ローグライクゲームの研究用プラットフォーム

高橋一幸[†1]　Temsiririrkkul Sila[†2]　池田心[†3]

概要：ローグライクゲームは，マップや敵配置がランダムで臨機応変にアイテム等を使い分ける必要がある一人用ダンジョン探索ゲームの総称である．日本では不思議のダンジョンシリーズなどが有名である．これらは，短期的戦術と長期的スケジューリングの両方を要すること，ゲームの設定を徐々に学んで行くことなど，高度なゲーム AI を研究するうえでの良い課題をいくつも持つ．本論文では，ローグライクゲームの学術用プラットフォームを紹介し，そこに適用可能な手法群や，これを用いてどのような研究・競技会が行えるかを考察する．

キーワード：ローグライク，不思議のダンジョン，プラットフォーム

Research Platform of Rogue-Like Games

Kazuyuki TAKAHASHI[†1]　Sila TEMSIRIRIRKKUL[†2]　Kokolo IKEDA[†3]

Abstract: Roguelike Game is a genre of one-player role-playing video games, where the map is randomly generated and the monsters are randomly located every play, then the player needs to select his strategy properly according to the given situation. In Japan, "Mystery Dungeon" is very popular as a roguelike game. Roguelike games have several typical problems for sophisticated game AIs, for example both of short-time strategy and long-time scheduling are necessary and the balancing is important, or the player needs to learn the settings by trials and errors. In this paper, we introduced a roguelike game platform, and discuss the possible methods for this game, possible researches to be interesting, and possible competitions to be held.

Keywords: Roguelike Games, Mystery Dungeon, Game Platform

1. はじめに

近年のコンピュータゲームプレイヤ（ゲーム AI）の研究では，ボードゲームのみならずビデオゲームも対象とし，様々なジャンルのゲームにおいて高性能な AI の研究が進められている．不完全情報性・多人数性・リアルタイム性など，従来取り組まれてきた囲碁や将棋とは違った困難さを持つゲームも多く，それぞれ新しい解決手段が必要なため，これからも研究の重要性は高い．

市販のゲームそのものを用いようとすると，環境の構築が面倒であったりルールが煩雑すぎたりするため，AI 部分のみを開発できるようにした統一のプラットフォームが用いられることも多い．我々は，人気の高いゲームジャンルの一つである“ローグライク(Rogue like)ゲーム”を取り上げ，この学術プラットフォームを 2017 年の Game Programming Workshop (GPW) において提案した[8]. ローグライクゲームには「長期的なスケジューリング」「短期的な戦術」「探索と収穫のジレンマ」「疎密のある行動選択」など AI が取り組むべき複数の課題があり，また Starcraft などより複雑な問題に比べれば比較的取り組みやすい対象であると考えたためである．

本論文はこのプラットフォームをより多くの人に使ってもらうための拡張や，それによってどのような競技会ができるかなどを議論するものである．単独で読んだ場合の利便性を考慮し，本論文の文章や図の多く，特に第２章第３章は前論文[8]と重複した内容となっている．

2. 関連研究とプラットフォームの必要性

ローグライクゲームの詳細な定義は次章で述べるが，概ね，1 人の冒険者がダンジョンを探検し，アイテムを集めそれを駆使しながら敵を倒し，力を強め，複数階層を突破してゴールを目指す形態を取る（図 1）.

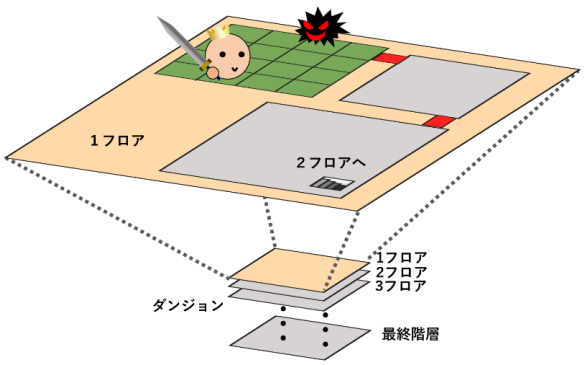

図 1：ダンジョンの構造，概略

†1 北陸先端科学技術大学院大学 s1610110@jaist.ac.jp
Japan Advanced Institute of Science and Technology
†2 temsiririrkkul@jaist.ac.jp
†3 kokolo@jaist.ac.jp

