部材の逐次的な付加・除去過程を訓練した強化学習エージェントによる平面トラスの位相最適化

Topology Optimization of Planar Trusses by Reinforcement Learning Agents Trained for the Sequential Member Addition and Removal Process

○林　和希[*1]，大崎　純[*2]
Kazuki HAYASHI[*1] and Makoto OHSAKI[*2]

*1 京都大学工学研究科建築学専攻　助教　博士(工学)
Assistant Professor, Department of Architecture and Architectural Engineering, Kyoto University, Ph.D.
*2 京都大学工学研究科建築学専攻　教授　博士(工学)
Professor, Department of Architecture and Architectural Engineering, Kyoto University, Ph.D.

キーワード：構造最適化; 位相最適化; グラフ埋め込み; 強化学習; 機械学習
Keywords: Structural optimization; topology optimization; graph embedding; reinforcement learning; machine learning.

1. 序

著者らはこれまで，不規則な接続関係を有するデータ構造から部材の特徴量を抽出するグラフ埋め込みと深層強化学習 (Deep-Q Network, DQN) [1]を組み合わせた強化学習手法を提案した[2]．しかし，その手法は密な部材配置から逐次的に不要部材を除去する一方向的な設計プロセスに限定されていた．強化学習エージェントが部材除去だけでなく部材付加もできるようになれば，任意の初期位相に対してエージェントを適用した設計変更が可能となる．

本研究では，応力・変位制約下で部材総体積を最小化することを目的とする平面トラスの位相最適化問題に対して，部材の付加と除去を同時に取り扱えるエージェントを定義し，グラフ埋め込みと DQN の複合手法による訓練を行う．さらに，DQN の改善手法である Dueling Network [3], QR-DQN [4], Double Q-learning [5], Multi-step learning [6]，Prioritized Experienced Replay [7]を導入して学習効率を向上させる．

2. 強化学習タスク

2.1. 構造最適化問題

節点位置と初期グランドストラクチャ（GS），支持・荷重条件を与え，存在する（初期 GS から除去されていない）部材 i の荷重条件 j での応力 $\sigma_{i,j}$ の絶対値が上限値 $\bar{\sigma}$ を超えず，存在する（1 本以上の存在部材が接続されている）節点 i の荷重条件 j での k 方向の変位 $u_{i,j,k}$ の絶対値が $\bar{u}$ を超えないように部材総体積 V を最小化するトラスの位相を求める次のような最適化問題を考える．

$$\text{minimize}\ \ \mathrm{V}(\mathbf{A}) \tag{1a}$$

$$\text{subject to}\ \max_{i\in\Omega_{\mathrm{m}},j\in\{1,2\}}\left(\left|\sigma_{i,j}\right|/\bar{\sigma}\right)\le 1 \tag{1b}$$

$$\max_{i\in\Omega_{\mathrm{n}},j\in\{1,2\},k\in\{1,2\}}\left(\left|u_{i,j,k}\right|/\bar{u}\right)\le 1 \tag{1c}$$

$$A_i\in\left\{\bar{A}\times 10^{-5},\bar{A}\right\} \tag{1d}$$

Ω_{m}，Ω_{n} はそれぞれ存在する部材の集合，存在する節点の集合である．最適化途中で剛性行列が特異となることを防ぐため，除去したと見なす部材には式(1d)のように存在する部材の断面積 $\bar{A}$ を 10^{-5} 倍した微小断面積を与える．

2.2. 強化学習タスクへの変換

部材付加と除去が混在する状況では強化学習エージェントが同一部材の付加・除去を繰り返すなど冗長な行動を取ることで学習が困難になるおそれがある．したがって，以下の手順で冗長な行動を制約して部材付加・除去のシミュレーションを行う．図 1 にプロセスを併せて図示する．

① まず，制約を満たさない疎な初期位相からスタートし，制約を満たすまで最も効率の良いと推定した部材を逐次付加する．
② 制約を満たしたら，今度は制約を満たさなくなるまで最も効率の良い部材を除去する．このとき，接続部材数が 1 の不安定節点（支持点・載荷点除く）に接続する部材も併せて除去する．
③ 制約超過した直前の状態をエージェントが生成した解と見なす．

