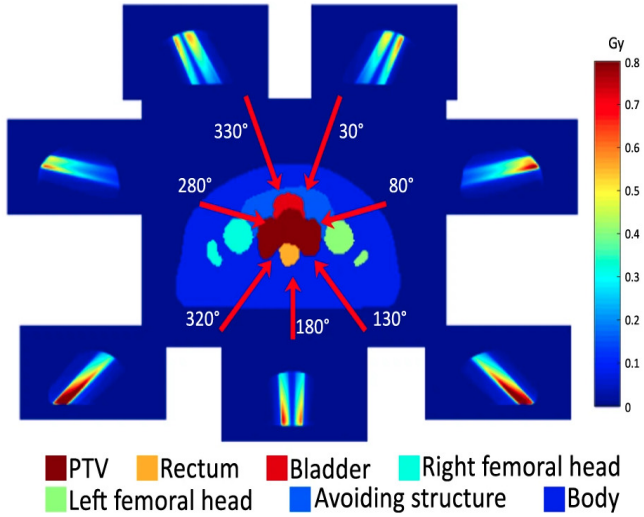


BOO Process: Fluence map optimization

Prostate CT slice



Prostate before BOO



Fluence Map

BOO workflow

Manually Selection/Protocols Adoption

Laborious process; could take up to 5 days for head and neck cancer treatment.

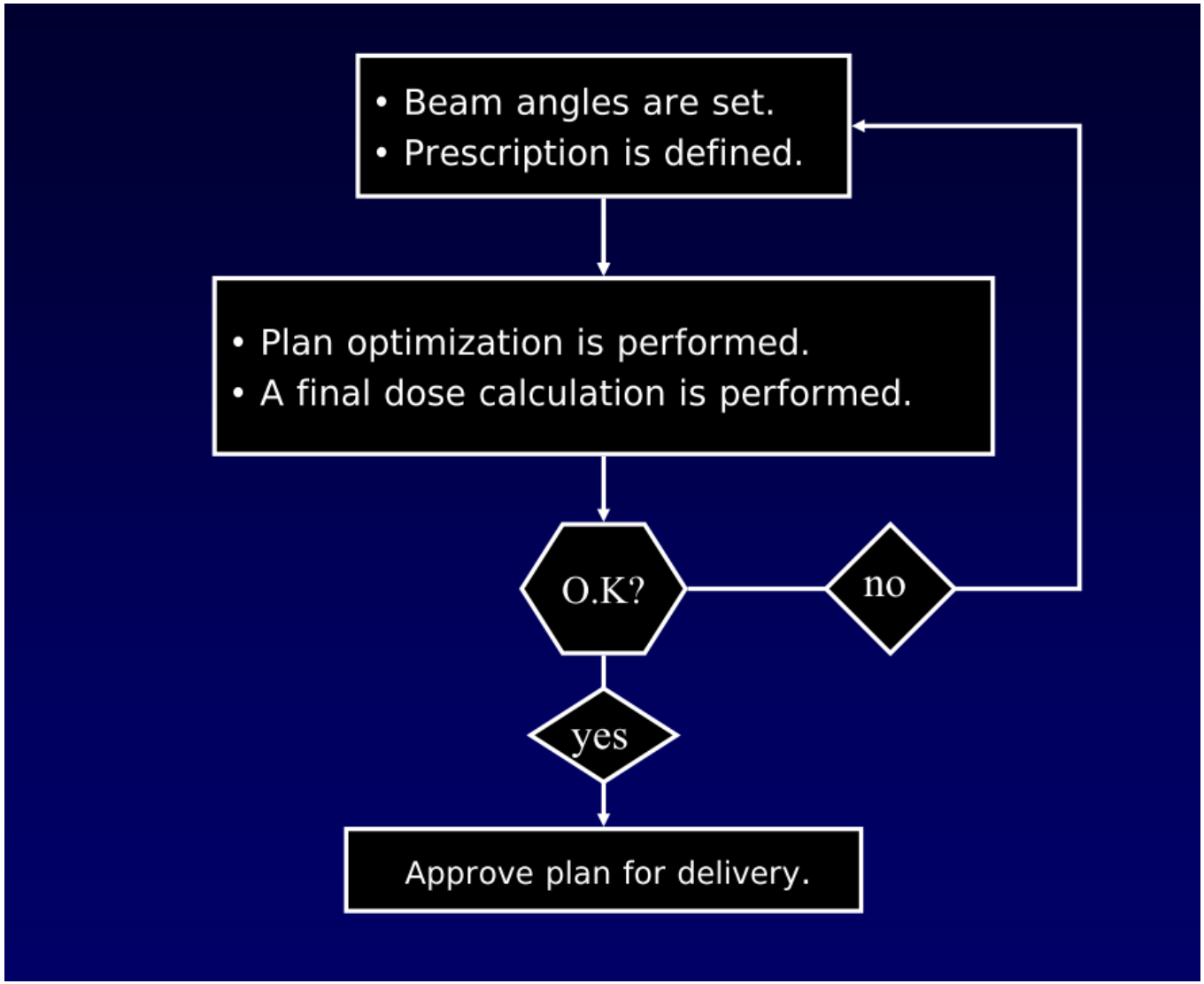
Pre-solve Large Sparse Dose Influence Matrix

Takes hours to solve for a single patient. Days/months for multiple patients.

Solve Fluence Map Optimization

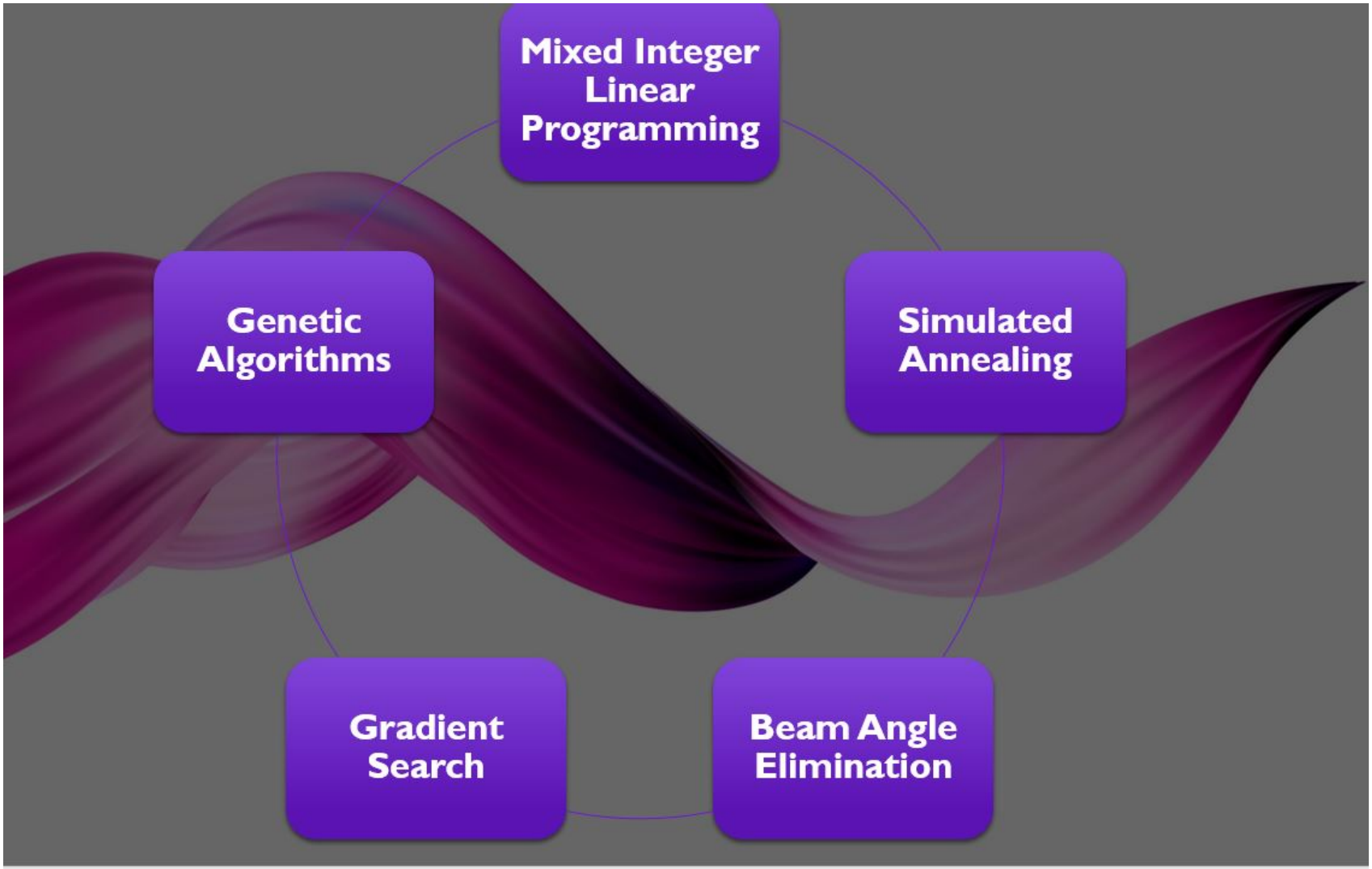
Time-consuming: Often takes minutes.