ⓒ2018 Information Processing Society of Japan

マップや敵配置・アイテム配置がランダムであることが特徴の一つであり，プレイヤは毎回異なる状況に合わせて作戦を立てる必要がある．

ローグライクゲームを直接扱った研究としては，田中らによる提案[1]や，Vojtech らによるものがある[2]．田中らの提案は前論文[8]執筆時には追試・実装が困難な状況であった．現在はソースコードを提供頂いたが，今のところ競技会が開かれたり他論文で採用されたりはしていないようである．また Vojtech らは Desktop Dungeon というゲームを対象としているが，これには複数階層性がなく，長期的なスケジューリングが必要ないという不満がある．これらから，新規プラットフォームを作成公開することには一定の価値があったと考える．

またそもそもローグライクゲームに絞って AI を研究することに価値があるのかという議論もあり得るだろう．近年は IEEE-CIG などの国際会議で General Game Playing (GGP) および General Video Game Playing (GVGP) といった汎用ゲーム AI の競技会が行われ，様々なモンテカルロ探索の拡張[3]や Deep Q-Network[4]などによりその性能も向上しているためである．しかし，いずれはそういった問題群の一つになるにせよ，課題を明確にしてそれを解決していくことには価値があると考える．

なお，ローグライクゲームの枠組みに絞ったとしてもその設定（システム，敵やアイテムなど）は様々であり，プレイしながらゲームごと・ダンジョンごとの攻略法を考えるのは人間プレイヤにとっても面白いことである．これをコンピュータに行わせることは，言わば GGP の 1 グループ，General Rogue-Like Game Playing とでも言えるものかもしれない．

もう一点，ローグライクゲームはプレイヤ対ルールベースの単純な敵キャラ複数という形を取り一人ゲームと称されることが多いが，実は「AI が制御する味方キャラ」がいるゲームも多く，その動きは時としてプレイヤに不満を感じさせるものになってしまっている．注意したい点は，これが必ずしも“仲間 AI の行動が勝利のために最適でないから”ではなく，“人間プレイヤの好みや意図に合った行動をしてくれないから”であることである．これは我々の価値観学習チームプレイ AI の研究[5]と同様の背景であり，この点を解消するためにも，AI プレイヤの性能向上と，将来的には味方キャラがいるプラットフォームの提案が必要だと考えている．

前論文[8]は，これらの背景および次章で再掲する必須要素の考察に基づいて一つのルールを提案するものであった．さらにそこでは簡単なモンテカルロ法など手法面での実装も行っていた．今回は，UEC-GAT 競技会でのポスターということもあり，今後どのようにしていけばこのプラットフォームを利用してもらえるか，どのような競技会が可能かという点について議論していきたい．

3. 必要な要素と中心的ルール

ローグライクゲームは Rogue のほか Angband, Nethack, 不思議のダンジョンシリーズなど数多くの有名な作品がありその数はフリーゲームを除いても数十にのぼる．当然ルールも様々であり，そもそも何をもってローグライクゲームと呼ぶのかすら明確ではない．2008 年の International Roguelike Development Conference では，Roguelike とは何かを定義する試みが行われ，Berlin Interpretation としてまとめられた[6]．その中では，全てが必須ではないと断った上で，以下のような要素が high value factors として提唱されている．