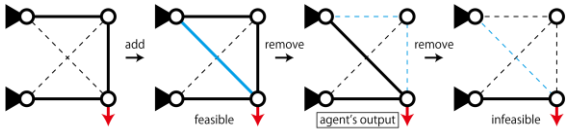

図 1　強化学習タスクにおける部材付加・除去過程

日本建築学会情報システム技術委員会
第44回情報・システム・利用・技術シンポジウム論文集，319–322，2021年12月，京都
Proceedings of the 44th Symposium on Computer Technology of Information, Systems and Applications，AIJ，319–322，Dec.，2021，Kyoto

3. グラフ埋め込みと深層強化学習

3.1. グラフ埋め込みによる特徴量抽出

節点数を n_{n} とし，特徴量抽出のために必要な各節点の入力値 $\hat{\mathbf{v}} = \left[\mathbf{v}_1,\cdots,\mathbf{v}_{n_{\mathrm{n}}}\right] \in \mathbb{R}^{3\times n_{\mathrm{n}}}$ と各部材の入力値 $\hat{\mathbf{w}} = \left[\mathbf{w}_1,\cdots,\mathbf{w}_{n_{\mathrm{m}}}\right] \in \mathbb{R}^{7\times n_{\mathrm{m}}}$ を表 1, 2 にそれぞれ定義する．これらの入力をトラスの状態 $s = \{\hat{\mathbf{v}},\hat{\mathbf{w}}\}$ とみなす．

表 1: 節点 k の入力値 $\mathbf{v}_k \in \mathbb{R}^3$

index	入力値の説明
1	ピン支持点なら 1，それ以外は 0
2	荷重条件 1 で節点kに作用するx方向荷重 [kN]
3	荷重条件 2 で節点kに作用するy方向荷重 [kN]

表 2: 部材 i の入力値 $\mathbf{w}_i \in \mathbb{R}^7$

index	入力値の説明
1	全部材が存在する場合の荷重条件 1 での応力比
2	全部材が存在する場合の荷重条件 2 での応力比
3	存在部材は 1，除去部材は 0
4	部材除去の行動が適用可能な場合 1，その他 0
5	部材付加の行動が適用可能な場合 1，その他 0
6	現在の位相における荷重条件 1 での応力比
7	現在の位相における荷重条件 2 での応力比

グラフ埋め込みは節点とエッジからなるデータ（グラフ）に対して特徴量を抽出する手法の総称である．ここではエッジの特徴量を図 2 の概念に基づいて抽出する[2].

抽出する部材特徴量の次元を n_{f} とする．線形変換を行うための学習可能なパラメータ $\boldsymbol{\theta}_1 \in \mathbb{R}^{n_{\mathrm{f}}\times 7}$，$\boldsymbol{\theta}_2 \in \mathbb{R}^{n_{\mathrm{f}}\times n_{\mathrm{f}}}$，$\boldsymbol{\theta}_3 \in \mathbb{R}^{n_{\mathrm{f}}\times 3}$，$\boldsymbol{\theta}_4 \in \mathbb{R}^{n_{\mathrm{f}}\times n_{\mathrm{f}}}$，$\boldsymbol{\theta}_5 \in \mathbb{R}^{n_{\mathrm{f}}\times n_{\mathrm{f}}}$，$\boldsymbol{\theta}_6 \in \mathbb{R}^{n_{\mathrm{f}}\times n_{\mathrm{f}}}$ を用いて，部材の特徴行列 $\hat{\boldsymbol{\mu}}_{(t_u)} \in \mathbb{R}^{n_{\mathrm{f}}\times n_{\mathrm{m}}}$ を次式で更新する．

