

深層強化学習の人流誘導への適用に関する基礎的研究

A Fundamental Study on Pedestrian Flow Control using Deep Reinforcement Learning

○恒川 裕史*¹
Hiroshi Tsunekawa*¹

*¹ 株式会社竹中工務店技術研究所 グループ長
Group Leader, Research and Development Institute, Takenaka Corporation

Summary:

A method using deep reinforcement learning that incorporates multi-agent simulation to induce so as to be able to exit the crowd smoothly from theaters and arenas are proposed. Cellular automata was used for pedestrian flow simulation. Two deep reinforcement learning methods, i.e., a method based on Q learning and a method based on actor-critic, were used for flow control. For a 28×29 space, first, the Q-learning based method was applied to the problem that the correct answer is known, and it was confirmed that the exiting situation was the same as that of correct answer case. Next, actor-critic based method was applied to more general situation and it showed that the exiting situation was statistically significantly improved. Furthermore, we analyzed the learning results and showed the possibility to extract useful knowledge for pedestrian flow control. Limiting the range of people that can be grasped reduces the effect of induction but never loses the effect even if only 1/3 of area can be observed.

キーワード: 深層強化学習; 誘導; 人流; マルチエージェントシミュレーション; 知識抽出

Keywords: Reinforcement learning; induction; pedestrian flow; multi-agent simulation; knowledge discovery.

1. はじめに

劇場や球場、アリーナなどでそれぞれのアトラクションを楽しみ、帰宅しようとした時に長蛇の列が微動だにせず、不快な思いをした人は多いであろう。このような場合、円滑に退出できるように警備員などが退出の誘導を行う。建築計画的には、うまく退出できるようなレイアウト上の工夫やサイン計画を予め検討しておくのであるが、それに加えてうまく退出の誘導ができれば、退出に際する待ち時間を短縮できる可能性がある。大西ら¹⁾は花火大会での混雑緩和を目的に計測やシミュレーションを行うとともに、外壁にプロジェクションマッピングを用いたガイドを表示する実験を行ったが、表示は目視による人の判断であった。杉浦ら²⁾はシミュレーションと数値モデルを用いた誘導灯の最適配置と動的な補助誘導灯の制御を行ったが制御はルールベースであった。中島ら³⁾はGPS携帯電話で個別に誘導情報を提供する実験を仮想空間を用いて拡張し大規模群衆実験を行う環境を試行したが、誘導情報としては最寄り避難所を知らせるに留まっている。

第3次人工知能ブームの中、2016年に Google の AlphaGo⁴⁾が囲碁のトッププロである李世石九段に勝利し、世間を驚かせた。AlphaGo は深層学習による教師あり学習と強化学習を用いた手法⁵⁾である。特に後者の強化学習は、途中の一手一手が良手か否かがわからず、結果

は最後にしかわからない囲碁をプレイするために使われたものであるが、群衆の誘導も、その時々指示の善しあしは必ずしも不明で、最終的に短時間で退出できたことでしか誘導の善しあしは判断できないという点で共通点がある。また、AlphaGo の成功は、仮想的な対局を自由に行えたことが大きい。群衆誘導の分野でもマルチエージェントシミュレーション (MAS)⁶⁾を用いて仮想的に誘導を試行することができ、この点も共通である。囲碁はコンピュータがすべての状態を観測できるのに対して群衆誘導では必ずしもすべての状態を観測できるわけではないため同一ではないが、その可能性には期待を持てる。本研究では、まず深層強化学習による誘導が可能か否かを小規模モデルで検証し、そのうえで学習結果の分析による誘導のための知見抽出を試行したので報告する。

2. 群衆誘導の方針

人流予測において、短期間の移動の予測ではカルマンフィルタやパーティクルフィルタが用いられるが、人同士の相互作用を考慮する場合には前述の MAS が用いられる⁷⁾。そこで、本研究では、群衆の誘導を MAS を組み込んだ深層強化学習により試行した。