Treatment plan flowchart



Reprinted from "IMRT Optimization Algorithms. David Shepard. Swedish Cancer Institute. AAPM 2007."

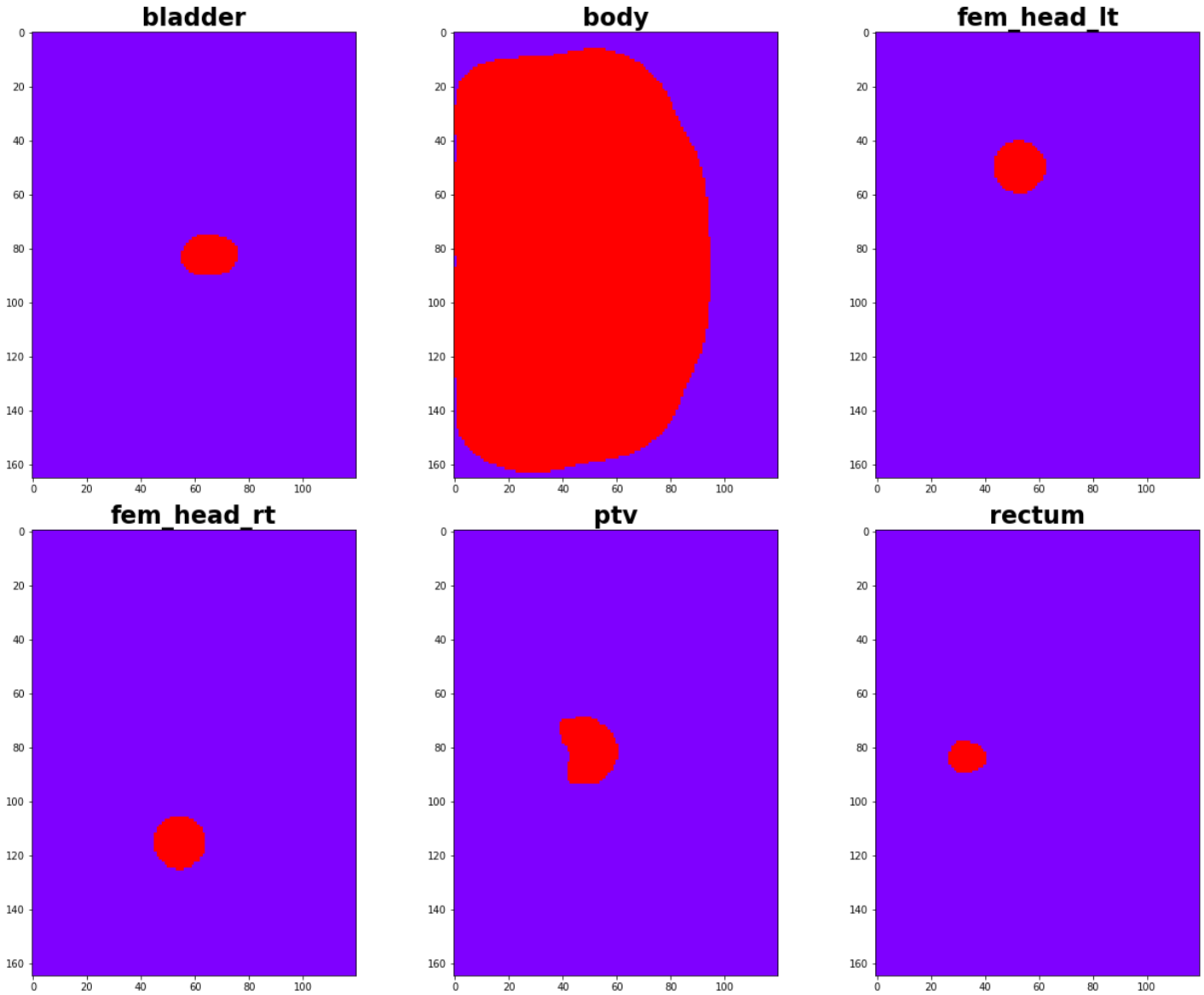
Current approaches and limitations



Innovation

- A tower neural network generates a policy that guides MCTS simulations for two players in a zero-sum Markov game
  - Produces a *utility (value) function* & a subjective *probability distribution*
- Each player in a two-player Markov game finds an alternating best response to the current player's average strategy
  - driving the neural network policy's weights toward an approximate **saddle equilibrium** [Heinrich et al. (2015)].
  - aids network in finding an *approximately optimal* beam angle candidate set that meets a dosimetric requirements.

