One Model, Any CSP: Graph Neural Networks as Fast Global Search Heuristics for Constraint Satisfaction

Jan Tönshoff, Berke Kisin, Jakob Lindner, Martin Theisen, Martin Grohe

RWTH AACHEN UNIVERSITY

Abstract

We propose ANYCSP[1], a universal Graph Neural Network architecture which can be trained as an end-2-end search heuristic for any Constraint Satisfaction Problem (CSP). Our architecture can be trained unsupervised with policy gradient descent to generate problem specific heuristics for any CSP in a purely data driven manner. The approach is based on a novel graph representation for CSPs that is both generic and compact and enables us to process every possible CSP instance with one GNN, regardless of constraint arity, relations or domain size. Unlike previous RL-based methods, we operate on a global search action space and allow our GNN to modify any number of variables in every step of the stochastic search.

[1] Tönshoff, Jan, et al. "One model, any CSP: Graph neural networks as fast global search heuristics for constraint satisfaction.", IJCAI-23 (2023)

Constraint Satisfaction Problems

Generic framework for modelling discrete optimization problems. Well known CSPs are SAT, graph coloring and maximum cut.

CSP-Instance $\mathcal{I} = (\mathcal{X}, \mathcal{C}, \mathcal{D})$:

- Variables $\mathcal{X} = \{X_1, \dots, X_n\}$, Domains $\mathcal{D} = \{\mathcal{D}(X_1), \dots, \mathcal{D}(X_n)\}$
- Constraints $\mathcal{C} = \{C_1, \dots, C_m\}$ of the form $C = (s^C, R^C)$:

$$s^C = (X_1^C, \dots, X_\ell^C) \qquad R^C \subseteq \mathcal{D}(X_1^C) \times \dots \times \mathcal{D}(X_\ell^C)$$

Assignment $\alpha(X) \in \mathcal{D}(X)$: $\alpha \models C \Leftrightarrow (\alpha(X_1^C), \dots, \alpha(X_\ell^C)) \in R^C$

Constraint Value Graph

CSP Instance $\mathcal{I}$:

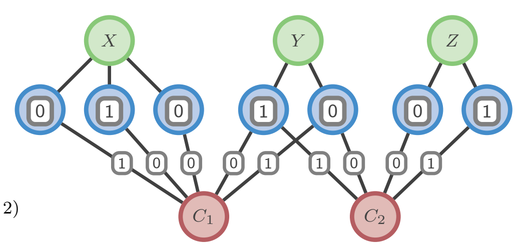
$\mathcal{X} = \{X, Y, Z\}$

$\mathcal{D}(X) = \{1, 2, 3\}$
$\mathcal{D}(Y) = \{1, 2\}$
$\mathcal{D}(Z) = \{1, 2\}$

$C_1 : X \leq Y$
$C_2 : Y \neq Z$

Assignment $\alpha = (2, 1, 2)$

$G(\mathcal{I}, \alpha)$:



Training

Quality of assignment α for instance $\mathcal{I} = (\mathcal{X}, \mathcal{C}, \mathcal{D})$:

$$Q_\mathcal{I}(\alpha) = |\{C \in \mathcal{C} \ : \ \alpha \models C\}| / |\mathcal{C}|$$

Assume training distribution of CSP instances Ω. Objective:

$$\theta^* = \arg\max_\theta \mathop{\mathbb{E}}_{\substack{\mathcal{I} \sim \Omega \\ \alpha \sim \pi_\theta(\mathcal{I})}} \left[\sum_{t=1}^{T} \gamma^{t-1} r^{(t)} \right]$$

Reward in iteration t encourages iterative improvements:

$$r^{(t)} = \begin{cases} 0 & \text{if } Q_\mathcal{I}(\alpha^{(t)}) \leq q^{(t)} \\ Q_\mathcal{I}(\alpha^{(t)}) - q^{(t)} & \text{if } Q_\mathcal{I}(\alpha^{(t)}) > q^{(t)} \end{cases}$$

with $q^{(t)} = \max_{0 \leq t' < t} Q_\mathcal{I}(\alpha^{(t')})$. Optimize with policy gradients (REINFORCE):

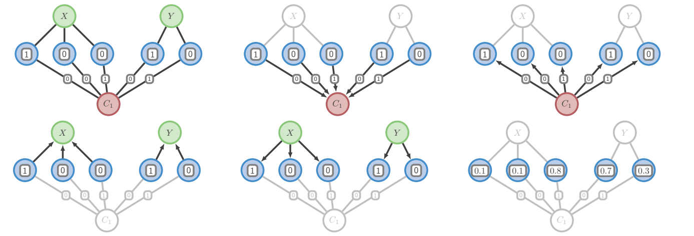
$$\nabla\theta = -\nabla \sum_{t=1}^{T} G_t \log \mathrm{P}(\alpha^{(t)} | \varphi_\theta^{(t)}), \qquad G_t = \sum_{k=t}^{T} \gamma^{k-t} r^{(k)}$$

