

部 門	競 技 部 門	No.1 登録番号	30039
-----	---------	-----------	-------

No.2	1) 予定開発期間：7カ月 2) 予定開発人数：3人																																								
	<table border="1"> <thead> <tr> <th></th> <th>4月</th> <th>5月</th> <th>6月</th> <th>7月</th> <th>8月</th> <th>9月</th> <th>10月</th> </tr> </thead> <tbody> <tr> <td>問題分析</td> <td colspan="2" style="text-align: center;">←————→</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>設計</td> <td></td> <td colspan="5" style="text-align: center;">←————→</td> <td></td> </tr> <tr> <td>実装</td> <td></td> <td colspan="6" style="text-align: center;">←————→</td> </tr> <tr> <td>試用・トレーニング</td> <td></td> <td></td> <td></td> <td></td> <td colspan="3" style="text-align: center;">←————→</td> </tr> </tbody> </table>		4月	5月	6月	7月	8月	9月	10月	問題分析	←————→							設計		←————→						実装		←————→						試用・トレーニング					←————→		
		4月	5月	6月	7月	8月	9月	10月																																	
	問題分析	←————→																																							
	設計		←————→																																						
実装		←————→																																							
試用・トレーニング					←————→																																				

No.3	<p>実現方法</p> <p>1) 陣地の取得アルゴリズム</p> <p>マルチエージェントの深層強化学習を用いて、それぞれの職人の最適な方策を求める。 具体的には AlphaZero で使われているモンテカルロ木探索を使用する。 過去のモンテカルロ木探索の履歴をニューラルネットで近似し、方策の事前分布としてこれを PUCT-MCTS へ導入する。 また、過去のある盤面の最終的な勝敗結果をニューラルネットで学習し、盤面評価関数とする。 そして、セルフプレイを繰り返すことで最適なニューラル方策関数を求める。 最適なニューラル方策関数を使用することで職人が陣地を効率的に取得することができる。</p>
	<p>2) 職人の行動決定方法</p> <p>職人の行動は強化学習による結果と、焼きなまし法や <math>\alpha\beta</math> 法といった複数のアルゴリズムでの探索を行い、その探索による評価結果から最も良い行動であると考えられるものを採用する。 強化学習による探索は上記の陣地の取得アルゴリズムを用いて行動を決定する。 焼きなまし法による探索は、まずビームサーチを使って一定ターン数の自分のチームの職人の行動をそれぞれいくつか列挙し、次にその経路の候補からどれを選ぶと最適な行動となるのかを計算していく。 具体的には、ビームサーチによる探索では、ターン数に応じて職人の動きや、陣地がどのくらいで囲めそうなのかといった期待値を用いながら少しずつ陣地を囲んでいけるような行動を選ぶ。焼きなまし法では、職人とその職人の移動経路の候補をランダムに決定し、その状態の職人の行動から城ポイントや領域ポイント、城壁ポイントを含んだ評価関数を用いて確率的に遷移を行いながら探索する。 そして、<math>\alpha\beta</math> 法による探索は、長期的に探索を行うと時間がかかってしまうという点があるため、最終手まで探索することができる状態になったときだけ行う。最終手まで最もスコアが高くなるような行動を探索し、最適に職人の行動を決定する。</p>
	<p>3) その他（独創的などころ）</p> <p>OpenSiv3D を用いてビジュアライザを作成する。プログラムの挙動を視覚的に表現することによって、どのように職人を行動させ陣地を取らせるかを可視化させる。それに合わせて、プログラムだけでなくユーザの操作も受け付けることによって人対プログラム、人対人の対戦を実現させる。これによって実装したアルゴリズムに対して素早くフィードバックを行うことが可能になり、詳細なパラメーターの調整やそのアルゴリズムの長所短所を発見することにより更なる強化を狙う。 また、これらの試合中ではターンごとにフィールドの情報を記憶する。ゲーム中のフィールド状況を再現可能にすることによって、強化学習の学習データに使えるようにする。 本番環境では通信もビジュアライザによって行う。つまり、データの受け取り、プログラムの実行、データの送信までを一繋がりですmoothに実行させる。これによってヒューマンエラーを防ぐだけでなく、最短3秒という短い制限時間でもできるだけ多くの時間を職人の行動の探索に割くことができる。</p>

No.4	<p>開発環境</p> <p>ソルバー: Visual Studio Code GUI: Visual Studio 2022, OpenSiv3D (v0.6.9)</p>
------	---