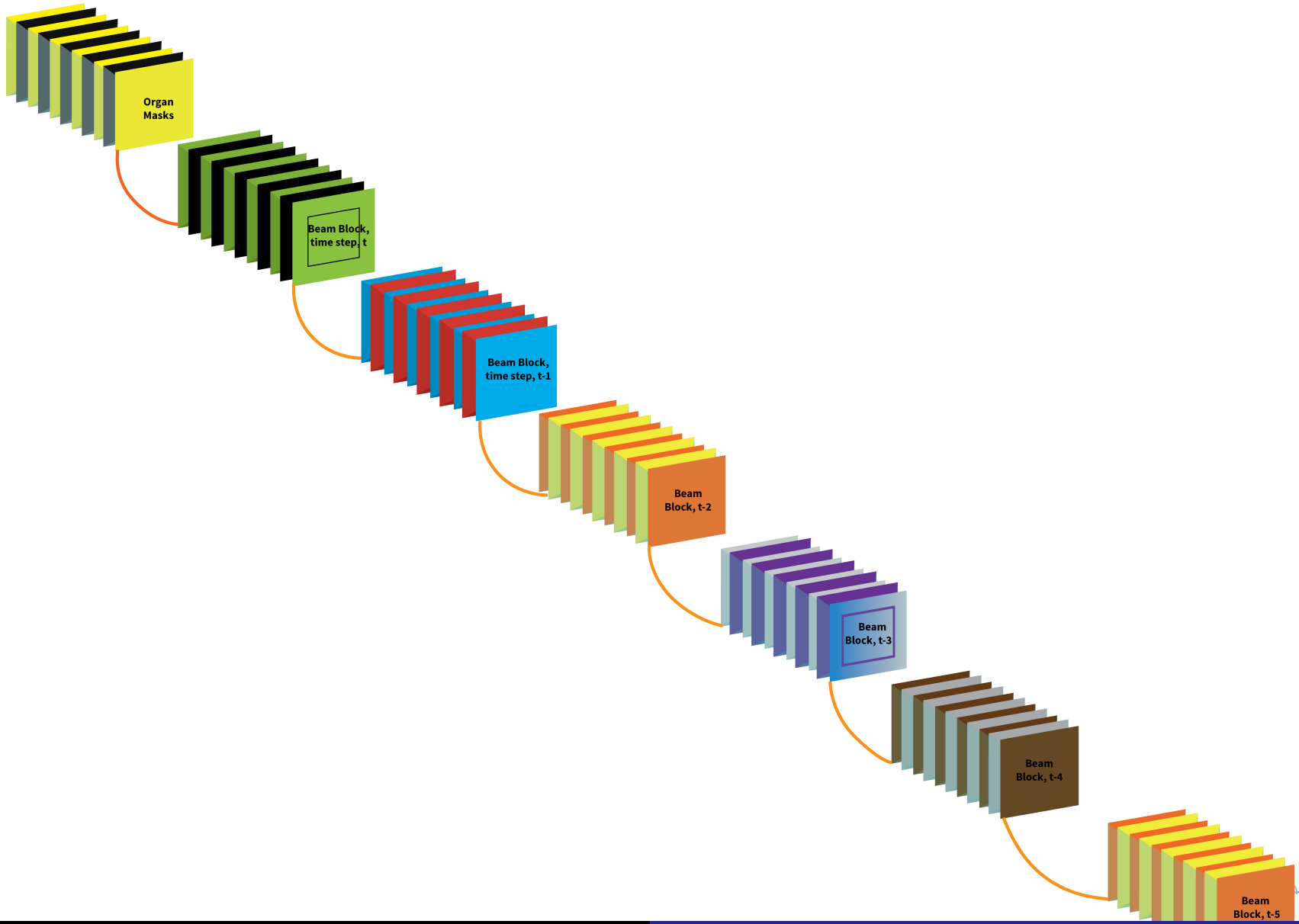
State encoding: prostate organ masks



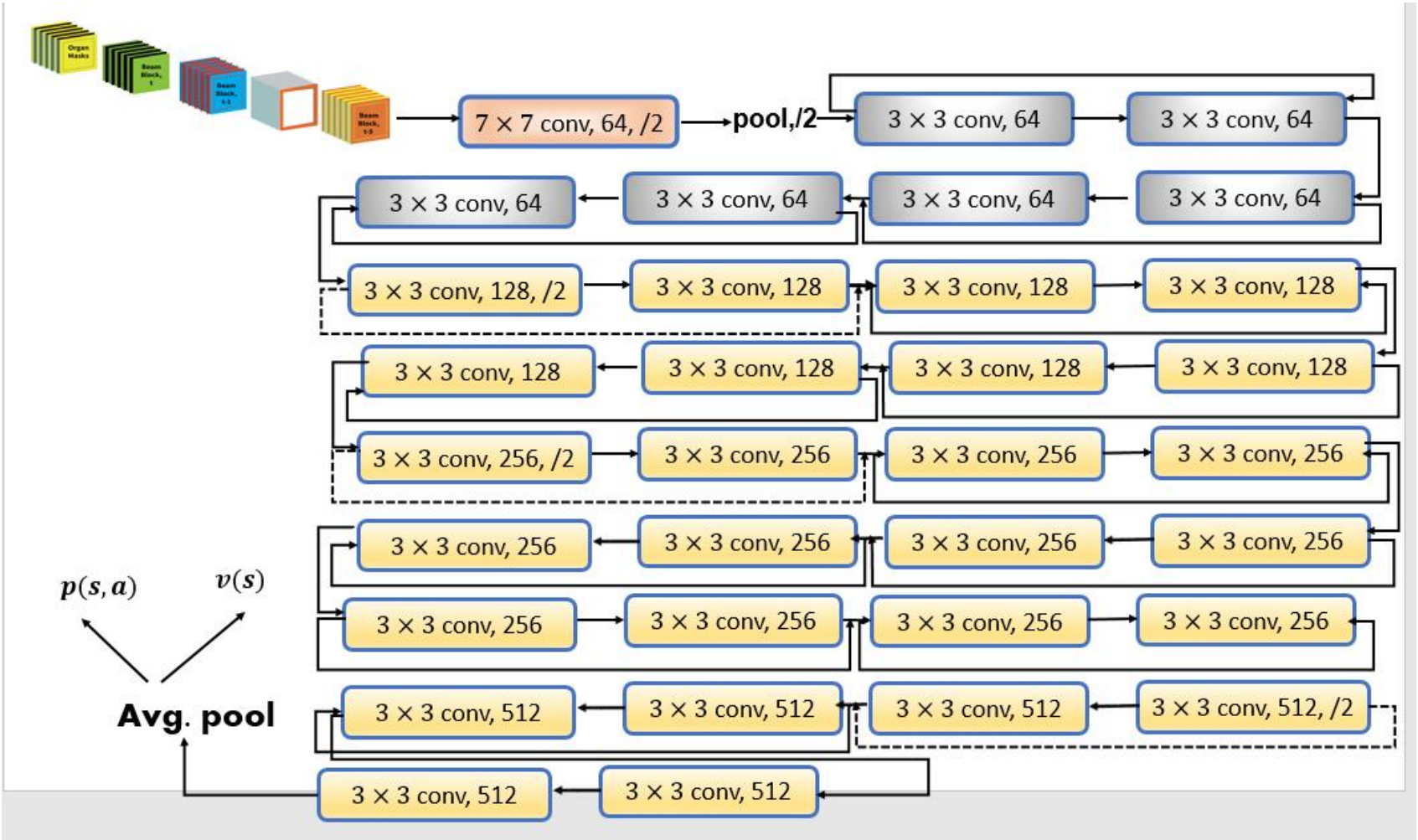
State representation: beam angles

10°	82°	166°
		

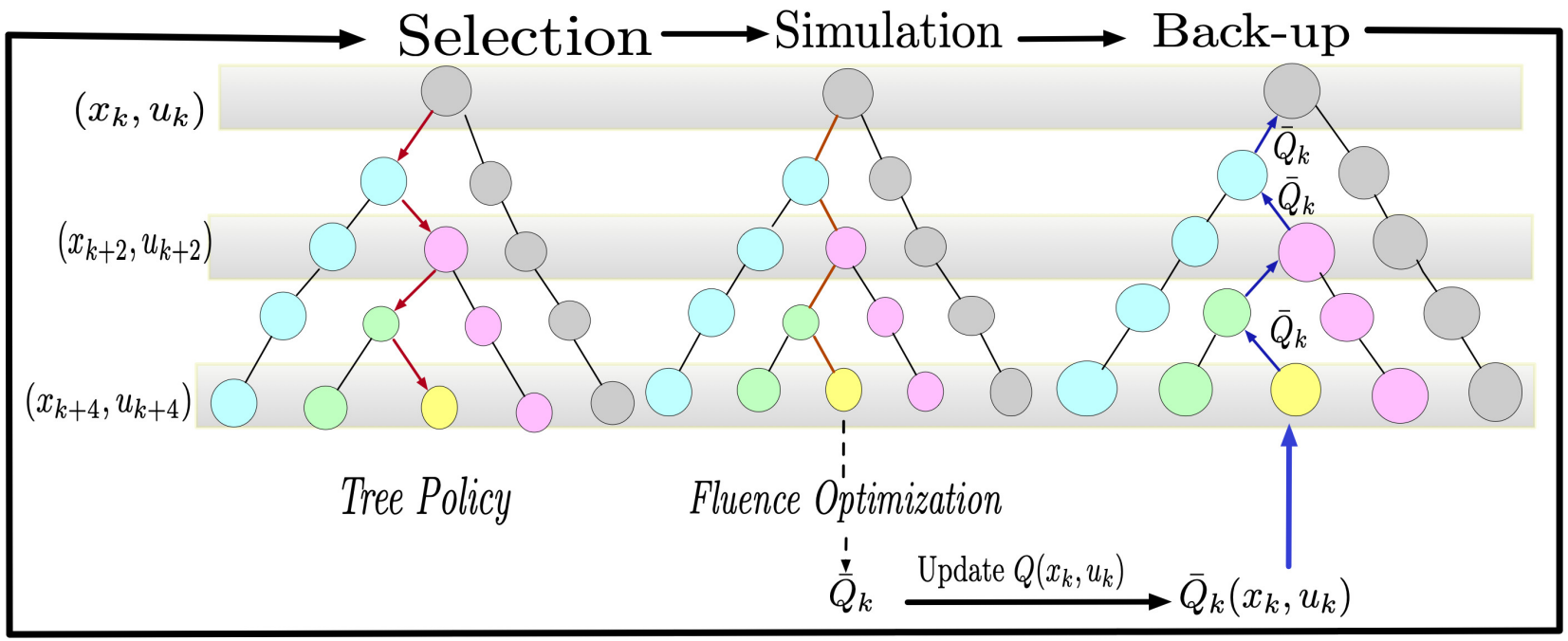
State representation



Two-player fictitious network play with ResNet



Tree Representation and Game Simulation



Tree Composition

Every **node** of the tree, $\boldsymbol{x}$, has the following fields:

- a pointer to the parent that led to it, $\boldsymbol{x}.p$;
- the beamlets, $\boldsymbol{x}_b$, stored at that node; $b = \{1, \ldots, m\}$;
- a set of move probabilities prior, $p(s, a)$;
- a pointer $\boldsymbol{x}.\boldsymbol{r}$, to the reward r_t, for the state $\boldsymbol{x}_t$;
- a pointer to the state-action value $Q(s, a)$ and its upper confidence bound $U(s, a)$;
- a visit count $N(s, a)$, that indicates the number of times that node was visited; and
- a pointer $\boldsymbol{x}.child_i$ to each of its children nodes.