$$\hat{\boldsymbol{\mu}}_{(1)} = \boldsymbol{\theta}_1\hat{\mathbf{w}} + \boldsymbol{\theta}_2\left(\varphi\left(\boldsymbol{\theta}_3\hat{\mathbf{v}}\right)\right)\mathbf{C}_{\mathrm{A}}^{\mathrm{T}} \tag{2a}$$

$$\hat{\boldsymbol{\mu}}_{(t_u+1)} = \boldsymbol{\theta}_4\hat{\boldsymbol{\mu}}_{(t_u)} + \boldsymbol{\theta}_5\sum_{k=1}^{2}\frac{\varphi\left(\boldsymbol{\theta}_6\left(\mathbf{C}_k\mathbf{C}_{\mathrm{A}}^{\mathrm{T}}\hat{\boldsymbol{\mu}}^{(t_u)\mathrm{T}}\right)^{\mathrm{T}}\right)}{\mathbf{n}_k^{\mathrm{c}}} \tag{2b}$$

$\mathbf{C}_{\mathrm{A}}$ は有向グラフの接続行列 $\mathbf{C} \in \mathbb{R}^{n_{\mathrm{m}}\times n_{\mathrm{n}}}$ の各成分について絶対値をとる行列，$\mathbf{C}_1$ は接続行列の–1 の値をとる成分が 1 で他の成分が 0 の行列，$\mathbf{C}_2$ は接続行列の 1 の値をとる成分が 1 で他が 0 の行列であり，$\mathbf{C}_1 + \mathbf{C}_2 = \mathbf{C}_{\mathrm{A}}$ の関係にある．$\mathbf{n}_k^{\mathrm{c}} \in \mathbb{R}^{n_{\mathrm{f}}\times n_{\mathrm{m}}}$ は各部材の k 端に接続する部材数を格納した行ベクトルを列方向に n_{f} 個並べた行列である．φ は負の入力に対しては 0.2 倍，0 以上の入力に対してはそのままの値を出力する Leaky ReLU と呼ばれる活性化関数を用いる．直接隣接していない部材の寄与も考慮するため，以下では $\hat{\boldsymbol{\mu}} = \hat{\boldsymbol{\mu}}_{(3)}$ を現状態における各部材の特徴量と見なす．

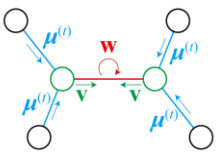

図 2 グラフ埋め込みの演算式(2)の概念図

3.2. 特徴量を用いた行動価値の計算

本研究では，各部材を除去/付加する設計変更を考慮するため，最大で $2n_{\mathrm{m}}$ 個の行動が存在する．トラスの各部材の現状態を表現した特徴行列 $\hat{\boldsymbol{\mu}}$ から各部材に設計変更を行う行動価値 $\hat{\mathbf{Q}} \in \mathbb{R}^{2\times n_{\mathrm{m}}}$ を $\boldsymbol{\theta}_7 \in \mathbb{R}^{n_{\mathrm{f}}\times 1}$，$\boldsymbol{\theta}_8 \in \mathbb{R}^{n_{\mathrm{f}}\times 2}$ を用いてまとめて次式で計算する．

$$\hat{\mathbf{Q}}\left(\hat{\boldsymbol{\mu}}\right) = V\left(\hat{\boldsymbol{\mu}}\right) + \hat{\mathbf{A}}\left(\hat{\boldsymbol{\mu}}\right) - \mathrm{mean}\left(\hat{\mathbf{A}}\left(\hat{\boldsymbol{\mu}}\right)\right) \tag{3a}$$

$$V\left(\hat{\boldsymbol{\mu}}\right) = \boldsymbol{\theta}_7^{\mathrm{T}}\hat{\boldsymbol{\mu}}_{\Sigma} \tag{3b}$$

$$\hat{\mathbf{A}}\left(\hat{\boldsymbol{\mu}}\right) = \boldsymbol{\theta}_8^{\mathrm{T}}\hat{\boldsymbol{\mu}} \tag{3c}$$