MAS では各人がどこに居るかをコンピュータが把握している。これに基づいた学習で得られた結果を実際の誘導に反映するためには実空間内の人の移動を把握する

日本建築学会情報システム技術委員会
第42回情報・システム・利用・技術シンポジウム論文集, 166-171, 2019年12月, 東京
Proceedings of the 42nd Symposium on Computer Technology of Information,
Systems and Applications, AIJ, 166-171, Dec., 2019, Tokyo

必要がある。また、MAS を実際の人の流れに近づけるためにも、実空間内の人の流れを把握し、分析する必要がある。人流についてはカメラによる計測が行われてきたが、深層学習の発展により近年技術レベルが急激に向上している。また、画像情報のほかに距離が取得できるカメラも登場しており、こうした流れが加速している中で、空間内の人の動きを正確に把握できるようになることが期待される⁹⁾。本研究では、空間内の人の動きはリアルタイムに把握できるものと仮定し、誘導により早期に退出させることをめざすこととする。

3. 誘導方法

3.1. 強化学習

強化学習⁷⁾は環境中のエージェントが環境から状態を受け取り、それに対して行動を選択し、報酬と次の状態を受け取るという枠組みで、受け取る報酬を最大化することが目的である。人工知能関連の研究では、単純化された問題の中では有効でも、現実の社会に適用することが難しいと言う一種の壁がある。強化学習の枠組みは環境に関する知識なしに、環境との相互作用から直接学習を行うが、教師あり学習のように行動の善し悪しに関する情報が直接得られることはなく、しかも行動の善し悪しが結果となって現れるまでには遅延が伴う。こうした状況は現実にもよくあてはまり、強化学習が現実の問題に効果を発揮することが期待される。

3.2. 深層強化学習

Sutton は強化学習 FAQ の中で強化学習の関数近似にニューラルネットワーク (NN) を適用するのはよくある間違いであると述べている⁸⁾。しかし、Mnih⁹⁾らがゲーム機アタリのゲームを強化学習でプレイするために深層ニューラルネットワーク (DNN) を使い、人間のエキスパートよりも高い得点を出したことで、DNN を用いた深層強化学習 (DQN) が注目されている。DQN は、Sutton の TD 法¹⁰⁾の発展形である価値関数ベースの Q 学習¹¹⁾に DNN を適用したものである。Q 学習では次式で示される割引期待報酬 R_t を最大化することを目的とする。

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$$

r は報酬、 γ は将来得られる報酬の割引率である。 R_t を最大化するため、 Q 値を以下のように更新する。

$$Q(s, a) = (1 - \alpha)Q(s, a) + \alpha \left(r + \gamma \max_{a'} Q(s', a') \right)$$

α は学習率、 s は環境の状態、 a は選択する行動、 s' および a' はそれぞれ次のステップの状態と行動である。あるスケジュールで学習率 α を減少させながら Q 値の更新を行い Q 値が収束すると、各状態の最大の Q 値となる行動を選択することが最適政策となることが証明されている¹⁰⁾。DQN では、この Q 関数を DNN で近似する。DNN で Q

関数を近似する場合、問題となるのは、一連の試行で得られる状態、行動、報酬のデータ列に高い相関があることである。NN では学習データ間に相関があると、学習が進まない。これを解決するため、DQN では経験リプレイという手法を採用している。これは予め多くのゲームを行うことで上記のデータ列を収集しておき、そのデータ列の中からランダムに学習データを選ぶことで学習データ間の相関をなくす方法である。