- Random environment generation. マップ，アイテムの配置や出現，モンスターの配置（種類は固定）が毎回ランダムに決定されること．
- Permadeath. ゲームキャラクタは，アイテムや経験を溜めても，死ねばそれらを失って最初からプレイすること．
- Turn-based. リアルタイムの判断や操作を要求せず，熟考できること．通常，自分のキャラクタが一つ移動または行動をすれば，その後にモンスターも一つ移動または行動する．ただし，倍速や 2 ターンに 1 回しか行動しない敵もいる．
- Grid-based. マップは正方形のマス目で構成され，敵味方のキャラクタは一つのマス目をふさぐこと．つまり，ピクセル単位の細かい位置を持たないこと．
- Non-modal. 移動と戦闘のシーンは別個でなく，全ての行動はゲームのどの時点でも利用できること．ただしいくつかのゲームでの“店”はこれにあてはまらない．
- Complexity. ゲームの目標は一つでも，それを達成するための方法や道筋は複数あること．敵とアイテムの相性，アイテム同士のシナジーなどがこれを生みだすこと．環境がランダムに生成されるが，これが単なる運不運にとどまらず，異なる戦略を要求する面白さを生み出す．
- Resource management. 食料や傷薬といったアイテムは有限で，利用が制限されること．従って長期的に計画を立てながら使わないといけない．
- Hack'n'slash. モンスターを倒す必要とメリット

ⓒ2018 Information Processing Society of Japan

があること．例えば全ての敵から逃げることはできないが，敵を倒せばアイテムが落ちることもあり，経験値を稼いでレベルを稼げること．

- Exploration and discovery. プレイヤはマップやアイテムの位置を知らされず，毎回慎重に探索してアイテムを探し，時には未鑑定アイテムの使用法を考える必要があること．

これらを他の有名なゲームと比較してみれば，例えば Starcraft とは Turn-based や Grid-based の点で異なる．つまり，Starcraft ではリアルタイムに行動選択をする必要があり，敵味方や資源の位置関係も細かく制御される．あるいは例えばドラゴンクエストなどの JRPG とは Permadeath，Non-modal などの点で異なる．つまり，JRPG では死んでも多くの場合レベルや所持品は失われず，フィールド・町・戦闘などは個別のシーンとなりとれる行動も異なる．

我々は，この high value factors をベースに， 多くのローグライクゲームに共通しているかどうか，また良い AI を作ることが「面倒」ではなく「挑戦的」になるかどうか，という基準に基づき，第一段のプラットフォームのルールを策定し，研究環境を実装した[8]．詳細は以下の通りであるが，これらはあくまで我々が提案したい枠組みの中の“一つのルールセット”に過ぎず，実際若干バランスの悪い部分もあることがその後の実験で分かっており，今後変更を検討する．

- マップは四角マスで，50×30.
- マップは 3〜4 部屋で構成され，一つの階段があり，階段を進むと新しい階層となる．4 階層から構成され，最後の階層で階段に辿りつけば勝ち．
- 部屋のサイズや通路は図 2 の 5 パターン．

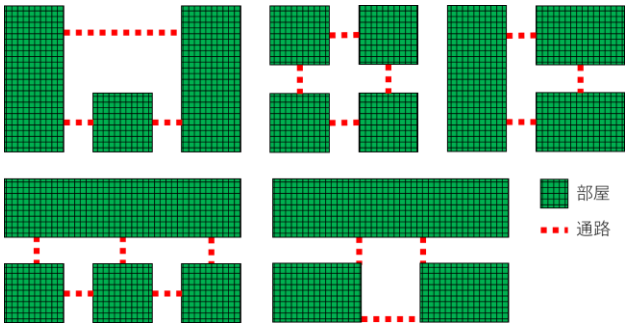

図 2：部屋と通路の構成

- プレイヤは，9×9 視野を持つ．さらに，部屋の内部にいる場合はその内部の全ての敵やアイテムの位置が分かる．
- ターン制で，自分の移動または攻撃後，敵の移動または攻撃が処理される．
- 移動や攻撃は上下左右斜めの 8 方向 1 マスである．ただし，上または右に壁がある場合，右上には移動や攻撃ができない．他の場合も同じ．（図 3）