$\hat{\boldsymbol{\mu}}_{\Sigma} \in \mathbb{R}^{n_{\mathrm{f}}\times n_{\mathrm{m}}}$ は $\hat{\boldsymbol{\mu}}$ の各行の和をとった列ベクトルであり，トラス全体の特徴量とみなせる．$\mathrm{mean}\left(\hat{\mathbf{A}}\left(\hat{\boldsymbol{\mu}}\right)\right)$ は $\hat{\mathbf{A}}\left(\hat{\boldsymbol{\mu}}\right)$ の各行の平均をとった列ベクトルを行方向に n_{m} 個並べた行列である．式(3)では，行動価値の推定に状態価値 $\hat{\mathbf{V}}\left(\hat{\boldsymbol{\mu}}\right)$ とアドバンテージ $\hat{\mathbf{A}}\left(\hat{\boldsymbol{\mu}}\right) - \mathrm{mean}\left(\hat{\mathbf{A}}\left(\hat{\boldsymbol{\mu}}\right)\right)$ の和を用いる Dueling Network を用いることで，どの行動をとっても次状態や報酬に差違がない場合の推定精度を改善している[3].

ここではさらに，分布強化学習手法の 1 つである QR-DQN [4]を用いて式(3)を改良する．DQN では行動価値 $\hat{\mathbf{Q}}$ を近似するのに対し，分布強化学習では行動価値の分布を近似することでエージェントの性能を改善する．分布強化学習の概念を最初に導入した Categorical DQN [8]では，行動価値の連続分布をカテゴリカル分布で近似している．ただし，Categorical DQN では行動価値の上下限値が既知でないとカテゴリカル分布のビン幅が決定できない点や，証明された理論と実装で損失関数の設定に乖離がある点などの問題があった．そこで，QR-DQN では行動価値の累積分布関数の逆関数（$F^{-1}:\mathbb{R}\to\mathbb{R}\in[0,1]$）をカテゴリカル分布で近似することで，Categorical DQN の問題点を解消している．ここでは，ビン数を 50 に設定する．したがって，QR-DQN を用いるとき，$\boldsymbol{\theta}_7$ と $\boldsymbol{\theta}_8$ の列数はそれぞれ 50 倍となる．累積分布関数から行動価値 $\hat{\mathbf{Q}}$ を推定するには，各分位点において近似した逆関数の値の平均をとればよい．

3.3. 行動価値の推定誤差

ある状態 s で行動 a をとり，次状態 s' と報酬 r を観測したときに訓練パラメータ $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1,\cdots,\boldsymbol{\theta}_8\}$ を変数として算出する行動価値 $Q(s,a|\boldsymbol{\Theta})$ が最小化すべき誤差 δ を次式で定義する．

$$\delta = \sum_{t=1}^{m}\gamma^{t-1}r_t + \gamma^m Q\left(s',\underset{a'}{\mathrm{argmax}}\,Q\left(s',a'|\boldsymbol{\Theta}\right)|\tilde{\boldsymbol{\Theta}}\right) - Q\left(s,a|\boldsymbol{\Theta}\right) \tag{4}$$

$\tilde{\boldsymbol{\Theta}}$ は学習過程で得られる過去の訓練パラメータの値であり，100 ステップごとに現在の訓練パラメータと同期する．式(4)では，元の DQN の行動価値推定から以下の改良を行っている．