DQN は価値関数をベースとした方法であるが、強化学習にはそのほかにポリシーベースの方法がある。これは、以下のポリシー関数 π を直接改善する方法である。

$$\pi(a|s; \theta)$$

θ はパラメータで、 π は状態 s の場合に行動 a を選択する確率を表す。ポリシーベースの代表的な手法である REINFORCE¹²⁾では、パラメータ θ を $\nabla_{\theta} E[R_t]$ の普遍推定量である以下の方向に更新する。

$$\nabla_{\theta} \log \pi(a_t | s_t; \theta) R_t$$

さらに、ベースライン b_t を差し引くことで期待値を保ちながら推定量の分散を小さくし、学習を促進する。

$$\nabla_{\theta} \log \pi(a_t | s_t; \theta) (R_t - b_t(s_t))$$

b_t は、報酬の平均が用いられるが、こちらも学習の対象とするのが、actor critic という手法である。

本研究では DQN では有意な結果が得られないケースがあったため、DQN に加えて actor critic の一手法である asynchronous advantage actor-critic (A3C)¹³⁾を用いた。A3C では、マルチスレッドで複数のシミュレーションを同時に行い、得られた状態、行動、報酬のデータ列を学習に使うことで、学習データにおける相関を排除する。また、tmax ステップごとにポリシー関数と価値関数を以下の式にしたがい、tmax に亘って累積したもので更新する。

$$d\theta \leftarrow d\theta + \nabla_{\theta} \log \pi(a_t | s_t; \theta') A(s_t, a_t; \theta, \theta_v)$$

$$d\theta_v \leftarrow d\theta_v + \frac{\partial A^2}{\partial \theta'_v}$$

A は、以下に示す action advantage 関数¹⁴⁾、 θ は価値関数のパラメータ、 θ' 、 θ_v はスレッドのパラメータである。

$$A^{\pi}(s, a) = Q^{\pi}(s, a) - V^{\pi}(s)$$

A3C では、この advantage 関数の推定値として以下の値を用いる。

$$A(s_t, a_t; \theta, \theta_v) = \sum_{i=0}^{k-1} \gamma^i r_{t+i} + \gamma^k V(s_{t+k}; \theta_v) - V(s_t; \theta_v)$$

実際の計算は tmax 分の試行を行ったうえでステップを巻き戻しながら更新を計算するというようにその試行で用いた方策を更新に用いる方策 on 型である。A3C では、ポリシー関数 $\pi(a_t | s_t; \theta)$ と価値関数 $V(s_t; \theta_v)$ との 2 つを出力する、前半部分を共有した NN を用いる。

3.3. 人流シミュレーションと誘導

人の流れのシミュレーションはセルオートマトン¹⁵⁾を用いた。セルオートマトンは Stephen Wolfram が体系化し

た時間・空間・状態量がすべて離散的な数理モデルである。セルオートマトンでは人の体格の差や身体の向きを考慮することが難しく、実際の人流を再現する最適な方法ではないかもしれないが、その簡易さ、わかりやすさから基礎的な検討には最適と考え、採用した。人が移動する近傍は縦と横のセルを近傍とするノイマン近傍とした。また、横入りする人が少ない日本人の特性¹⁴⁾を考慮し目的地からの距離計算にはマンハッタン距離を用いた。

人の移動の戦略としては、大きく、出口に向かって移動する場合(戦略(1)とする)と、人についていく場合(戦略(2)とする)の2種類を設定した。ただし、1人の人が完全にどちらかに分類されるのではなく、出口を知っている(1)の行動を主にする人も確率的に(2)の行動をとるように、またその逆の場合も設定した。また、いずれの場合も探知範囲内(直線で10マス)に出口を見つけた場合には直接出口に向かう。また、出口からの距離に差が無い場合には慣性に従う場合とランダムウォークをする場合を確率的に選択するように設定した。戦略(1)の場合は、低い確率で、向かう出口を変える確率も設定した。

誘導はロープ等物理的に行動を強制するものも考えられるが、本研究では壁に貼られた矢印表示器のようなものを想定し、その指示に従う確率、指示を忘れる確率を設定する。戦略(1)の場合に自分が向かっている出口とは異なる方向に矢印が表示されていた場合には、ある確率で向かう出口を変えることとした。

