NEURON AS AN AGENT

Shohei Ohsawa, Kei Akuzawa, Yusuke Iwasawa & Yutaka Matsuo
The University of Tokyo
7 Chome-3-1 Hongo, Bunkyo, Tokyo
ohsawa@weblab.t.u-tokyo.ac.jp

ABSTRACT

The reason why swarm of agents solve real-world problem well is, interestingly, same as the principle of representation learning: good representation improves perforamance of machine learning. Most of the problem in real-world is not Markov decision process (MDP) but partially observed MDP (POMDP). On the POMDP environment, good observation yields good action. In this paper, we optimise a deep neural network as a multi-agent system as a natural extension from representation learning to multi-agent reinforcement learning for POMDP. To achieve that, we propose a novel learning framework, neuron as an agent (NaaA). In NaaA, an individual unit is considered as an agent, and they maximizing profit instead of minimizing error. To prevent dillemma, we borrow idea from machanism design, a field of game theory. To this end, we show all the unit have valid price reflecting their contribution to performance at convergence. We confirm the result by numerical experiment using Atari and VizDoom.

1 INTRODUCTION

マルチエージェント強化学習が現実の課題に対して有効である理由は、興味深いことに表現学習の原理と一致する：有用な表現は予測性能を向上させる。現実にある問題の多くは、DQNが前提としているマルコフ決定過程 (MDP) ではなく部分的観測マルコフ決定過程 (POMDP)である (Sutton & Barto, 1998, p.258)。POMDP において真の状態は完全には不可視であり、良質な観測結果が良質な行動に結びつくため、有限のリソースを使って何を観測するかの設計が必要である。必然的に、観測に用いられるエージェントが多いほど、将来的に獲得できる報酬が高いアクションを取りやすくなる。今後、IoT やブロックチェーンといったブレークスルーにより、自律分散化したセンサーネットワークが出現すると考えられ、環境に直接作用して報酬を獲得するエージェント以外に、センサーを使って観測するエージェント、得られたデータを分析するエージェントなどが出現すると考えられる。これらのエージェントを協調動作させ環境の観測範囲を広げることで、より実環境の問題を解きやすくなると考えられる。

マルチエージェントが観測した情報を活用してより精度の高い方策の決定を行うために、エージェント間を行き来する信号を工夫するコミュニケーションモデルの研究が重要である。CommNet (Sukhbaatar et al., 2016) は、エージェントが受け渡し合う信号にベクトルを用い、有用な信号をバックプロパゲーションによって学習することを可能にしている。これは、マルチエージェントシステム全体を一つのニューラルネットワークとみなしているのと同義である。彼らの手法は、すべてのエージェントは協力的で、一つの目的関数を最大化するという仮定を置いている。

CommNet のようなコミュニケーションモデルが解決していない問題は、情報の重要性に応じた報酬の分配であり、信頼度割り当て問題として知られている。各エージェントが同一報酬を受け取る設定は、有用な情報を提供しなくても報酬を受け取るフリーライダーの出現という問題を持つ。これは、エネルギーや通貨のように、エージェント全体が獲得できる報酬の総和が限られている場合に、よりシビアな問題となる。なぜなら、フリーライダーの存在が、他のエージェントの報酬を減らすためである。一般に、ジレンマの存在により、マルチエージェントシステムとしてニューラルネットワークを構築することは容易ではない。ジレンマとは、個別最適であるナッシュ均衡が全体最適であるパレート効率性を達成しない問題である。フリーライダー問題は、最終的に系がノイズのみを提供するエージェントで埋め尽くされる、コモンズの悲劇と呼ばれるジレンマを誘発する。

本研究では、報酬の分配にゲーム理論におけるメカニズムの一つである digital goods auction (Guruswami et al., 2005) を用いることで、自然にパレート効率性が達成されることを示す。ナッシュ均衡として、ユニットは予測誤差の最小化の代わりに、そのユニットが存在する場合と存在しない場合の系全体のリターンの差である counterfactual return を最大化する。これは、マルチエージェントシステムの報酬分配問題に対して提案されている指標である counterfactual reward (Agogino & Tumer, 2006) を時間方向に拡張したものである。

本研究のゴールは、すべてのユニットが自律的に動作すると仮定した場合に、システム全体が獲得する累積報酬（リターン）を最大化することである。これを、新しいタスクである *Neuron as an Agent* (NaaA) として定式化する。NaaA では、以下の前提を設ける。

環境の観測について、エージェントとユニットの役割は等価である。

すなわち、パーセプトロンや Convolutional Net (CNN) などの微分可能 (differentiable) な関数で構成されるニューラルネットワークの個々のユニットを、自身のリターンを利己的に最大化するエージェントであると仮定する。通常、ニューラルネットワークの目的は、目標値と予測値の間の誤差などの、共通の criterion e の最小化である。そのため、各ユニットはバックプロパゲーションによって出力 y による微分 $\partial e/\partial y$ を計算し、e が小さくなる方向に y の大きさを制御していた。NaaA では、メカニズムデザインの帰結として、各エージェントが自身の counterfactual return を最大化する。