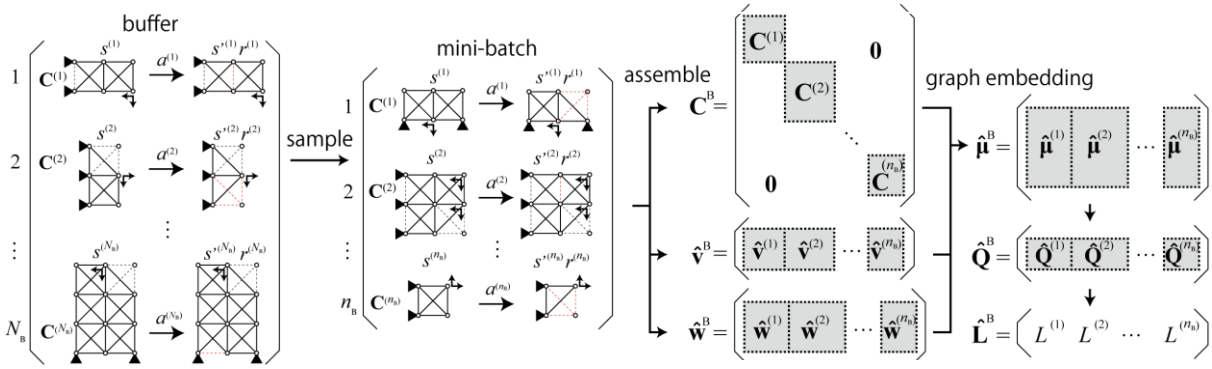

図 3 ブロック行列を用いたミニバッチ学習の演算過程

(i) Double Q-learning

次状態でとる行動を現在のパラメータ $\boldsymbol{\Theta}$ を用いて決定し，その行動価値の評価を $\tilde{\boldsymbol{\Theta}}$ を用いて行う[5]．行動価値が過大に推定されることを防ぐ効果がある．

(ii) Multi-step Q-learning

現状態から m ステップ先までの報酬を記憶し，それらの報酬と m ステップ先の状態での行動価値の推定値を用いて学習を行う[6]．ある行動を選択してから報酬が得られるまでに遅延がある場合に学習効率を改善できる．

本手法では行動価値の分布を近似するため，損失関数 L には行動価値の累積分布関数の逆関数の分位点ごとに重みづけを行う Quantile Huber loss (ただし Huber loss のパラメータは 1.0) L を用いる．

3.4. ミニバッチ学習と損失関数の補正

ミニバッチ学習では，複数のサンプルを同時に用いて訓練パラメータを更新する．バッファ内に一時保存している直近 N_{B} 個のサンプルから取得した 1 ミニバッチ内のサンプル数を n_{B} とする．このときのサンプリングには，エージェントにとって意外性の高いサンプルを優先的に取得する Prioritized experience replay [7]を導入する．具体的には，各サンプル i に対して計算した直近の Huber loss L_i を用いてサンプリング確率 p_i に以下の重みをつける．

$$p_i = \upsilon_i \Big/ \sum_{k=1}^{N_{\mathrm{b}}} \upsilon_k \tag{5a}$$

$$\upsilon_i = \left(L_i + 0.01\right)^{0.6} \tag{5b}$$

ただし，まだ一度もサンプリングされていないサンプルの υ_i は 1.0 に設定する．

図 3 にブロック行列を用いた複数のトラスの接続関係の表現方法を示す．$\mathbf{C}^{(i)}\ (i=1,\cdots,n_{\mathrm{B}})$ は各サンプルのトラスの接続行列である．これらの接続行列を対角方向に配置することで，ミニバッチ内の接続行列が異なる複数のトラスを連成させずに 1 つの行列 $\mathbf{C}^{\mathrm{B}}$ で表現できる．また，図 1 右にミニバッチ化した節点と部材の入力 $\hat{\mathbf{v}}^{\mathrm{B}}$ と $\hat{\mathbf{w}}^{\mathrm{B}}$ を示す．サンプルによらず $\hat{\mathbf{v}}$ と $\hat{\mathbf{w}}$ の行数はそれぞれ一定なので，行方向に連結している．これらの入力を用いて，式(2)によるグラフ埋め込みを行うことで，ミニバッチ内の全部材の特徴量 $\hat{\boldsymbol{\mu}}^{\mathrm{B}}$ を異なるトラスの入力が互いに連成することなくまとめて計算できる．

同様にしてミニバッチ内全ての部材に対応する行動価値 $\hat{\mathbf{Q}}^{\mathrm{B}}$ も式(3)に基づいて計算できるが，$\hat{\boldsymbol{\mu}}_{\Sigma}$ の計算においてミニバッチ内全ての部材ではなく各サンプルの部材ごとに和をとることに注意する．