Saddle Point Strategy Formulation

- **Saddle point strategies** for optimal control sequence pair $\{a_t^{p_1^*}, a_t^{p_2^*}\}$ recursively obtained by optimizing, $V_t(s, a)$

$$V_t^*(s) = Q_t^*(s_t, \pi_t^{p_1}, \pi_t^{p_2}) = \min_{\pi^{p_1} \in \Pi^{p_1}} \max_{\pi^{p_2} \in \Pi^{p_2}} Q_t^\star(s_t, \pi^{p_1}, \pi^{p_2})$$
$$\forall s_t \in \mathcal{S}, \pi^{p_1} \in \Pi^{p_1}, \pi^{p_2} \in \Pi^{p_2}.$$

such that

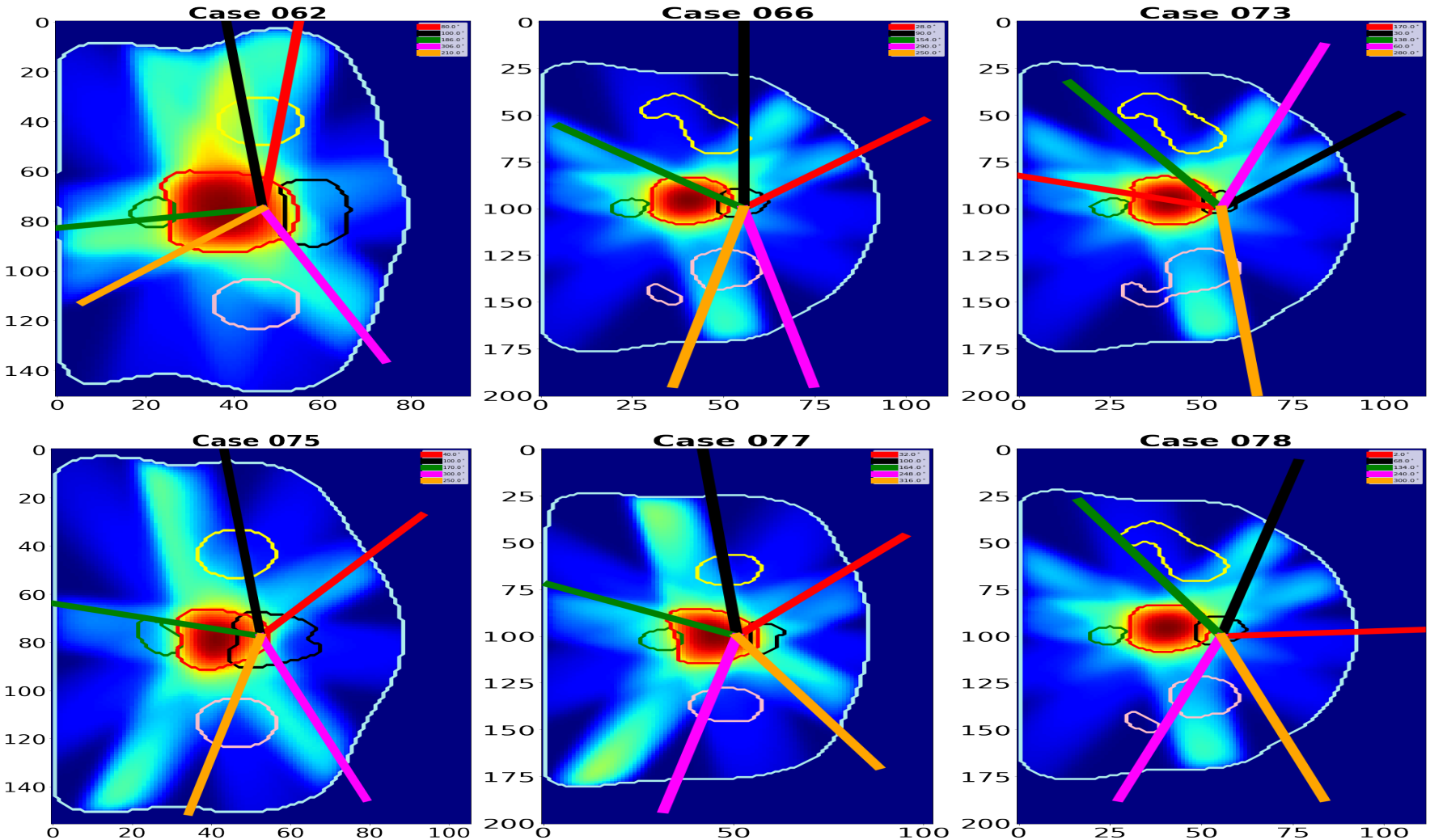
$$v_{p_1}^\star \leq v^\star \leq v_{p_2}^\star \quad \forall \{\pi_t^{p_1}, \pi_t^{p_2}\}_{0 \leq t \leq T}.$$

- p_1, p_2 respectively generating a **mixed strategy** via **averaging the outcome** of individual plays.

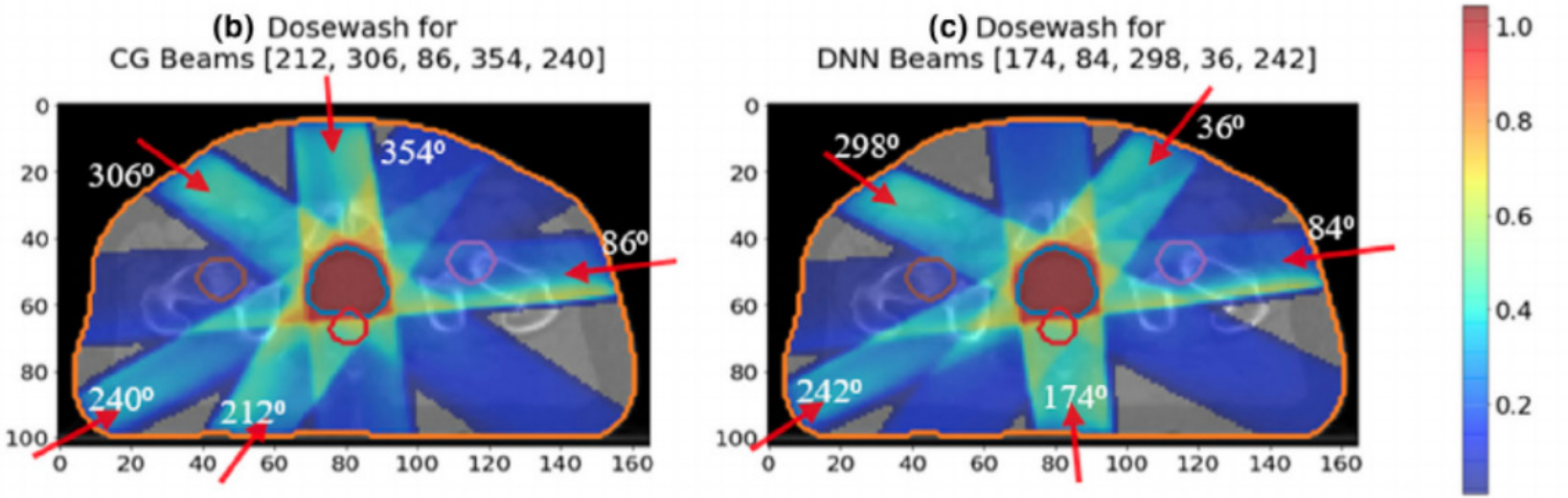
Training and Validation Loss

Average training (solid) and validation (dotted) loss function (MSE) values across six cross-validation folds for the network (blue) and full network.

BOO Results: Testing of self-play network



Column Generation vs Neural Network



Dose-Volume Histogram of CG vs DNN architectures [Sadeghnejad Barkousaraie, Azar and Ogunmolu, Olalekan and Jiang, Steve and Nguyen, Dan (2019)].

Conclusions

- Deep Neural Network optimizes network weights in a separate multiprocessing thread; Network outputs probabilities used to guide search;
- Sparse lookahead search builds tree with nodes labeled by state-action pairs in an alternating manner; sample rewards stored on edges connecting state-action with state nodes;
- Beam angles prediction takes between 2-3 minutes with MCTS vs. $\sim$ 60 seconds with Column Generation Pre-training.

Transition slide: Representation learning.

This page is left blank intentionally.