学習結果からの知見は、どのような状態の時にどのactionを選択するかを中心に、クラスタリングで共通の状態をまとめることで抽出した。

4. 試行

4.1. 試行 1

まず簡単で正解の自明な問題として、Figure 1 に示す空間で試行を行う。出口(赤色)は右上と左上の2か所にあるが、左上の出口は右上の出口の3倍の幅がある。この空間に右側に偏って人を初期配置し、右下の黄色で示す位置に矢印表示器を設定する。矢印の有効範囲は1マスとした。この場合、人を広い出口の左上に誘導するのが得策であるので、正解は左向きとなるのが自明であるが、強化学習によってそれが解けるか否かを確認する。移動戦略を Table 1 に、各種の確率を Table 2 に示す。試行 1 では、多くの人が出口を知っているものとして、全体人数の9割を占める typeA のエージェントは、戦略(1)を90%、戦略(2)を10%の確率でとり、残り1割の typeB のエージェントは、戦略をとる確率を反転させ、合計200人配置し、誘導はDQNを用いた。ネットワークへの入力には空間の状態(人:1,壁:-1,無:0)とし、ネットワーク構成は特徴量32の畳み込み層1層、全結合層3層(順にノード数512,256,3)とした。報酬 r は出口への到達総人数か

らその時点のステップ数の半分をペナルティとして引いた値としたが、学習が進むよう Q 関数の範囲を1以下に収めるため、全体の人数(200)×4で割った。 γ は0.9、学習率は0.00001、OptimizerはAdamとした。プログラミングにはTensorFlow¹⁶⁾を用いた。

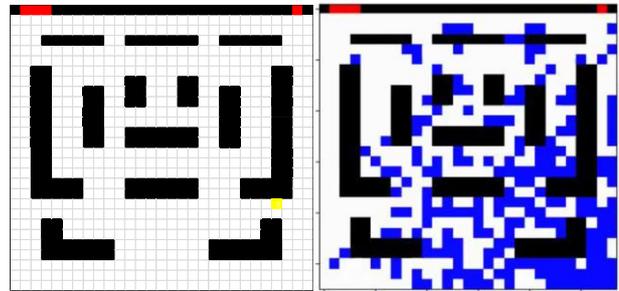


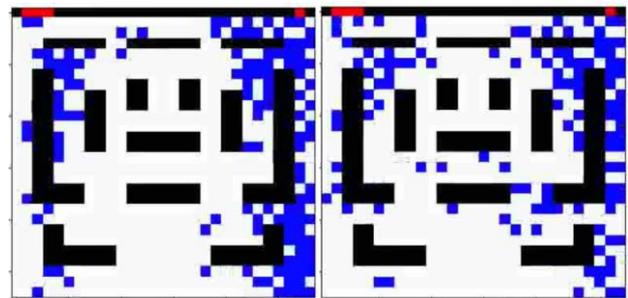
Figure 1. Experiment space (1) Figure 2. Initial position (1)

Table 1. Strategy definition (experiment 1)

	ratio	strategy1	strategy2
type A	90%	90%	10%
type B	10%	10%	90%

Table 2. Probability of action (experiment 1)

item	probability
forget the previous move	10%
change the exit	0.5%
follow the sign	80%
forget the sign	50%
change the exit by sign	80%



(a) no action (b) controlled

Figure 3. Progress of experiment 1

初期配置の状態を Figure 2 に青色で示す。Figure 3 に、誘導を行わない場合(no action)と、誘導を行った場合(controlled)の途中経過を示す。誘導している(b)では中央部に右から左へ向かう人群が見られるが、誘導していない(a)では、それが見られない。人の流れと誘導を200ステップに亘って繰り返す試行を、誘導を行わない場合(no action)、誘導を行った場合(controlled)、矢印を左向きに固

定した場合(all Left)の 3 通りでそれぞれ人流シミュレーションの乱数を変えて 1000 ケースずつ行い、各ステップでの出口に到達した人の累積数の平均値を Figure4 に示す。誘導を行った場合には 70 ステップ辺りから出口に到達する人が no action に比べて増えており、その傾向は矢印を左向きに固定した場合とほぼ同じであることから正解の学習ができていないと判断した。

