

6K-07

パフォーマンス低下を抑制するオンラインゲーム向け遅延補償技術に関する実験的評価

赤間 俊介[†] 本生 崇人[‡] 石岡 卓将[‡] 藤橋 卓也[‡] 猿渡 俊介[‡] 渡辺 尚[‡]

[†]大阪大学工学部 [‡]大阪大学大学院情報科学研究科

1 はじめに

家庭へのブロードバンドネットワークの普及によって、Esportsに挙げられるネットワークを介したオンラインゲームへの需要が増加している。オンラインゲームでは、一般的に、ネットワーク中で生じる遅延がプレイヤーのゲームパフォーマンスに影響を与える [1] [2]。本稿では、ネットワーク遅延がゲームパフォーマンスにもたらす影響を軽減する遅延補償技術を提案するとともに、3D ゲーム ViZDoom を利用して提案手法の効果を実験的に明らかにする。

2 オンラインゲームにおけるネットワーク遅延の問題

一般的に、オンラインゲームは、あるサーバにネットワークを介して複数プレイヤーが参加するクライアント・サーバ型と、サーバを介さずプレイヤー同士が接続するピアツーピア型に分けられる。ネットワークを介して、あるサーバに複数プレイヤーが同時参加して協力・対戦するクライアント・サーバ型のオンラインゲームでは、ゲームに参加するプレイヤー間でゲーム情報の同期を取ってからゲームに各プレイヤーのゲーム情報を反映する必要がある。サーバとあるプレイヤー間ネットワークで生じる遅延がプレイヤーのパフォーマンスに影響を与える。具体的には、サーバがネットワーク遅延のあるプレイヤーとネットワーク遅延の無いプレイヤーとの間でゲーム情報の同期を取るには、ネットワーク遅延のあるプレイヤーのゲーム情報を受信するまでサーバがゲームへの反映を待つ必要がある。プレイヤーのターンが順番に回ってくるゲームなど同期のために生じる待機遅延を許容可能なゲームでは、サーバにおけるゲーム情報の同期の遅れを演出によって対処できる。しかしながら、シューティングゲームなど、リアルタイム性が高く同期のために生じる待機遅延を許容できないゲームでは、サーバはある周期にしたがって各プレイヤーのゲーム情報を同期しないまま各時点の情報をゲームに反映する。ゲーム情報の同期ずれはプレイヤーのパフォーマンス低下を招く。

ネットワーク遅延によるゲーム情報の同期ずれを緩和する既存手法として、サーバ側でネットワーク遅延を補償する方法が提案されている [3]。本手法では、サーバが各銃弾に対して命中を判定するゲームにおいて、サーバが観測したクライアント間のネットワーク遅延と過去のプレイヤー位置情報から、ネットワーク遅延量に対応する過去のプレイヤー位置を命中判定に利用するプレイヤー位置としてみなす。サーバ側の遅延補償技術はネットワーク遅延によるプレイヤーパフォーマンスの低下を一部抑制できる。一方で、サーバ側の遅延補償技術は未だ実力以上あるいは実力以下のプレイヤーパフォーマンスを招く可能性がある。より具体的には、ネットワーク遅延発生時のプレイヤー位置を命中判定に利用するため、ネットワーク遅延発生後にあるプレイヤーが相手プレイヤーからの攻撃を回避するための操作をしても、その操作に関わらず銃弾はプレイヤーに命中する。

3 提案手法

本稿では、クライアント・サーバ型のオンラインシューティングゲームを対象としてネットワーク遅延によるプレイヤーパフォーマンスへの影響を軽減するために、ネットワーク遅延下における相手プレイヤーの将来の位置を推定する遅延補償技術を提案する。プレイヤー端末とサーバ間でやり取りするパケットに対してネットワーク遅延が発生すると、サーバが保持する相手プレイヤーの位置情報とプレイヤー端末が保持する相手プレイヤーの位置情報の間にずれが生じる。オンラインシューティングゲームにおけるプレイヤー位置情報のずれは、位置情報を元にしたプレイヤーによる射撃命中確率の低下、すなわちプレイヤーパフォーマンスの低下につながる。そこで、提案手法では、プレイヤーが本来のパフォーマンスを発揮できることを目的として、各プレイヤー端末が推定した相手プレイヤーの位置情報をプレイヤー端末上に表示する。プレイヤー端末は表示された相手プレイヤーの推定位置に向けて射撃することで、ネットワーク遅延による命中確率低下を抑制する。