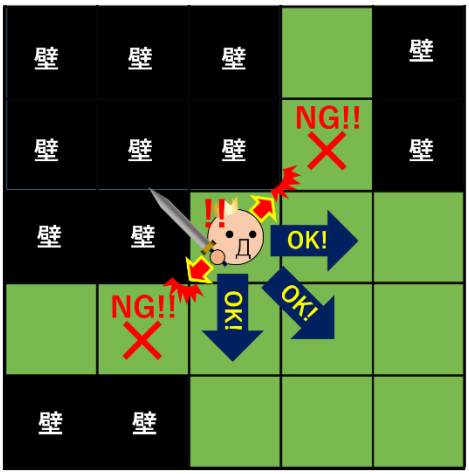

図 3：移動・攻撃できないマップの例

- プレイヤキャラクタにはレベル・経験値・体力・満腹度・攻撃力のパラメータがある．
  - レベルは 1 を初期値とし，経験値は 0 を初期値とする．経験値は敵を倒すと入手でき，$10\times1.1^{(\text{レベル}-1)}$ たまるとレベルが 1 上昇し，経験値はその分減る．
  - 攻撃力は 35 を初期値とし，レベルごとに 1 上昇する．
  - 体力は 100 を初期値とし，レベルごとに上限が 1 上昇する．また，毎ターン，上限の 1/200 だけ回復する．
  - 満腹度は 100 を初期値および上限とし，10 ターンに 1 減り，0 になると体力が毎ターン 1 減る．
- 飢餓または敵の攻撃で体力が 0 になると負けである．
- 通常攻撃は周囲 8 マスのいずれかに対して行え，攻撃力分のダメージを与える．
- アイテムは食料，回復薬，矢，杖がある．
  - アイテムは各階層に合計 3〜5 つ，数も種類も位置もランダムに配置される．通路には配置されない．アイテム数は 50%の確率で 4 つ配置され，3 つまたは 5 つの場合が 25%ずつである．
  - 食料は満腹度を 20 回復する．
  - 回復薬は体力を 50 回復する．
  - 杖と矢は消耗品で，上下左右斜めの 8 方向，プレイヤから 5 マス以内に見えている敵に遠隔で利用することができる．
  - 杖は敵を別の部屋のどこかに飛ばす（一時

ⓒ2018 Information Processing Society of Japan

しのぎ）ことができる．
  - 矢は敵に 15 ダメージ与えることができる．矢は 3 本まとめて拾える．
  - アイテムは 10 個まで所持できる．
  - アイテムを置いたり投げたりはできない．
- 敵キャラクタは各階層 1 種類である．
  - 敵の体力は階層ごとに 60,80,90,95，攻撃力は 20,25,30,35，得られる経験値は 15,20,25,30 と少しずつ増える．
  - 敵キャラクタは階段を下りた時点で 4 匹ランダムな位置に配置され，さらに HP が 0 になってから 64 ターン後にランダムな位置に再配置される．
  - 敵はプレイヤキャラクタを見つけると最短距離で接近して攻撃し，そうでなければランダムに移動する．

このルールセットに限らず，ローグライクゲームのプレイヤは短期的には目前の敵を倒したり，強い敵や複数の敵にはアイテムを消費したりして凌ぐ・階段から次階層に逃げるなどの選択を行わなければならない．一方で長期的には将来を見据えアイテムの収集と節約，食料に余裕があれば経験値稼ぎ（敵の復活を待って倒してレベルを上げること）などを考慮する必要がある．「突発死を恐れるかジリ貧を恐れるか」「満腹度と経験値とアイテムのバランス」などはローグライクゲームの中心的な課題であり，最適な行動を簡単に判断することはできない．

本ルールは第一段階のテーマとして多くの要素を排除したが，それでも十分に困難で面白い課題になっていると考える．

発展的な要素としては例えば，

- お金パラメータと，店要素，泥棒要素．
- モンスターハウス．敵が大量出現する部屋．
- 罠．踏むとペナルティのある隠れタイル．
- 特殊状態．暗闇，睡眠，混乱など．
- MP などの追加のステータス．
- 自分と敵以外のキャラクタ．味方など．
- 様々な個性のある敵やアイテム．
- 装備と防御力の概念
- 使うまで効果の分からない未鑑定アイテム．