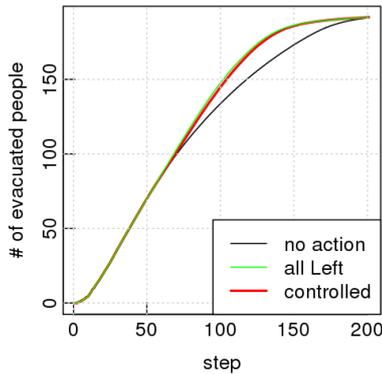


Figure 4. History of evacuated people (experiment 1)

4.2. 試行 2

より一般的な条件として、初期状態が左右で均一な条件で試行した。試行空間と矢印位置を Figure5 に示す。矢印が 5 か所で各々、非表示|左|右の 3 種類の action があり、何もしないを含めると action の総数は $3 \times 5 + 1 = 16$ である。action 0 は何もしない、action 1 は矢印 0 を非表示に、action 2 は矢印 0 を左向きにする等である。出口の条件は試行 1 と同様である。矢印の有効範囲を水色で示す。

試行 1 に比べ誘導無しで出口に到達する人が少なくなるよう、Table3 の通り、typeA を 80%、typeB を 20%とし、合計 300 人設定した。また各種確率については、Table4 に示すように、矢印の効果が出やすいように設定した。

誘導は当初 DQN で行ったが、統計的に有意な結果が得られなかったため、A3C を用いた。入力は空間を表す 2次元データの人を表す層(0/1)、左向きの矢印を表す層(0/1)、右向きの矢印を表す層(0/1)の3層とした。畳み込み層は2層とし、そのパラメータを Table5に示す。畳み込み

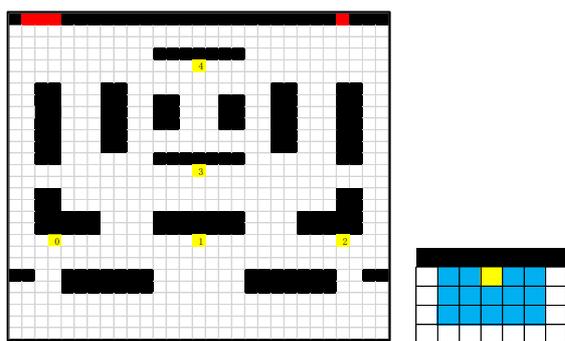


Figure 5. Experiment space (experiment 2)

Table 3. Strategy definition (experiment 2)

	ratio	strategy1	strategy2
type A	80%	90%	10%
type B	20%	10%	90%

Table 4. Probability of action (experiment 2)

item	probability
forget the previous move	10%
change the exit	0.5%
follow the sign	80%
forget the sign	20%
change the exit by sign	90%

Table 5. Specification of network (experiment 2)

layer	kernel size	stride	feature
1	8	2	16
2	4	1	32



Figure 6. Initial position (experiment 2)

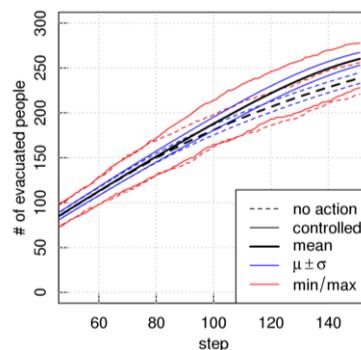


Figure 7. History of evacuated people (experiment 2)

層の後には256ノードの全結合層を配し、action 総数16のノードと Softmax 出力を持つ全結合層でポリシーを、1ノードでリニア出力の価値関数を表す。報酬は文献13を参考に当該ステップで出口に到達した人数とし、100で除した。 γ は0.98、学習率は0.0001、OptimizerはAdamとし、tmaxは4とした。プログラミングにはMXNet¹⁷⁾を用いた。