State representation in RL: credits

A. Koul

Y. Efroni

D. Misra

D. Foster

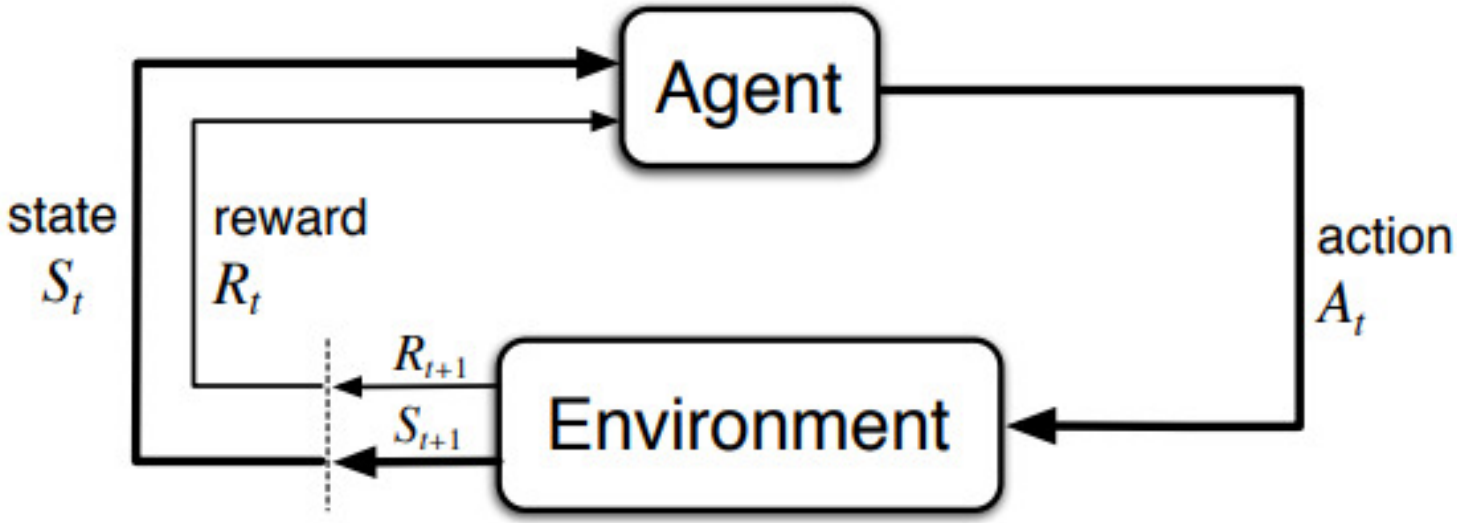
A. Lamb

M. Dudik

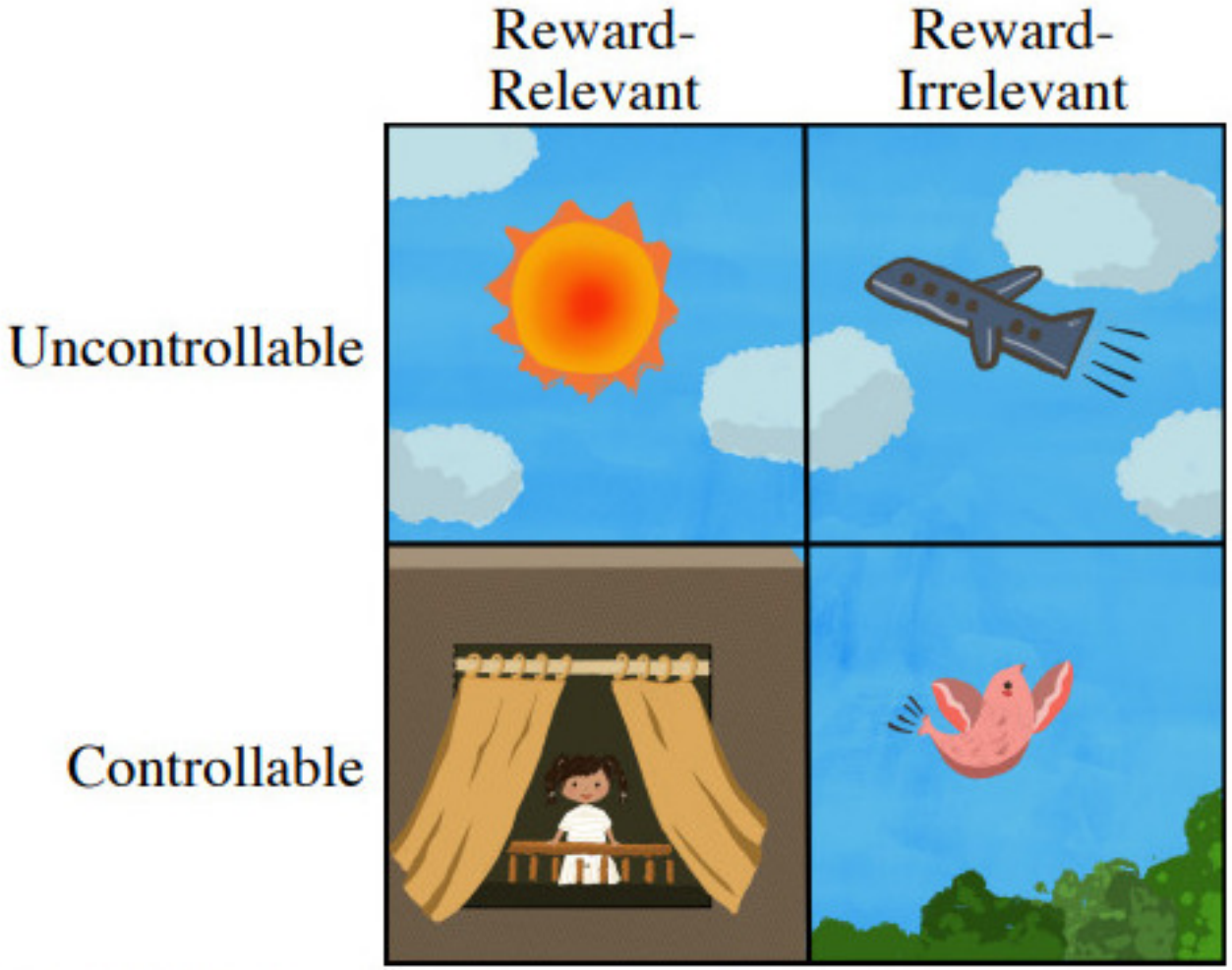
A. Krish.

J. Langford

Standard reinforcement learning

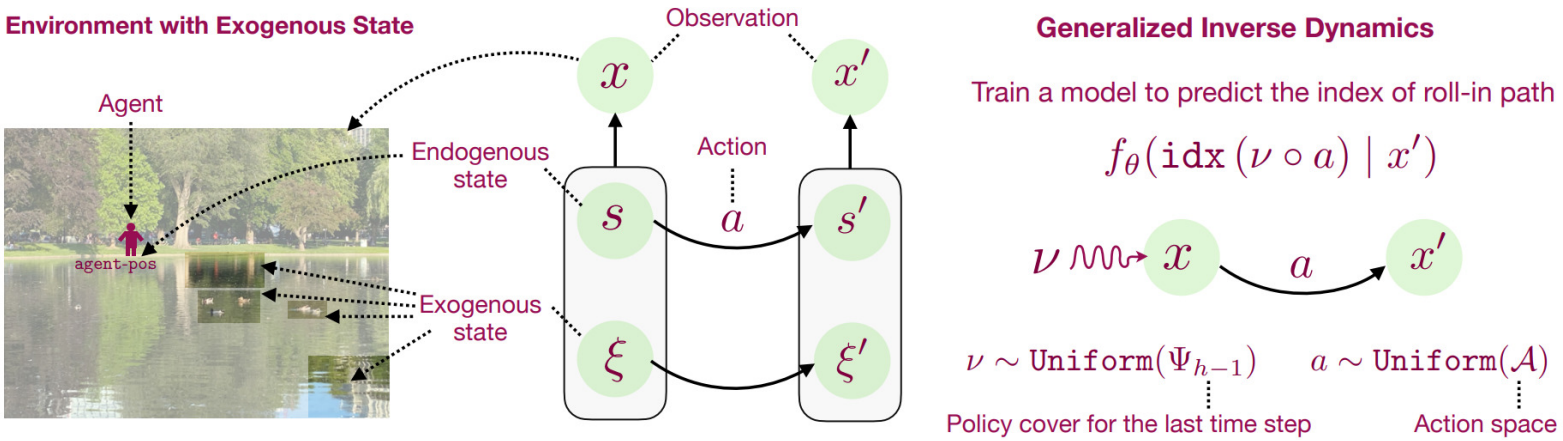


Without exogenous distractors



Source: (**?**)

Compact states without exogenous distractors



Source: (**?**).

Exo-MDP machinery

- Tuple $\mathcal{M} := (\mathcal{X}, \mathcal{Z}, \mathcal{A}, T, R, H)$
  - Starting distribution $\mu \in \Delta(\mathcal{Z})$;
  - Observations $\{x_h\}_{h=1}^{H} \in \mathcal{X}$ from $q : \mathcal{Z} \to \Delta(\mathcal{X})$;
  - Transitions, $T : \mathcal{Z} \times \mathcal{A} \to \Delta(\mathcal{S})$;
  - Rewards by $R : \mathcal{X} \times \mathcal{A} \to \Delta([0, 1])$

Exo-MDP machinery

- Trajectories: $(z_1, x_1, a_1, r_1, \cdots, z_H, a_H, r_H)$ from repeated interactions
  - $z_1 \sim \mu_1(\cdot)$,
  - $z_{h+1} \sim T(\cdot|z_h, a_h)$,
  - $x_h \sim q(\cdot|z_h)$ and
  - $r_h \sim R(x_h, a_h, x_{h+1})$ for all $h \in [H]$.
- Block MDP assumption: $\text{Supp}(q(\cdot|z)) = \{x \in \mathcal{X} | q(x|z) > 0\}$ for any z.