などが挙げられるが，これは敢えて含めていなかった．これらは多くのゲームに登場し，面白い要素ではあるが，AI 作成をむやみに煩雑にする可能性が高いと判断し，今後その導入を検討する予定である．これらを導入したときに，プラットフォームや各 AI プログラムがどの程度の変更を必要とするかは設計上重要な論点である．我々は，少ない変更で多くの追加要素に対応できるような設計を心掛けた．

4. 作成したプラットフォーム

4.1 設計

前論文[8]では，Java をプログラミング言語とし，10 以上のクラスからなる設計を行ったが，GPW での発表後に「他の言語を使えるように，標準入出力に対応してほしい」「Java でなく，c にしてほしい」「ルールを簡単に変更できるようにしてほしい」などの要望を受けた．そこで，設計を大幅に考え直すことにした．

まず，ローグライクゲームには 3 章でも述べたように様々なルールが存在する．ルール変更を容易にするためには，ルールに関するファイルを一つにまとめ，それを差し替えることでルール更新ができるようにすべきである．次いで，現在非常に多くの言語が使われているため，特定の言語のみをコンパイルできるようにするのではなく，環境として高速な c を基盤としつつも，AI は標準入出力を用いて他の言語で記述できるようにした．

基盤プラットフォームは以下のクラスにより構成される．クラス図概要を図 4 に示す．

- State:ゲームの状態（プレイヤ情報，マップ情報など）を持つ. Player に渡す際には一部の情報（未探索部分のマップ情報など）を隠蔽し，文字列での受け渡しを行う．
- GameManager：ゲームのメインループを管理し，Rule・Player 間の State の受け渡しを行う．
- Rule：State の初期化，ゲームの状態遷移を行う．多くのルールはここで記述する．
- Player：State を与えられ，次の行動を決定する．人間プレイヤ・AI プレイヤのどちらかが該当する．人間プレイヤは GUI を通じて操作を行い，AI プレイヤは c で記述するか，標準入出力を介して動作を行う別プログラムとして記述する．

ⓒ2018 Information Processing Society of Japan

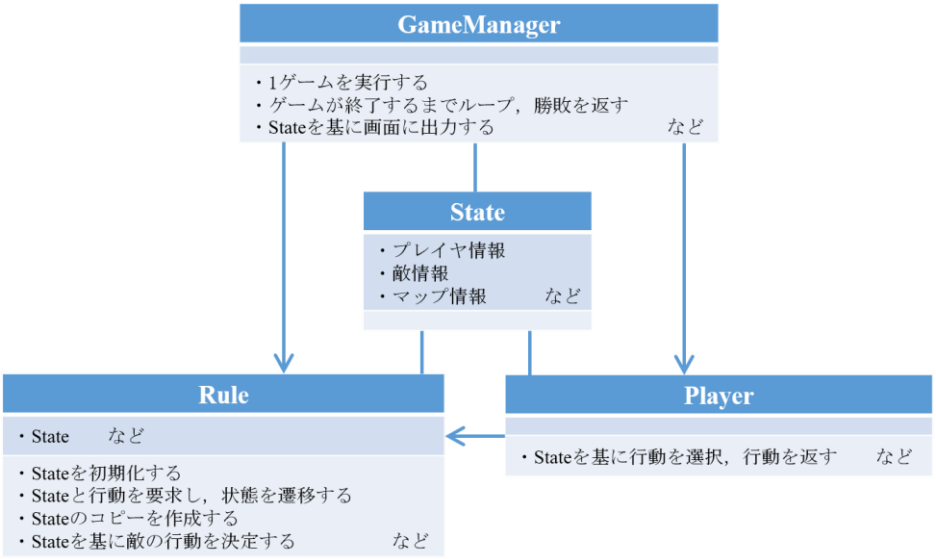

図 4：クラス図概要

実行時のプログラムの流れは以下の通りである．

1) GameManager によりゲームを開始し，Rule により State を初期化．
2) State のコピーを Player に渡し，行動を決定．
3) Rule で State を遷移（Player が選択した行動を用いる）．
4) 目的の階層にたどり着くかプレイヤの HP が 0 になるまで 2)・3)を繰り返す．