Figure6に初期状態を、最大150ステップの試行を1000回行った結果から Figure7に各ステップの出口に到達し

た累積人数の平均（黒）、平均±標準偏差（青）、最大最小値（赤）を示す。破線が no action、実線が誘導である。100 ステップ以降で見ると、no action の最大のケースよりも誘導した場合の平均値の方の人数が多くなっており、誘導による効果が見られる。最終ステップでの両者の平均値の差は標準偏差以上であるが、1000 回試行した場合の平均値の分散は 1/1000 となるので、統計的にも誘導の優位さが確認できる。tmax を増加させると学習の収束が早まる傾向があるが、最終的な人数には影響なかった。

4.3. 学習結果の分析

試行 2 の学習結果の分析を行った。

Figure8 に矢印操作の推移の一例を示す。各矢印で、グラフで下側が右向き、上側が左向きである。矢印 0 は、はじめ左向きで、中間に短時間右向きがあり、終盤は右向きとなっている。1000 回の試行の各ステップ各矢印の右向き、左向きの頻度の履歴を Figure9 に示す。全体的に左向きの頻度が高いが、終盤に向けて右向きの頻度が高くなっている。特に矢印 1（中央後方）は 80 ステップ辺りから右と左の頻度が拮抗している。これらの結果から、全体的には出口の幅が広い左側へ誘導するのが良いが、後半には逆に右側への誘導も有効となり、試行 1 のような一様な結果ではないことがわかる。

次に、どのような状態になった時に矢印を右向きもしくは左向きにするのが良いかの分析を試みた。まず、人の状態を K-mean 法で 4 つに分類し、action ごとに分類された各クラスの状態の平均値を計算した。Figure10 に action6（矢印 1 を右向き）の結果を示す。各クラスは、

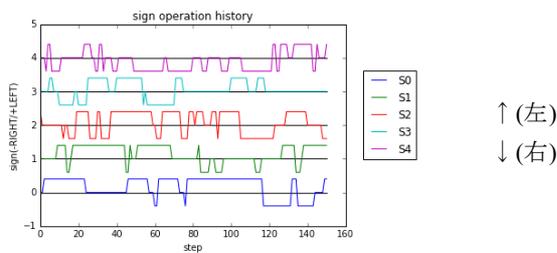


Figure 8. Sign operation history example

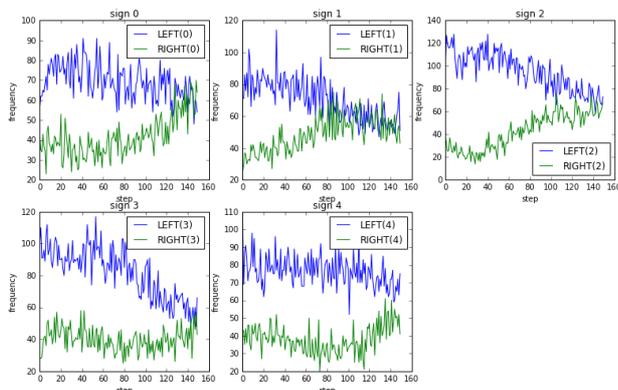


Figure 9. Frequency history of direction of signs

ステップの時期別に分類された。Figure10は、これをステップが進む順（No.0は初期、No.3は終盤など）に並べて表示している。

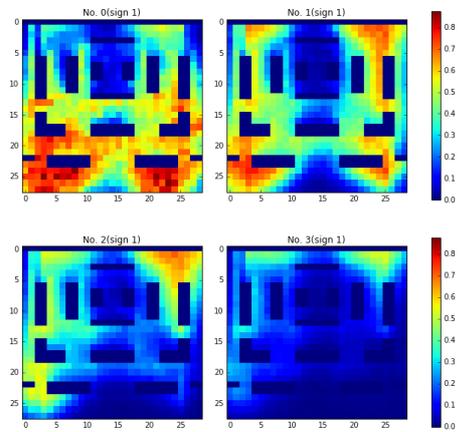


Figure 10. Clustering result of action 6 (sign 1)