3.1 ViZDoom

図 1 に本研究で利用するオンラインシューティングゲーム ViZDoom を示す。ゲームの種類は対戦型シューティングゲーム、障害物のない三次元のフィールド、プレイヤー数は 2 人である。ViZDoom はピアツーピア型またはクライアント・サーバ型のいずれかを選択できる。クライアント・サーバ型でゲームを実行する場合、各クライアント端末上で銃弾の命中判定を行う。ViZDoom は tic と呼ばれる時間単位でゲーム情報を管理する。サーバと各クライアントの端末間のネットワークは tc コマンドを使用することで任意の双方向遅延を設定できる。プレイヤーの移動方法として、キーボードとマウスを用いた人間による操作を可能としている。キーボードの↑キー、↓キー、←キー、→キーを用いて画面上のプレイヤーを前後左右斜めに移動できる。また、マウスを動かすことで上下左右斜めにエイム操作ができるとともに、マウスを左クリックすることでエイムに合わせて銃弾を発射する。

3.2 非線形回帰を用いた遅延補償技術

本手法では、プレイヤー端末が保持する相手プレイヤーの過去の位置情報を用いて 2 次の非線形回帰から相手プレイヤーの現在位置を推定する。具体的には、過去 10 個の相手プレイヤー位置情報を元にして回帰曲線を算出する。回帰曲線は python ライブラリである numpy に含まれる polyfit を用いて算出する。得られた回帰曲線、プレイヤー端末上での相手プレイヤーの位置情報、サーバ・プレイヤー端末間のネットワーク遅延の大きさにしたがって、遅延時間経過後に相手プレイヤーがいる位置を推定して画面上に表示する。なお、本方式では遅延時間の間、相手プレイヤーが回帰曲線にしたがってフィールドを移動し続けているものと仮定する。このとき、相手プレイヤーは得られた回帰曲線と現在の位置から遅延時間後に移動可能な位置との交点にいと予想できる。相手プレイヤーが 1 (tic) あたりに移動できるピクセル数を d (px)、ネットワーク遅延によってプレイヤー端末上に表示されている相手プレイヤーの位置情報が N (tic) 前の位置情報であると仮定すると、相手プレイヤーが N (tic) 後にいる位置は、現在の相手プレイヤーの位置を中心とする半径 dN の円上に相当する。半径 dN の円が得られた後、プレイヤー端末は回帰直線と半径 dN の円との交点を相手プレイヤーの位置と推定してフィールド上に反映する。このとき、推定した相手プレイヤー位置と実際の相手プレイヤー位置との間でずれが小さければ、プレイヤーは相手プレイヤーに対して正確に銃弾を命中させることができる。

本手法では、ネットワーク遅延の大きさに応じて、相手プレイヤーの位置情報を推定するパラメータ N を適切に定める必要がある。ネットワーク遅延に対して設定した N が小さすぎる、あるいは、大きすぎる場合、相手プレイヤーの位置を正確に推定できず、命中確率低下に起因するプレイヤーのパフォーマンス低下を招く。一方で、ネットワーク遅延に対して設定した N 次では、「偏差撃ち」と同じ現象がプレイヤーの実力に関係なく発生してしまうため、プレイヤーが持つ本来のパフォーマンス以上の命中確率を招く可能性もある。偏差撃ちとは、オンラインシューティングゲームに長けたプレイヤーが相手プレイヤーに攻撃するとき、移動方向・移動速度・銃弾速度を元に、あえて相手プレイヤーの位置から少しずれた位置に射撃して銃弾を命中させる技術である。

提案手法では、ネットワーク遅延に応じて、プレイヤーが本来のパフォーマンスを発揮できる N を適切に設定することを考える。本研究では、ViZDoom におけるゲーム情報の更新頻度が 35 (tic/s) であることを元にして、 $N = 35d$ と定めた。ここで、 d (s) はサーバ・クライアント間の遅延量である。

3.3 深層強化学習を用いた遅延補償技術

本手法では、ゲーム画面のスナップショットを入力とする深層強化学習 Rainbow [4] を用いて相手プレイヤーの操作を推定する。時刻 t におけるゲーム画面のスナップショットを f_t 、同時刻におけるプレイヤー i による操作を a_t^i とすると、提案手法では a_t^i を推定するために直近 4 枚分のスナップショット $\{f_{t-4}, \dots, f_{t-1}\}$ を入力として与える。各スナップショットは 180×240 画素のグレースケール画像とした。入力したゲーム画面のスナップショットに対して畳み込み層と活性化関数である ReLU (Rectified Linear Unit) を 3 回繰り返してゲーム画面内・ゲーム画面間の特徴を捉える。ここで、ReLU は負の値を 0 とする活性化関数 $f(x) = \max(0, x)$ である。その後、畳み込み層が出力した 3 次元テンソルを Reshape 層で 1 次元テンソルに変換する。得られた 1 次元テンソルは半分に分割してそれぞれを状態価値関数・アドバンテージ関数の