ルール変更は，State の初期化や状態遷移を行っている Rule.c を書き換える（置き換える）ことで実現することができるが，容易に変更できるものとできないものが存在する．例えば，

・クリアとなる階層
・敵の数
・敵のパラメータ
・アイテムの追加

などは比較的容易に変更することができる．一方で，

・マス目状でないマップ
・リアルタイム制

など，Berlin Interpretation に沿わないルールへの変更は難しく，考慮しない．

4.2 ゲーム画面と操作方法　（未修整）

図 5 が作成したプラットフォームのゲーム画面である．画面上部（①）にはプレイヤの行動やダメージ，画面中部（②）にはマップ，画面下部（③）にはプレイヤの情報を表示する．マップ上の各文字に対応する役割を表 1 に示す．また，操作方法を表 2 に示す．

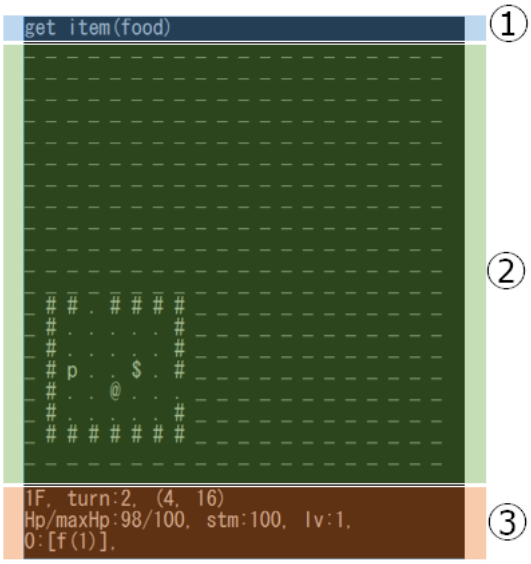

図 5：ゲーム画面（実際は白黒）

表 1：マップ上の記号とその役割

文字	役割
@	プレイヤ
.	床
#	壁
_	未探索
%	階段
$	敵
f	食料
p	回復薬
a	矢
s	杖

©2018 Information Processing Society of Japan

表 2：操作方法

キー操作	行動
テンキー 方向キー	移動，攻撃，待機
f	食料使用
p	回復薬使用
a	矢使用，方向入力
s	杖使用，方向入力
Escape	ゲーム終了

テンキー・方向キーには移動・攻撃・待機の 3 つの行動が割り振られている．これは，プレイヤと敵が隣接している状態で敵の方向に入力すると攻撃，隣接マスに敵がいない状態で移動，テンキーの 5 を入力することでその場で待機（足踏み）するということである．

5. AI のアプローチ

ゲーム AI に用いられる手法は教師あり学習，強化学習，木探索，Direct Policy Search など様々であり，それぞれ一長一短があるうえ，問題に対する向き不向きもある．

If-then ルールからなる決定木は，特定の複雑すぎない問題について，“その問題を良く知っている人が作成した場合”，十分有力な手法になりうる．我々もルールベースプレイヤを提案している[8]．２人対戦ゲームではルールベースプレイヤは「単調になり，癖を読まれ対応される」という大きなデメリットがあるが，１人ゲームでは大きなデメリットにならない．無論性能には限界があり，あくまでベースラインとして使われることが多い手法であろう．

教師あり学習は，ボナンザや AlphaGo の初期版で用いられるなど，“強いプレイヤの行動が入手できる場合”非常に有力な手法であり，またαβ探索など他の手法と組み合わせることもできる．ルールが固定されにくいローグライクゲームでは，教師あり学習だけで強いプログラムを作るに十分なほどの履歴は得にくいと考える．