Exo-MDP machinery

- $\text{Supp}(q(\cdot|z_1)) \cap \text{Supp}(q(\cdot|z_2)) = \emptyset$ for all $z_1 \neq z_2$.
  - $a \sim \pi(z_h|x_h)$
  - Non-stationary episodic policies $\Pi_{NS} := \Pi^H \supseteq (\pi_1, \cdots, \pi_H)$;
  - Optimal policy $\pi^\star = \text{argmax} V_{\pi \in \Pi_{NS}}(\pi)$ for $V_{\pi \in \Pi_{NS}} = \sum_h 1^H r_h$.
- Latent states admits the form $z = (s, e)$, where $s \in \mathcal{S}$, $e \in E$ (**?**).
- $\mu(z) = \mu(s)\mu\xi$ and $T(z'|z, a) = T(s'|s, a)T_e(e'|e)$

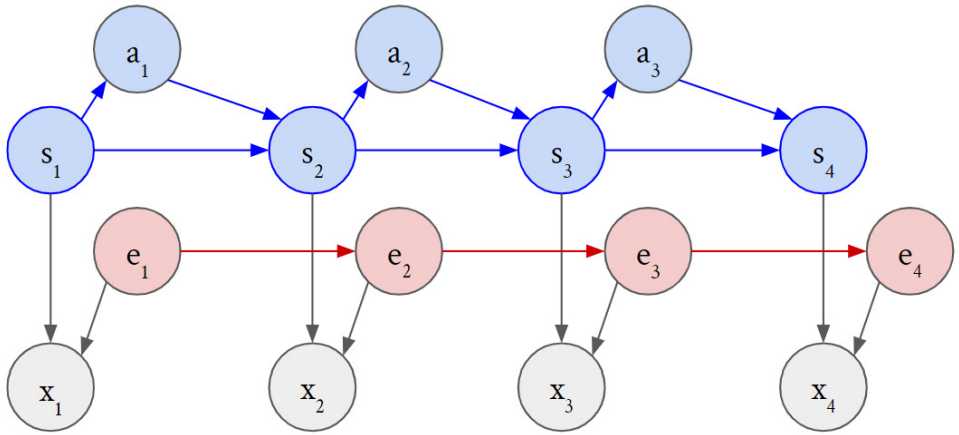
Literature comparison

Algorithms	PPE	OSSR	DBC	CDL	Denoised-MDP	1-Step Inverse	AC-State (Ours)
Exogenous Invariant State	✓	✓	✓	✓	✓	✓	✓
Exogenous Invariant Learning	✓	✓	✗	✗	✗	✓	✓
Flexible Encoder	✓	✗	✓	✗	✓	✓	✓
YOLO (No Resets) Setting	✗	✓	✓	✓	✓	✓	✓
Reward Free	✓	✓	✗	✓	✓	✓	✓
Control-Endogenous Rep.	✓	✓	✗	✓	✓	✗	✓

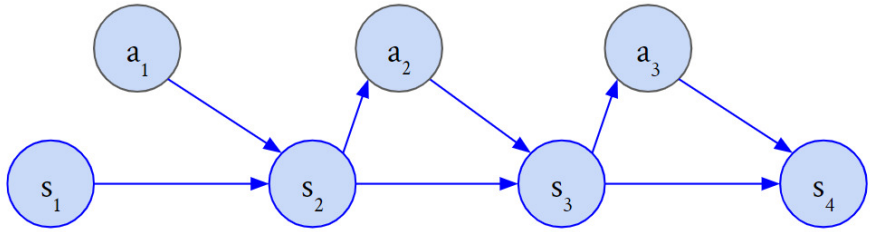
Emphasis on robustness to exogenous information. Comparison with baselines including PPE [3], OSSR [2], DBC [6] , Denoised MDP [5] and One-Step Inverse Models [4].

Rewards-agnostic state invariance

AC-State Discovers the smallest control-endogenous state s assuming factorized dynamics

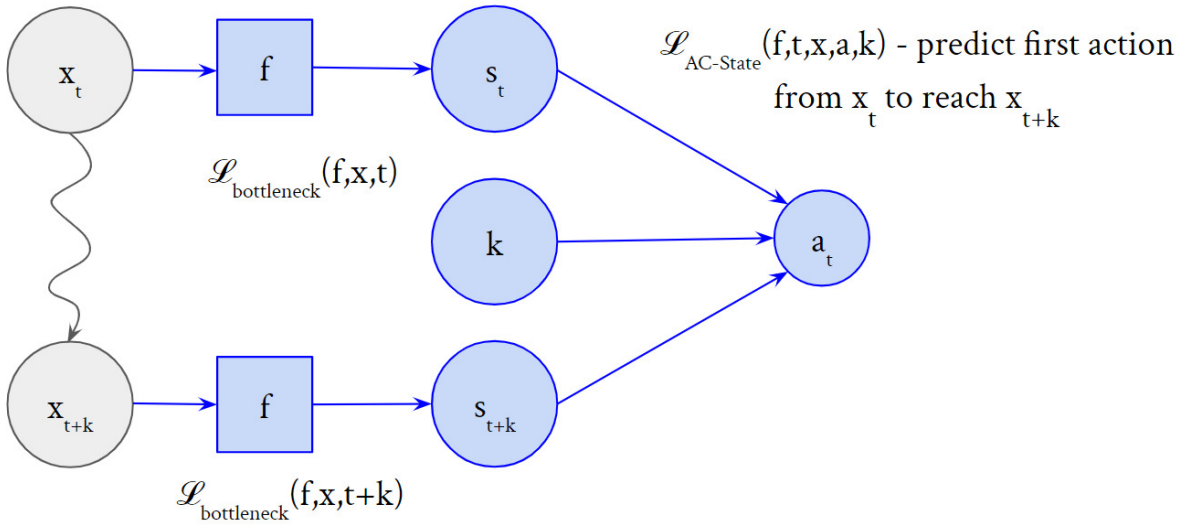


AC-State collects data with a single random action followed by a high-coverage endogenous policy for k-1 steps