次に、action6 と action5 の平均値の差を Figure11 に示す。これは、矢印 1 の右向きから左向きを引いたことに相当する。プラスの個所が矢印を右向きにした時に人が多く居た場所、マイナスの個所が矢印を左向きにした時に人が多く居た場所になる。Figure9 から、矢印 1 では後半に左向きと右向きが拮抗するが、Figure11 の No.2から、後半には左上に人が多い場合には右向き、右下に人が多い場合には左向きになっていることがわかる。

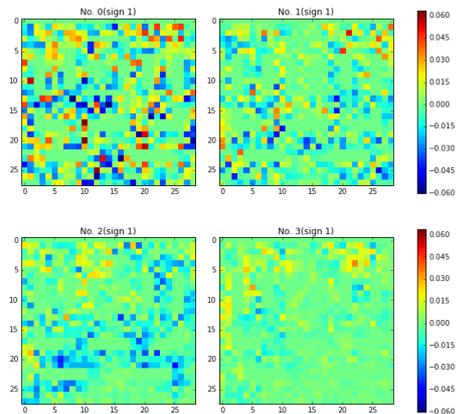


Figure 11. Pattern analysis

4.4. 限定された情報による学習

2 章でも述べたように人の流れをセンシングする技術は近年急激に進歩しているが、それでもすべての人の動きを把握できるようになるには時間がかかるものと思われる。そこで、試行 2 で一部の人の動きしか把握できないとした場合の影響を検討した。

検討ケースは、試行空間の中で、出口に近い方からそれぞれ 2/3, 1/2, 1/3 の範囲の人の動きだけを把握できる

ものと設定した。Figure12に、各ケースで1000回の試行をした場合の各ステップでの出口に到達した人数の平均値を示す。最終の150ステップで見ると、把握できる範囲が広いほど、出口に到達できる人が多くなっている。等分散を仮定しない2標本t検定であるウェルチ検定で各平均値の差の検定を行った結果、最終ステップの平均値は最も範囲の狭い1/3でさえ、no actionに比べて有意に大きかった。さらに、2/3の場合にはすべてを把握している場合とほとんど差がなかった。ただし、1/3の場合には逆に108ステップまでは誘導しない場合よりも出口に到達できる人が減少した。これは、前半ステップでは多くの人が1/3の範囲外に居るためと思われる。

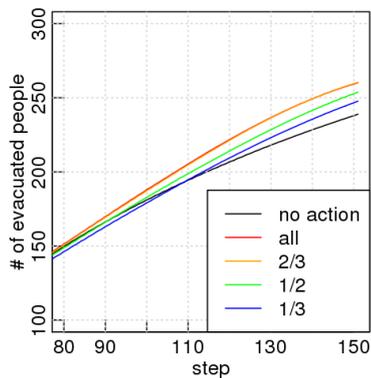


Figure 12. Effect of partial observation

5. おわりに

群衆誘導への深層強化学習の適用性を確認するため、28×29の空間と方向を自由に切り替えられる矢印表示装置を使った誘導を試行した。人の初期配置を偏らせた正解が既知の単純な試行1では、DQNを用いた誘導結果が、正解である常に左向き矢印による試行と同様の結果となり、誘導が可能であることがわかった。次に人の初期配置が偏っておらず矢印を単純にどちらかに向ければ良い訳ではない試行2では、DQNを用いた誘導はうまくゆかず、代わりにA3Cを用いた結果、統計的にも有意な誘導ができた。ただし、時々刻々と方向が変わる矢印に人がどのように反応するかという課題はある。また、今回の方向切替えはそれ程頻繁ではなかったが、頻繁な切替えを防ぐためにはペナルティ項の導入も検討したい。