以上の手順で行動価値の計算をミニバッチ化できたので，ミニバッチ内の n_{b} 個のサンプルの Quantile Huber loss $\hat{\mathbf{L}}^{\mathrm{B}}(\boldsymbol{\Theta})$ もまとめて計算できる．

Prioritized experience replay を利用するにあたり，同じサンプルを繰り返し学習することによる不安定性を回避するため，各サンプルの Quantile Huber loss にサンプリング確率に応じて以下のように補正した損失関数 $\tilde{L}$ を用いる．

$$\tilde{L}_i = \left(\frac{1}{N_{\mathrm{B}}} \cdot \frac{1}{p_i}\right)^{\beta} L_i \tag{6}$$

β は補正の強さを決めるハイパーパラメータであり，学習開始時には $\beta = 0.4$，学習終了時には $\beta = 1.0$ となるよう，訓練エピソード数に応じて線形に増加させる．このようにして求めた n_{B} 個の損失関数の平均値を，ミニバッチ学習において最小化すべき損失関数と定義する．損失関数の勾配に基づく訓練パラメータの最適化アルゴリズムには RMSprop [9]を用いる．

4. 数値例題

4.1. 解析条件

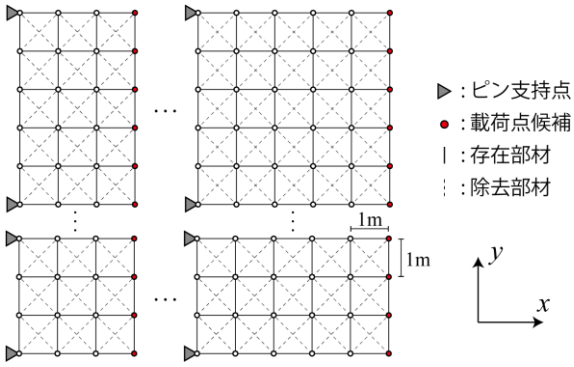

図 4　訓練に用いるトラス

図 4 に示すような，各軸方向に 3，4 あるいは 5 グリッドを有する GS を用いて訓練を行う．左上・左下の端点をピン支持とし，右端のエッジ上の節点から 1 つ以上載荷点をランダムに選ぶ．載荷点には x 軸正（または負）の向き，y 軸正（，または負）の向きにそれぞれ一様に 1 kN の荷重を作用させて 2 つの荷重条件を定義する．

$\bar{\sigma}=200$ [N/mm²]，$\bar{A}=25$ [mm²]，$n_{\mathrm{f}}=100$，$N_{\mathrm{B}}=10^5$，$n_{\mathrm{B}}=32$ とする．学習時の ε は 0.1 に設定し，$\bar{u}$ は全ての部材が存在する場合の最大節点変位の 100 倍の値とする．

4.2. 結果

5000 エピソードの訓練を行い，10 エピソードごとに $\varepsilon=0$ の greedy 方策を用いて図 6 の step 0 の状態から 1 エピソードのシミュレーションを行い得られた累積報酬の履歴を図 5 に示す．訓練エピソードの増加に伴いエージェントがより多くの累積報酬を獲得する傾向が確認できる．最も多くの累積報酬を獲得した 3970 エピソード学習時点での訓練パラメータ $\boldsymbol{\Theta}$ を用いて，二種類のトラスの位相を最適化した結果を図 6，7 に示す．ただし，x 軸正の方向の荷重と y 軸負の方向の荷重は学習時と同様に別々の荷重条件として作用させ，荷重の大きさも学習時と同じ 1kN とする．斜材の配置が非効率ではあるが，疎な初期位相から部材付加・部材除去を行うことに成功した．訓練パラメータのサイズが節点・部材数に依存しないため，学習済みのエージェントがトラスの規模に依らず適用可能なことも本手法の大きな利点である．