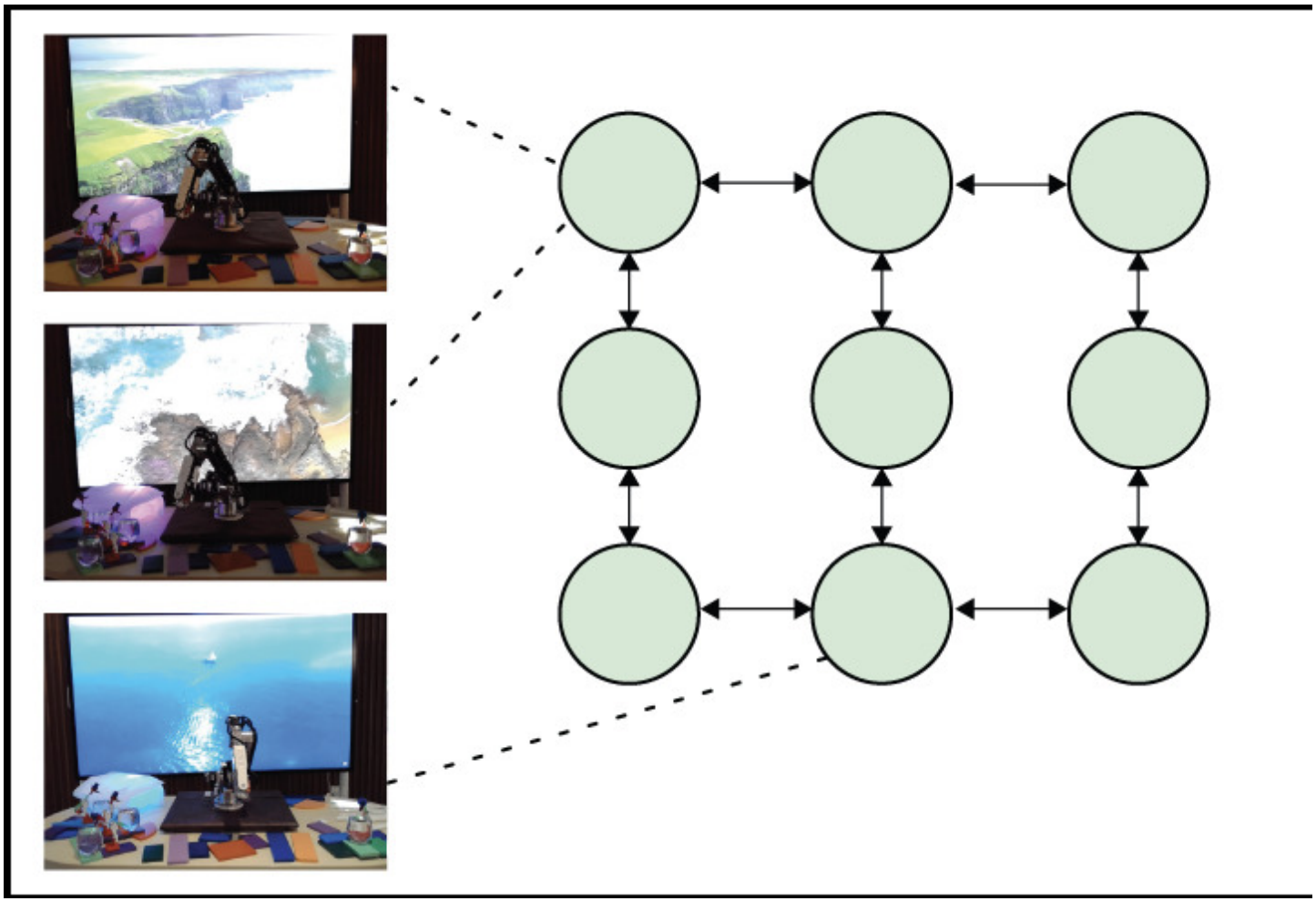


Rewards-agnostic state invariance

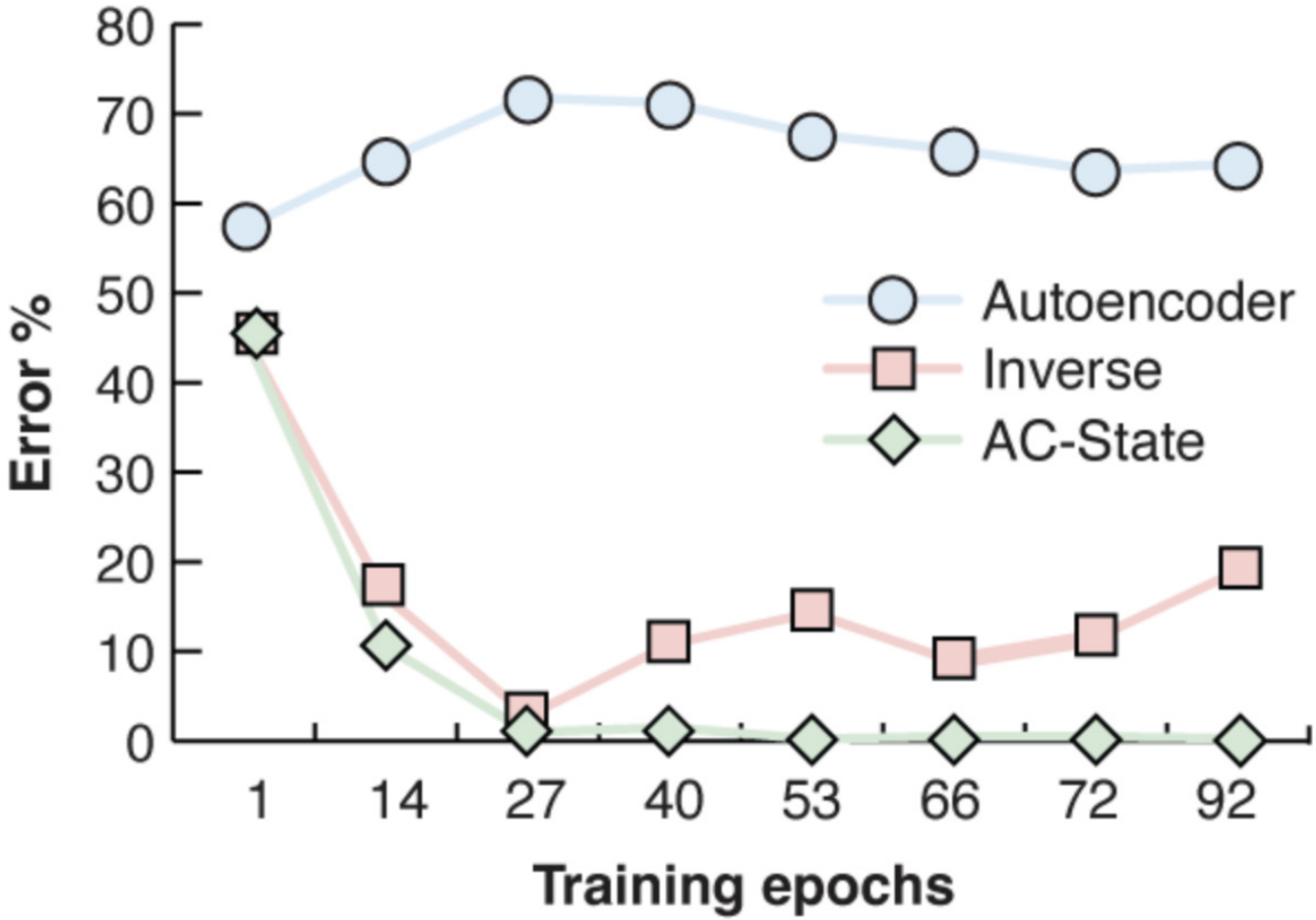
AC-State learns an encoder f for s = f(x) by optimizing a multi-step inverse model with a bottleneck



AC-State in action



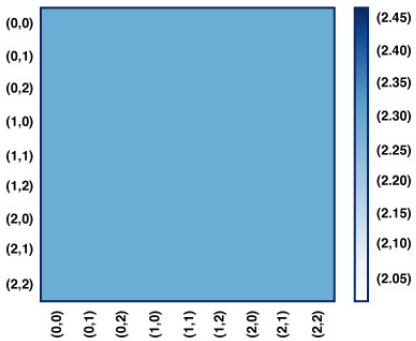
AC-State in action



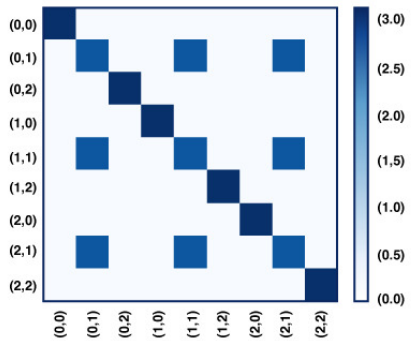
AC-State in action

Exogenous distractors riddance.

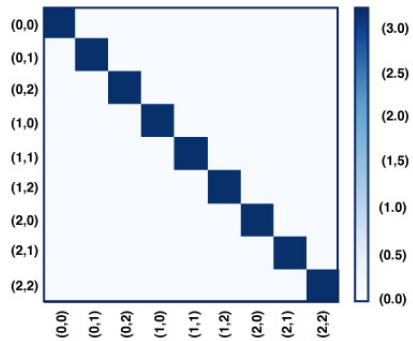
AC-State results



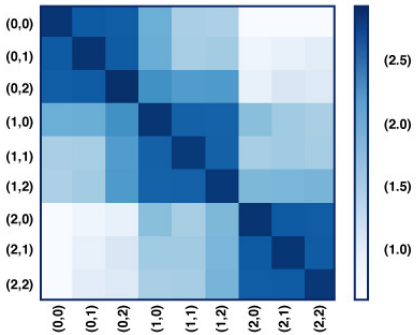
(a) Autoencoder (Theory worst-case)



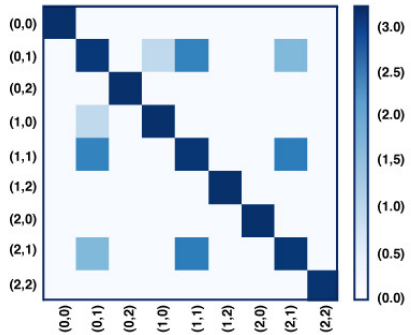
(b) Inverse (Theory worst-case)



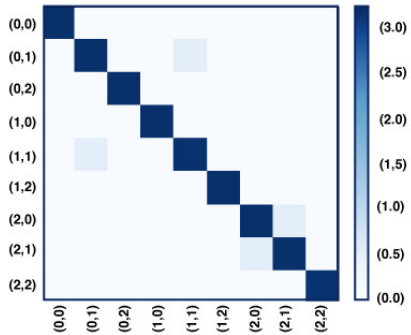
(c) AC-State (Theory worst-case)



(d) Autoencoder (Empirical)



(e) Inverse (Empirical)