図 1: ViZDoom のゲーム画面のスナップショット

学習に用いる。最後に、状態価値関数から得られた出力とアドバンテージ関数の出力を結合した状態行動価値関数から相手プレイヤーによる操作 \hat{a}_i^t を推定する。各プレイヤー端末上では推定した相手プレイヤーの操作にしたがってゲーム画面 f_t を生成・表示する。

ここで、相手プレイヤーと同じ操作を推定できる学習済モデルを取得するためには報酬関数は下記の式にしたがうものとした。

$$l(\hat{a}_i^t, a_i^t) = \begin{cases} 1, & \hat{a}_i^t = a_i^t \\ 0, & \text{else} \end{cases}$$

具体的には、相手プレイヤーの操作と提案手法が推定した操作が一致したら報酬を 1、一致しなければ報酬を 0 とする関数である。

ネットワーク遅延が大きくなるにつれて、提案手法は複数手先の相手プレイヤーの操作を先読みする必要がある。しかしながら、従来の深層強化学習では、1 手先の相手プレイヤーの操作を推定することを前提としているため、ネットワーク遅延の増大に対処できない。本研究では、学習済モデルが一度推定した操作を相手プレイヤーが実行し続けることと仮定して相手プレイヤーの操作を先読みする。例えば、 t 番目および $t+1$ 番目のゲームフィールドに対する相手プレイヤーの操作を推定することを考える。まず、 $\{f_{t-4}, \dots, f_{t-1}\}$ を入力として学習済モデルが時刻 t における相手プレイヤーの操作 \hat{a}_i^t を推定する。推定した相手プレイヤーの操作を元にして相手プレイヤーの位置を更新したゲーム画面 f_t を生成する。その後、 $\hat{a}_i^{t+k} = \hat{a}_i^t$ ($k = 1, 2, \dots$) であると仮定して、相手プレイヤーの位置を更新した $t+1$ 番目のゲームフィールド f_{t+1} を生成する。

4 実験評価

4.1 評価環境

3.1 節で示した ViZDoom を用いて、ネットワーク遅延がオンラインシューティングゲームのプレイヤーパフォーマンスにもたらす影響を評価した。

ネットワーク遅延: 本評価では、tc コマンドを用いることでサーバに接続する 2 プレイヤ端末のうち、一方のプレイヤー端末とサーバとの間に 50 (ms), 100 (ms), 200 (ms), 300 (ms) の 4 種類のネットワーク遅延を与えた。このとき、もう一方のプレイヤー端末とサーバとの間のネットワーク遅延は 0 (ms) とした。

プレイヤーによる操作: ネットワーク遅延を 0 (ms) と定めたプレイヤーはゲーム画面上を左右に往復する行動を続ける。一方で、ネットワーク遅延を体感するプレイヤーは、直立したまま画面上に表示される相手プレイヤーの推定位置が自身の位置と直線上に表示されたとき、銃弾を発射するものとした。

比較手法: 比較手法として遅延補償を利用しない No Compensation (NC), 線形回帰を用いて遅延補償を実現する Linear Regression (LR) [2], 3.2 節に示した非線形回帰による遅延補償を実現する Non-Linear Regression (NLR), 3.3 節に示した深層強化学習による遅延補償を実現する Deep Reinforcement Learning (DRL) を用意した。

評価指標: 各遅延補償技術による効果を示す指標として、対戦時間中に、ネットワーク遅延を体感したプレイヤーが保持する相手プレイヤーの実際の位置情報と推定した位置情報とのずれの平均、ネットワーク遅延下にあるプレイヤーが発射した銃弾 20 発に対してネットワーク遅延下でないプレイヤーに命中した銃弾数の割合を示す命中率を利用する。

4.2 位置推定精度

図 2 にネットワーク遅延下にあるプレイヤー端末が各遅延補償技術を用いて推定した相手プレイヤーの位置と実際の相手プレイヤーの位置とのずれを示す。ここで、横軸はサーバとネットワーク遅延下にあるプレイヤー間の遅延の大きさ (ms)、縦軸は平均絶対誤差 (Mean Absolute Error: MAE) を示す。