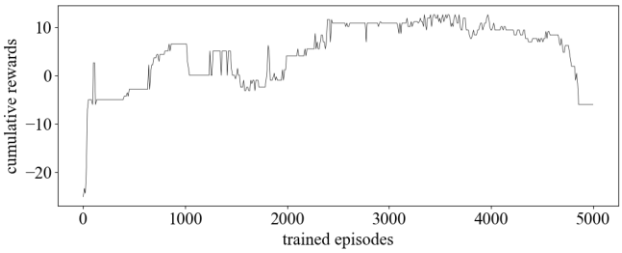

図 5　学習過程での累積報酬の履歴

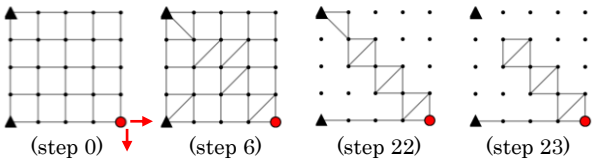

図 6　4×4 トラスの部材付加・除去過程

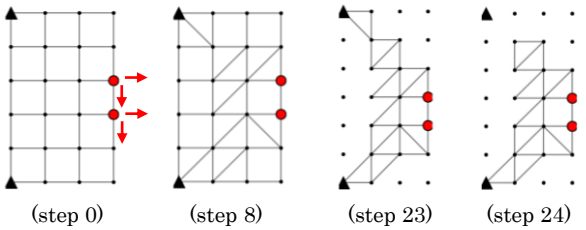

図 7　3×5 トラスの部材付加・除去過程

5. 結

複数荷重条件に対する部材応力と節点変位の制約下で部材総体積を最小化する平面トラスの最適化について，疎な初期位相から部材付加・除去を行うことができるグラフ埋め込みと強化学習の複合手法を提案した．

それぞれの部材に対して部材付加・除去の二種類の行動が適用できるよう行動価値の近似式を改善した．また，強化学習の効率化を図るため，損失関数の計算やミニバッチ学習のサンプリング過程において DQN の改善手法を適用した．

部材付加過程では不安定な位相に対してどの位置に部材を配置するかを考えるため，全ての部材が存在すると仮定した場合の応力比をグラフ埋め込みの入力に追加することで安定トラスにおける力の流れを考慮できるように工夫した．しかし，数値例題においてエージェントの部材付加性能は十分に優れているとは言えず，入力やグラフ埋め込みの演算に改良の余地がある．

謝辞

本研究は，JSPS 科研費 JP20H04467，京都大学若手研究者スタートアップ研究費，JSPS 研究活動スタート支援 JP21K20461 の助成を受けた．ここに記して謝意を表する．

[参考文献]

1) Mnih, V., et al., Human-level control through deep reinforcement learning. Nature 518, 529–533, 2015.
2) Hayashi K. and Ohsaki M., Reinforcement learning and graph embedding for binary truss topology optimization under stress and displacement constraints. Front. Built Environ., 2020; doi:10.3389/fbuil.2020.00059.
3) Wang Z. et al., Dueling network architectures for deep reinforcement learning. In Proc. of 33rd Int. Conf. on Machine Learning - Volume 48, ICML'16, p. 1995–2003, New York, NY, USA, 2016.
4) Dabney W. et al., Distributional reinforcement learning with quantile regression. In Proc. of 32nd AAAI Conf. on Artificial Intelligence (AAAI-18), p. 2892–2901, New Orleans, LA, USA, 2018.
5) Hasselt H. et al., Deep reinforcement learning with double q-learning. CoRR, 1509.06461, 2015.
6) Sutton, R. S. and Barto, A. G., Introduction to Reinforcement Learning, MIT Press, Cambridge, MA, USA, 1st edition, 1998.
7) Schaul T. et al., Prioritized experience replay. In Proc. of 4th Int. Conf. on Learning Representations, ICLR 2016, San Juan, Puerto Rico, 2016.
8) Bellemare M. G. et al., A distributional perspective on reinforcement learning. In Proc. of 34th Int. Conf. on Machine Learning - Volume 70, ICML'17, p. 449–458, Sydney, NSW, Australia, 2017.
9) Tieleman, T. and Hinton, G., Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning，2012.