Policy GNN π_θ

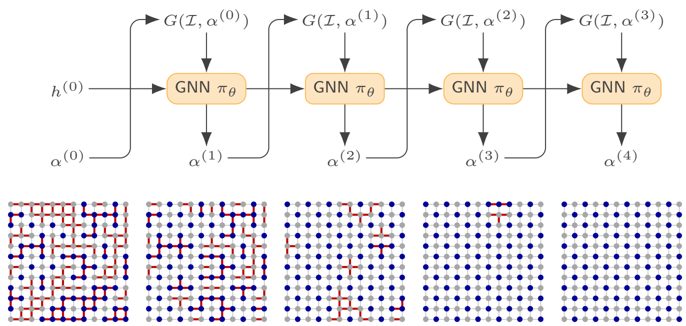
A trainable policy π_θ maps the current constraint value graph to a new soft assignment $\varphi^{(t)}$ and updates recurrent states $h^{(t)}$:

$$\pi_\theta : G(\mathcal{I}, \alpha^{(t-1)}), h^{(t-1)} \mapsto \varphi^{(t)}, h^{(t)}$$

π_θ is a heterogeneous GNN based on message passing:



Global Search



REINFORCE vs Actor-Critic

A **critic** $\mathfrak{c}$ learns to estimate gain G_t from recurrent state $h^{(t)}$, which speeds up learning through a baseline (replacing G_t in policy gradient):

$$A_t = G_t - \mathfrak{c}(h^{(t)})$$

and temporal difference learning (replacing G_t everywhere):

$$G_t^{(n)} = \gamma^{n+1} \mathfrak{c}(h^{(t+n+1)}) + \sum_{k=t}^{t+n} \gamma^{k-t} r^{(k)},$$

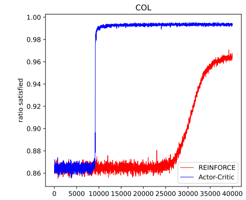
$$G_t(\lambda) = \lambda^{T+1} G_t + (1 - \lambda) \sum_{n=0}^{T} \lambda^n G_t^{(\min(n, T-t))}$$

Geometric reward including quality and improvement:

$$r_g^{(t)} = \sqrt{(Q_\mathcal{I}(\alpha^{(t)}) - Q_\mathcal{I}(\alpha^{(0)})) r^{(t)}}$$

Entropy regularization incentivizes exploration:

$$r_e^{(t)} = r_g^{(t)} - \sigma \log \mathrm{P}(\alpha^{(t)} \mid \varphi_\theta^{(t)})$$

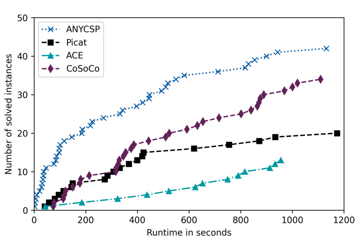


Results

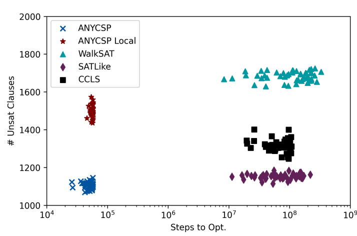
- earlier, faster, and more robust learning
- better in distribution and on decision problems
- less stable

Results

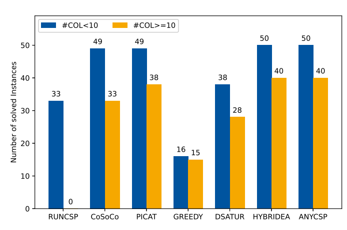
Model RB



MAX-5-SAT



Graph Coloring



MAXCUT

Method	$\lvert V\rvert$=800	$\lvert V\rvert$=1K	$\lvert V\rvert$=2K	$\lvert V\rvert \geq$3K
Greedy	411.44	359.11	737.00	774.25
SDP	245.44	229.22	-	-
RUNCSP	185.89	156.56	357.33	401.00
ECO-DQN	65.11	54.67	157.00	428.25
ECORD	8.67	8.78	39.22	187.75
ANYCSP	**1.22**	**2.44**	**13.11**	**51.63**

Jan Tönshoff
RWTH Aachen University
toenshoff@informatik.rwth-aachen.de