さらに、深層強化学習の結果獲得した矢印操作について分析した結果、全体としては出口の幅の広い左側への誘導の頻度が多いものの、後半からは右側への誘導も見られ、どのような人の配置状況の場合に矢印を右向きとするのかという知見も得られた。

本研究では基礎的な検討のため28×29と言う小さな空間で試行を行った。それでも学習には日単位の時間がかかり、適用が期待される数千人、数万人規模の学習には何らかの工夫が必要である。また、シミュレーション

が現実の人の動きを忠実に反映していることが必要である。また、実際に誘導が可能となるためには誘導の際の人の流れを正確に把握している必要があり、それには計測技術の進歩が必要である。しかし、本研究で示した、シミュレーションと学習結果により誘導方法の知見を取り出す技術を発展させれば、リアルタイムの誘導が現時点では難しくとも、実際の誘導に応用することは可能であり、今後ともこのような技術を高度化していきたい。また、シミュレーションが単純で人命を扱う避難誘導を行えるレベルではないため本研究では退出誘導としているが、将来的には避難誘導にも展開していきたい。

謝辞

竹中工務店技術研究所の鈴木琢也氏、東城峻樹氏には論文執筆に当たりご助言頂きました。記して感謝致します

【参考文献】

- 1) 大西正輝ほか:カメラによる人流計測とシミュレーションによる人流予測に基づく人流解析に関する研究動向について、知能と情報, Vol.28, No.6, pp.187-193, 2016
- 2) 杉浦啓太ほか:最適配置問題とエージェントシミュレーションを利用する建物内の最適誘導法の研究,システム制御情報学会論文誌,Vol.28, No.5, pp.213-220, 2015.
- 3) 中島悠ほか: マルチエージェントシミュレーションを用いた避難誘導実験の拡張,情報処理学会論文誌, Vol.49, No.6, pp.1954-1961, 2008.6
- 4) Silver, D. et al.: Mastering the Game of Go with Deep Neural Networks and Tree Search, Nature 529, pp.445-446, 2016.
- 5) 伊藤毅志: ディープラーニングを用いたコンピュータ囲碁〜AlphaGoの技術と展望〜, 情報処理 Vol.57, No.4, 2016.4
- 6) 村上陽平ほか: インタラクション設計に基づくマルチエージェントシミュレーション,人工知能学会論文誌, Vol.18, No.5E, pp.278-285,2003
- 7) 小林重信: 特集「強化学習」にあたって, 人工知能学会誌, Vol.12, No.6, p.810, 1997,Nov.
- 8) Sutton, R.: Reinforcement Learning FAQ, <http://incompleteideas.net/RL-FAQ.html>, (accessed 2019-06-20)
- 9) Mnih, V. et al.: Playing Atari with Deep Reinforcement Learning, Deep Learning Workshop NIPS 2013, 2013
- 10) Sutton, R. S.: Learning to Predict by the Methods of Temporal Difference, Machine Learning 3, pp.9-44, 1988.
- 11) Watkins, C.J.C.H. et al.: Q-Learning, Machine Learning, 8, pp.279-292, 1992
- 12) Williams,R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning, Machine Learning, 8, pp.229-256, 1992
- 13) Mnih, V. et al.: Asynchronous Methods for Deep Reinforcement Learning, Proc. of the 33rd int. conf. on Machine Learning, NY, USA, 2016. JMLR: W&CP volume 48.
- 14) Wang, Z. et al.: Dueling Network Architecture for Deep Reinforcement Learning, eprint arXiv:1511.06581, 2015.11
- 15) 柳澤大地,西成活裕: 群衆運動のセルオートマトンモデル, 情報処理, Vol.58, No.7, pp.570-573, 2017.7
- 16) Google: TensorFlow, <https://www.tensorflow.org>, (accessed 2019-06-20).
- 17) Apache Software Foundation: Apache MXNet (Incubating), <https://mxnet.incubator.apache.org/>, (accessed 2019-06-20)