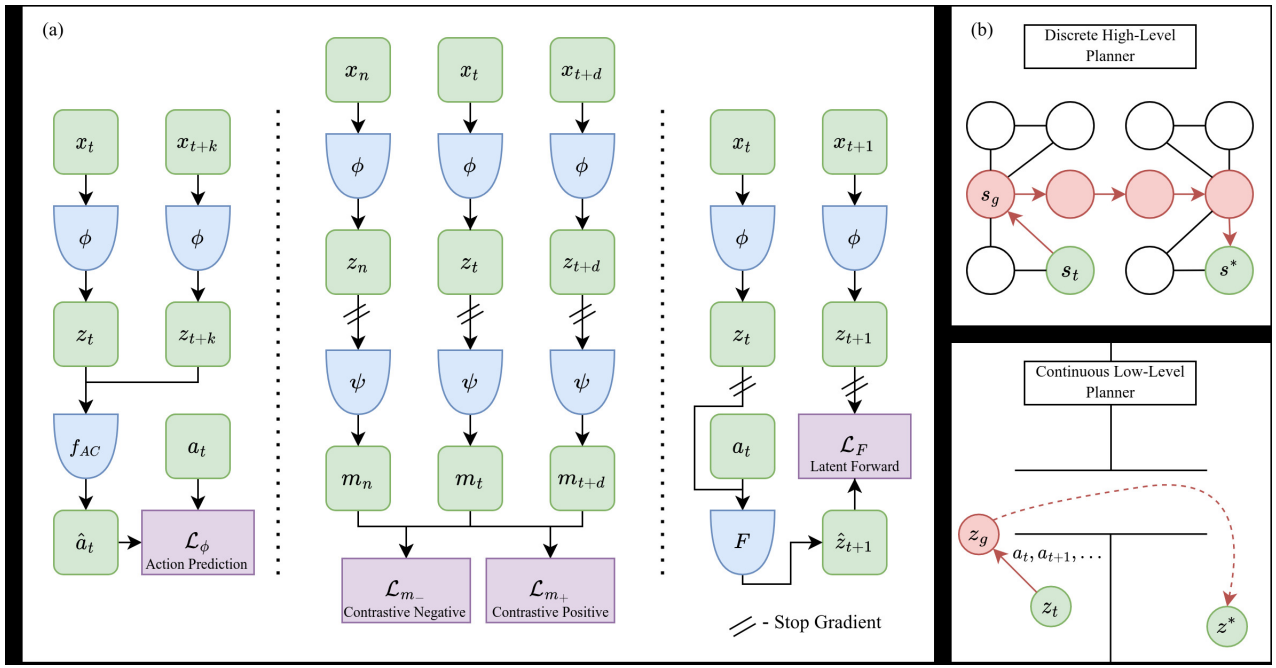


(f) AC-State (Empirical)

PCLAST: Agent plannable continuous latent states

PcLast: Discovering Plannable Continuous Latent States

Anurag Koul [*1] Shivakanth Sujit [*2 3 4] Shaoru Chen [1] Ben Evans [5] Lili Wu [1] Byron Xu [1] Rajan Chari [1]
Riashat Islam [3 6] Raihan Seraj [3 6] Yonathan Efroni [7] Lekan Molu [1] Miro Dudik [1] John Langford [1] Alex Lamb [1]



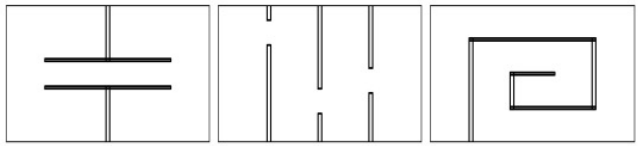
PCLAST algorithm

Algorithm 1 n-Level Planner

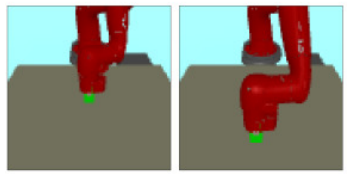
Require:
- Current observation x_t
- Goal observation x_{goal}
- Planning horizon H
- Encoder $\phi(\cdot)$
- PCLAST map $\psi(\cdot)$
- Latent forward dynamics $\delta(\cdot, \cdot)$
- Multi-Level discrete transition graphs $\{\mathcal{G}_i\}_{i=2}^{n}$

Ensure: Action sequence $\{a_i\}_{i=0}^{H-1}$

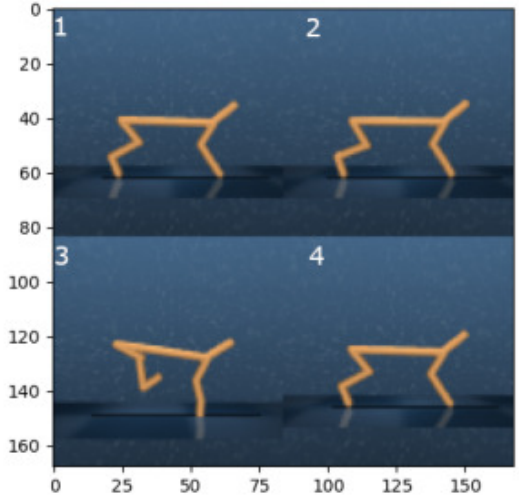
1: Compute current continuous latent state $\hat{s}_t = \phi(x_t)$ and target latent state $\hat{s}^* = \phi(x_{goal})$.
{See Appendix E for details of high-level planner and low-level planner.}
2: **for** $i = n, n-1, \ldots, 2$ **do**
3: $\hat{s}^* =$ high-level planner$(\hat{s}_t, \hat{s}^*, \mathcal{G}_i)$
{Update waypoint using a hierarchy of abstraction.}
4: **end for**
5: $\{a_i\}_{i=0}^{H-1} =$ low-level planner$(\hat{s}_t, \hat{s}^*, H, \delta, \psi)$
{Solve the trajectory optimization problem.}



(a) Hallway (b) Rooms (c) Spiral



(d) Sawyer Reach Environment



PCLAST results

METHOD	REWARD TYPE	HALLWAY	ROOMS	SPIRAL	SAWYER-REACH
PPO	DENSE	6.7 ± 0.6	7.5 ± 7.1	11.2 ± 7.7	$\mathbf{86.00 \pm 5.367}$
PPO + ACRO	DENSE	10.0 ± 4.1	23.3 ± 9.4	23.3 ± 11.8	84.00 ± 6.066
PPO + PCLAST	DENSE	$\mathbf{66.7 \pm 18.9}$	$\mathbf{43.3 \pm 19.3}$	$\mathbf{61.7 \pm 6.2}$	78.00 ± 3.347
PPO	SPARSE	1.7 ± 2.4	0.0 ± 0.0	0.0 ± 0.0	68.00 ± 8.198
PPO + ACRO	SPARSE	21.7 ± 8.5	5.0 ± 4.1	11.7 ± 8.5	$\mathbf{92.00 \pm 4.382}$
PPO + PCLAST	SPARSE	$\mathbf{50.0 \pm 18.7}$	$\mathbf{6.7 \pm 6.2}$	$\mathbf{46.7 \pm 26.2}$	82.00 ± 5.933
CQL	SPARSE	3.3 ± 4.7	0.0 ± 0.0	0.0 ± 0.0	32.00 ± 5.93
CQL + ACRO	SPARSE	15.0 ± 7.1	$\mathbf{33.3 \pm 12.5}$	$\mathbf{21.7 \pm 10.3}$	68.00 ± 5.22
CQL + PCLAST	SPARSE	$\mathbf{40.0 \pm 0.5}$	23.3 ± 12.5	20.0 ± 8.2	$\mathbf{74.00 \pm 4.56}$
RIG	NONE	0.0 ± 0.0	0.0 ± 0.0	3.0 ± 0.2	$\mathbf{100.0 \pm 0.0}$
RIG + ACRO	NONE	$\mathbf{15.0 \pm 3.5}$	$4.0 \pm 1.$	$\mathbf{12.0 \pm 0.2}$	100.0 ± 0.0
RIG + PCLAST	NONE	10.0 ± 0.5	4.0 ± 1.8	10.0 ± 0.1	90.0 ± 5
LOW-LEVEL PLANNER + PCLAST	NONE	86.7 ± 3.4	69.3 ± 3.4	50.0 ± 4.3	$\pm$
n-LEVEL PLANNER + PCLAST	NONE	$\mathbf{97.78 \pm 4.91}$	$\mathbf{89.52 \pm 10.21}$	$\mathbf{89.11 \pm 10.38}$	95.0 ± 1.54

Conclusion

- Questions?
  - Email: patlekano [@] gmail [dot] com

Publications

Johannes Heinrich, Marc Lanctot, and David Silver. Fictitious self-play in extensive-form games. In *International Conference on Machine Learning*, pages 805–813, 2015.

Sadeghnejad Barkousaraie, Azar and Ogunmolu, Olalekan and Jiang, Steve and Nguyen, Dan. A Fast Deep Learning Approach for Beam Orientation Selection Using Supervised Learning with Column Generation on IMRT Prostate Cancer Patients. *Medical Physics, American Association of Physicists in Medicine*, 2019.

Dimitris Bertsimas, Valentina Cacchiani, David Craft, and Omid Nohadani. A Hybrid Approach To Beam Angle Optimization In Intensity-modulated Radiation Therapy. *Computers & Operations Research*, 40(9):2187–2197, 2013.