評価結果から、以下に示す 4 つのことが分かる。

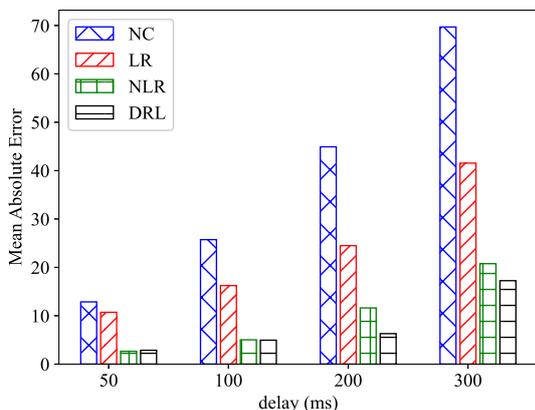


図 2: ネットワーク遅延に対する実際のプレイヤー位置と各遅延補償技術を用いて推定したプレイヤー位置との平均絶対誤差

- 遅延補償技術を利用しない場合、ネットワーク遅延が増大するにつれて平均絶対誤差が増大すること。
- プレイヤ端末上で遅延補償技術を用いることでネットワーク遅延の増加による平均絶対誤差の増加を抑制できること。
- 線形回帰を用いた遅延補償技術は平均絶対誤差の増加を抑制できるものの、その効果は限定的であること。
- 非線形回帰・深層強化学習を用いた遅延補償技術は線形回帰を用いた遅延補償技術と比較して平均絶対誤差を低減できること。

具体的には、ネットワーク遅延が 50 (ms) であるとき、遅延補償技術を利用しない場合の平均絶対誤差が 12.9、線形回帰・非線形回帰・深層強化学習による遅延補償技術を利用した場合の平均絶対誤差がそれぞれ 10.7, 2.7, 2.9 となること、ネットワーク遅延が 300 (ms) に増加したとき、遅延補償技術を利用しない場合の平均絶対誤差が 69.7、線形回帰・非線形回帰・深層強化学習による遅延補償技術を利用した場合の平均絶対誤差がそれぞれ 41.5, 20.8, 17.2 となることがわかった。

4.3 命中率

前節の評価結果から、提案手法を用いることでプレイヤー端末上に表示される相手プレイヤーの位置情報を正しく補償できることが分かった。本節では、補償した位置情報がオンラインシューティングゲームにおけるプレイヤーパフォーマンスにもたらす影響を評価する。本評価では、ネットワーク遅延下にある ViZDoom の操作に長けたプレイヤー 1 名が画面上に表示される相手プレイヤーの推定位置情報を元にして銃弾を 20 発発射したときの命中率を取得する。

図 3 にネットワーク遅延下において、各遅延補償技術を用いたときの命中率を示す。ここで、ネットワーク遅延が 0 (ms) であるときの命中率をネットワーク遅延下において再現するべきプレイヤーパフォーマンスであると定めた。評価結果から、遅延補償技術を利用しない場合、ネットワーク遅延が 50 (ms) 時において命中率が約 7 割まで低下していることが分かる。一方で、遅延補償技術を利用した場合、ネットワーク遅延がないときと同等の命中率が保たれていることが分かる。また、ネットワーク遅延が 100 (ms) まで増加したとき、線形回帰を用いた遅延補償技術は遅延補償技術を利用しない場合と同等の命中率まで低下するが、非線形回帰・深層強化学習を用いた遅延補償技術は命中率を約 6 割から約 8 割に維持できることが分かる。

5 おわりに

本稿では、オンラインシューティングゲームにおける過去のゲーム情報を用いてネットワーク遅延に起因するプレイヤーのパフォーマンス低下を軽減する手法を提案した。

謝辞

本研究は JSPS 科研費 JP20K19783 の支援の下で行った。

参考文献

- [1] M. Claypool, K. Claypool, "Latency and Player Actions in Online Games," Communications of the ACM, Vol. 49, No. 11, pp. 40–45, 2006.
- [2] 本生 崇人, 川崎 慈英, 藤橋 卓也, 猿渡 俊介, 渡辺 尚, "ネットワーク遅延がもたらすオンラインゲームプレイヤーへの影響に関する基礎評価," 電子情報通信学会ソサイエティ大会, p.1, 2019.
- [3] Y. Bernier, "Latency compensating methods in client/server in-game protocol design and optimization," Game Developers Conference, 2001.
- [4] M. Hessel, J. Modayil, H. Van Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver, "Rainbow: Combining improvements in deep reinforcement learning," in AAAI Conference on Artificial Intelligence, 2018.

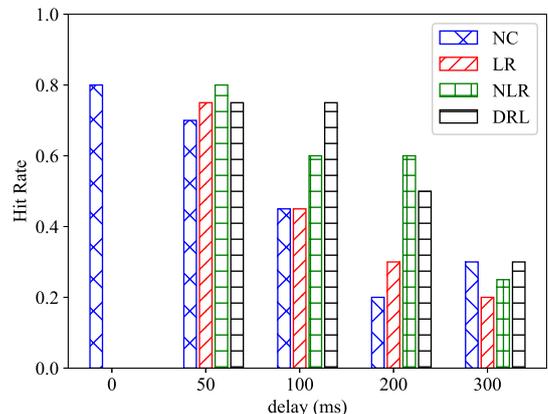


図 3: ネットワーク遅延に対して各遅延補償技術を用いたときの命中率