強化学習は，Atari ゲームや囲碁に用いられ近年その評価を高めている手法である．強化学習では，“ルールが固定され，かつそのルールで何度もプレイすることができる”ことが必要である．ローグライクゲームでも事前にルールが与えられれば，非常に強力な手法になると思われる．もし GGP などのように競技会時に新しいルールが与えられるような場合は事前学習が困難であろうと考える．

モンテカルロ法は，環境に対する事前学習が不十分であってもそれなりの行動をとることができることができる．また，そのためには多くのシミュレーションが必要で 1 手の決定に時間がかかることが多いが，それはリアルタイムゲームよりはローグライクなどのターン性ゲームに向いているということでもある．我々もモンテカルロ法を提案している[8].

Direct Policy Search は，政策を何らかの形（状態評価関数だったり，状態→行動のマップだったり）で表現し，それを問題に適用・評価し，悪い政策を表すパラメータは淘汰し，良い政策を表すパラメータを優遇・最適化するという枠組みである．これも強化学習と同じく事前学習が必要である．Vojtech らの Desktop Dungeon に関する研究ではこのような進化計算アプローチが用いられている[2].

6. 競技会の可能性

本論文では，ローグライクゲームのプラットフォームを提案し，様々なアプローチが可能であることを示した．これらのアプローチを比較するためには，プラットフォームを公開して論文を書いてもらうだけでなく，競技会を開いてより気楽にさまざまな手法を試してもらうことも重要であると考える．

ゲーム AI の最も重要な国際会議の一つである IEEE-CIG では毎年多くのコンペティションが開かれているが，例えば（スーパーマリオ風の）横スクロールアクションゲームについては，「強いプレイヤを作る」という目的の競技会以外にも，「人間らしいプレイヤを作る」「人間プレイヤのために良いマップを作る」などさまざまな競技会が開催されている．ローグライクゲームの AI 競技会も，さまざまな目的のものがあってよいし，またあるべきだろう．以下にその例を示す．どのような競技会であれば参加する気になるか，感想や要望を出していただければ幸いである．

【強さを競う競技会】

当面の間，中心となるのはやはり強さを競う競技会であろう．しかし，それでも非常に多くの設定がありえて，それによって有効となる手法も変わってくると思われる．

＜時間＞　行動決定のために許される時間を，例えば 1 手 0.01 秒などとリアルタイムゲームのように定めてもよいし，1 フロア 5 分以内などと少し長めに取ってもよいであろう．長ければ探索系の手法が有力になりうる．このような時間の設定は，UEC-GAT

ⓒ2018 Information Processing Society of Japan

の他のゲームでもしばしば別部門として行われる．ランダム性を伴うゲームのため，複数回の試行が比較のためには必要であることには注意しなければならない．

＜ルールの複雑さ＞　本原稿に書いたような最低限の単純なルールでもよいし，RogueClone のような複雑なルールでもよい．AI にやらせるには前者のほうがよいと思うが，人間にもやってもらって比較するには後者など知られているもののほうがよいかもしれない．

＜ルールの開示＞　ルールを数か月前に開示しておくこともできるし，GGP などのように当日開示して即興的な強さを競っても良い．前者ならば強化学習や Direct Policy Search，後者ならばモンテカルロ法などが有力になると思われる．

【人間らしさを競う競技会】

これも近年頻繁に行われるようになった競技会である．強さだけでなく，人間がプレイしていると思わせるような工夫を競うものである．人間とＡＩのプレイログを再生し，ブラインドで評価することで実施することができる．この再生の際に，「実際に使われた時間をそのまま再現する」か，「１ターンごと１秒など，固定スピードで再現する」かは，難しい問題である．“難しい局面で悩む”ようなことを人間らしさに含めるのであれば前者が良いだろう．

【良いパートナーを競う競技会】

ローグライクゲームの多くには，プレイヤと共に戦ってくれる仲間やペットのような存在がある．これらの行動は非常に単純なルールで制御されていることも多いと感じられる．ペットであればそれでもよいが，ゲームのストーリー上あるいは見た目上あきらかに知性を持つ存在（例えば人間）であるのに賢くない挙動をされれば，違和感や不満の原因となりうる．従って，パートナーとしての満足度を競うような競技会もありうるだろう．

このとき気を付けなければいけないのは，「最も勝率の高い行動」は必ずしも「最も人間プレイヤの満足度を上げる行動」とはならない点である．人間プレイヤは最適行動を間違えて認識していることもあるし，そもそも勝率以外の要素，例えば“早くクリアしたい”などを目指しているかもしれないからである．このような場合，プレイヤの行動からそのプレイヤの価値観や偏りを推定し，それに迎合する必要がある[9].