NaaA のユニットは、counterfactual return の期待値を予測するための *valuation net* を持つ。すなわち、系全体は各ユニットがニューラルネットワークを持つ入れ子構造をしている。本論文では、valuation net の構成例や、強化学習を通して訓練する方法を述べている。

実験では、標準的な強化学習のタスクによる数値実験を用いて NaaA が POMDP の問題の精度を高めることを示す。具体的に、Atari および VisDoom における環境を用いて、既存研究が DQN や A3C を上回ることを示す。

2 Related Work

現在成功している深層強化学習のモデルの多くは、単一のエージェントが環境の観測から認知、行動決定といった一連のプロセスを担う。DQN (Mnih et al., 2015; Silver et al., 2016) は、Atari のスクリーン系列から最適な行動を決定したり、AlphaGo のモジュールとして囲碁の盤面から勝利に最も近い一手を選ぶ。DDPG (Lillicrap et al., 2015) は物理空間において摩擦や重力係数などの条件を考慮した多関節の制御を実現する。単一のエージェントを用いて強化学習を解くという試みは、人間の持つ身体性のアナロジーから考えると一見して妥当であるように思えるが、現実世界は open world であり、単一のエージェントが完全に情報が観測することが難しい。そのためマルチエージェントによるアプローチが求められている。

深層強化学習を POMDP 環境に適用する研究はいくつか行われている。Deep Recurrent Q-Network (DRQN) (Sorokin et al., 2015) は、隠れマルコフ連鎖を想定し、リカレントニューラルネットワーク (RNN) を用いて真の状態を推定している。

マルチエージェントシステムによる強化学習へのアプローチにはいくつかの方法がある。一つは、学習の効率を高めるために、同一のモデルに従う複数のエージェントを用いて探索を行う方法であり、Gorilla (Nair et al., 2015) , A3C (Mnih et al., 2016) などで採用されている。二つ目は、アクチュエーターを増やすことによって行動の量を増やす方法であり、サッカーゲーム (Kalyanakrishnan et al., 2006) やテレビゲーム (Tampuu et al., 2017) で行われている。三つ目は、センサーを増やすことによって観測の量を増やす方法であり、自動運転 (Sukhbaatar et al., 2016) やセンサーネットワーク (Fox et al., 2000) などで行われている。本研究は、観測困難な環境からいかに良い状態表現を得るかに注目しているため、三つ目のケースをスコープとする。

マルチエージェントで強化学習の問題を解決する場合には、信頼度割り当て問題の解決が重要になる。そこで、エージェントの信頼度を、そのエージェントがいた場合と、いなかったと仮定した場合の差として定量化する研究が行われている。QUICR-learning (Agogino & Tumer, 2006) では、エージェント i が reward $R(a_t)$ の代わりに、そのエージェントがある行動 a_{it} をとった場合 a_t と取らなかった場合 $a_t - a_{ti}$ の差、counterfactual reward $R(a_t) - R(a_t - a_{it})$ の cumulative discount summation を最大化している。COMA (Foerster et al., 2017) は、actor-critic において critic が共通しており、actor がマルチエージェントであるという actor-critic の仕組みを考え、それぞれの actor が counterfactual reward を最大化するような仕組みを考えている。

マルチエージェントによる観測および認知の枠組みは、センサー処理の分野ではエッジコンピューティング (Bonomi et al., 2012) としても知られている。エッジコンピューティングは分散環境を前提とした信号処理のモデルであり、一つの処理系がすべてのデータを処理するのではなく、複数のセンサーの情報を一つのエッジサーバが集約し、複数のエッジサーバが次元削減したデータをデータセンターに送るという階層的な構造をしている。本研究は、実際に IoT の環境で本研究が適用されるケースを想定している。報酬はビットコインなどの決済手段によって送金が可能であるため、Web 全体でスケーラブルなモデルが実現できる可能性がある。

NaaA の設計は、神経科学からヒントを得ている。神経回路に含まれるニューロンは一つの細胞であるため、エネルギーを消費する。通常の細胞と同様に酸素や ATP がエネルギー源となり、これらはニューロンと接続した、アストロサイトから供給される。アストロサイトは脳の構造を支えるグリア細胞の一種であり、血管からニューロンへの栄養供給を行う。エネルギー量は有限であるため、不要なニューロンはアポトーシスによって死滅する。アポトーシスは NGF (nerve growth factor), BDNF (brain derived neurofactor) などの神経栄養因子 (neurotorophin; NTF) によって制御されるため、より多くの NTF を獲得できたニューロンが生存する。各神経細胞を独立した生物として捉える見方はニューラルダーウィズム (Edelman, 1987) と呼ばれる。