【良いバランスのルールを競う競技会】

ローグライクゲームなど一人ゲームのプレイヤを作成することに学術的価値があることは確かだが，商業的価値について考えてみると，“そのようなプレイヤを使って，ゲームバランスをテストする”ことが最も利用可能性が高いと考える．

そこで，ルールセットそのものを競う競技会も行えると考える．あまりに自由にルールを変えられるとそれはゲームデザインの競技会になってしまうので，マップや敵やアイテムなどの基本的枠組みは固定したうえで，その広さ・体力攻撃力・効果量や個数，などだけをパラメータとして変更して良いことにする．これをプレイしてもらい，楽しいと思うもの理不尽でないと思うものを選んでもらうような競技会である．

7. おわりに

我々の研究では，ローグライクゲームというジャンルを新たなゲーム AI の研究対象として取り上げ，何が面白くまた難しいのかをまとめた後，重要な課題を失わない程度には複雑さを保ちつつ，煩雑でない程度に単純化したルールを提案している．本論文では，そのプラットフォームを多くの人に利用してもらうための工夫を示したうえで，どのような AI プレイヤのアプローチがありうるのか，またどのような競技会がありうるのかを議論した．

本論文ならびに GAT 大会でのポスターセッションを通じて，ローグライクゲームのプラットフォームや競技会に関する議論が盛り上がり，さまざま意見が出されることを期待したい．

参考文献

[1] 田中 成俊，橋山 智訓，市野 順子，田野 俊一，ローグライクの AI コンペティション，第 29 回ファジィシステムシンポジウム，2013，pp.435-440.

[2] Vojtech Cerny and Filip Dechterenko, Rogue-Like Games as a Playground for Artificial Intelligence – Evolutionary Approach, International Conference on Entertainment Computing (ICEC), 2015, pp.261-271.

[3] D. J. N. J. Soemers, C. F. Sironi, T. Schuster and M. H. M. Winands, Enhancements for real-time Monte-Carlo Tree Search in General Video Game Playing, IEEE Conference on Computational Intelligence and Games (CIG), 2016, pp.1-8.

[4] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. Nature 518, 2015, pp.529-533.

[5] Naoyuki Sato, Kokolo Ikeda and Takayuki Wada, Estimation of Player’s Preference for Cooperative RPGs Using Multi-Strategy Monte-Carlo Method, IEEE Conference on Computational Intelligence and Games (CIG), 2015, pp.51-59.

ⓒ2018 Information Processing Society of Japan

[6] “Berlin Interpretation - RogueBasin” http://www.roguebasin.com/index.php?title=Berlin_Interpretation,s (参照 2017-07-24)
[7] 藤木 翼，村山公志朗，池田 心，ターン制ストラテジーにおける状態評価関数を用いた深さ限定モンテカルロの適用，エンターテイメントと認知科学研究ステーション（E&C）第８回シンポジウム，2014，pp.1-4.
[8] 高橋 一幸，Temsiririrkkul Sila，池田 心，ローグライクゲームの研究用ルール提案とモンテカルロ法の適用, 第 22 回ゲームプログラミングワークショップ(GPW-17), 2017-11, pp.19-25.
[9] 和田 堯之，池田 心，少数の記録からプレイヤの価値観を機械学習するチームプレイ AI の構成，第 33 回ゲーム情報学研究会(GI)，2015-03，pp.1-8.

ⓒ2018 Information Processing Society of Japan