3 BACKGROUND

まず、単一のエージェントが PODMP 環境で行動する場合について考える。POMDP 環境とは 7 つ組 $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, \mathcal{O}, \gamma)$ である。ただし、$\mathcal{S}$ は状態集合、$\mathcal{A}$ は行動集合、$\mathcal{T}$ は遷移確率、Ω は取りうる観測の集合、$\mathcal{O}$ は観測の集合、γ は減衰率である。エージェントは観測 $o \in \Omega$ を通して部分的に状態 $\mathcal{S}$ を推定する。一般に o の方が s よりも次元が大きく、複雑である。たとえば、Atari 2600 は真の状態である RAM は 128 バイトしかないが、そこから生成される画像 o は 10,000 以上の次元を持っている。そのため、DQN や DRQN では、o の情報を抽象化し、独自の状態表現を作っていると解釈できる（DQN の原著論文では MDP であることを前提としているが、DRQN の論文で環境が POMDP であることが主張されている）。もちろん、DQN はモデルフリーの手法であるため、直接は状態遷移を扱わないが、出力層の一個手前の層に状態が格納されているという解釈もできる (Zahavy et al., 2016)。以下では、エージェントはニューラルネットワークを通して行動を決定することを前提とする。

次に、互いに通信するマルチエージェントシステムを考える。一般にエージェントが多いほど、観測を増やすことが可能である。たとえば、自動運転のケースでは自動車同士が通信することでより正確な世界に対する知識を得ることができる。この時、エージェントが持っているニューラルネットワーク同士をつなげる方法がとられている (Sukhbaatar et al., 2016)。これは、マルチエージェントシステム全体を一つのニューラルネットワークをとらえることができると考えられる。そこで、本研究ではこれを拡張し、すべてのユニットをエージェントとみなす。

4 NEURON AS AN AGENT

ニューラルネットワークを、ユニット間の有向グラフ $\mathfrak{G} = (\mathcal{V}, \mathcal{E})$ で表す。$\mathcal{V} = \{v_1, \ldots, v_N\}$ はユニットの集合であり、$\mathcal{E} \subset \mathcal{V}^2$ はユニットの接続関係を表すエッジの集合である。$(v_i, v_j) \in \mathcal{E}$ であるとき、$v_i \to v_j$ という接続関係が成立し、v_j は v_i から値を入力する。ユニットの v_i の時刻 t における出力を $x_{it} \in \mathbb{R}$ で表す。ユニット i の出力先の集合を $N_i^{\text{out}} = \{j | (v_i, v_j) \in \mathcal{E}\}$、n 入力元の集合を $N_i^{\text{in}} = \{j | (v_j, v_i) \in \mathcal{E}\}$ で表現する。$N_i = N_i^{\text{in}} \cup N_i^{\text{out}}$ とする。

NaaA は v_i をエージェントとしてとらえる。すなわち、$\mathfrak{G}$ はマルチエージェントシステムである。v_i にとっての環境は、マルチエージェントシステムの自体が触れている環境と、v_i が直絶接続しているユニット群 $\{v_i \in V | i \in N_i\}$ である。前者を外的環境 (external environment)、後者を内的環境 (internal environment) と呼んで区別する。v_i は環境から報酬を受け取る。v_i の性質として以下の前提を加える。

- N1: （利己性）v_i は、各時点 t において、汎化誤差の最小化ではなく、自身のリターン（累積減衰報酬）$G_{it} = \sum_{k=0}^{T} \gamma^k R_{i,t+k}$ の最大化を目的として行動する。ただし $\gamma \in [0, 1]$ は減衰率 (discount rate)、T は終端時間である。
- N2: （保存則）v_i が環境から受け取る報酬 R_{it} の総和は、マルチユニットシステム全体が外的環境から得る報酬 R_0 に等しい。

N3: （取引）v_i は信号 x_i を $v_j \in \mathcal{V}$ に伝達する際に、信号と引き換えに報酬 ρ_{jit} を受け取る。

N4: （NOOP）v_i は、期待リターンが 0 の NOOP (no operation) という行動をオプションとして持つ。NOOP では、ユニットは何も入力せず、何も出力しない。

N1 はユニットがエージェントとして振る舞うことを述べている。N2, N3 は NTF の分配に、N4 はニューロンのアポトーシスに相当する。NOOP が選択されるのは、それ以外のすべての行動の期待報酬が負であった場合である。以下ではこれらの前提から出発して、NaaA の仕組みを構築していく。

4.1 Cumulative Discounted Profit Maximization Framework

ユニット i が時刻 t で外部から得る報酬を R_i^{ex} と書き、消費エネルギーを α_{it} で表す。時刻 t に i が獲得する報酬 R_i は次のように表現される。

$$R_{it} = \left[R_{it}^{\text{ex}} + \sum_{j \in N_i^{\text{out}}} \rho_{jit} \right] - \left[\sum_{j \in N_i^{\text{in}}} \rho_{ijt} + \alpha_{it} \right]. \tag{1}$$

この式は、符号が正の項と負の項の二つに分解される。前者を収益 (revenue)、後者をコスト (cost) と呼び、それぞれ r_{it}, c_{it} で表す。R_{it} を利益 (profit) と呼ぶ。

このとき、ユニット v_i は、次式で表現される累積減衰利益 G_{it} を最大化する。

$$G_{it} = \sum_{k=0}^{T} \gamma^k R_{i,t+k} = \sum_{k=0}^{T} \gamma^k (r_{i,t+k} - c_{i,t+k}). \tag{2}$$

$R_{it} > 0$、すなわち、$r_{it} > c_{it}$ であれば、ユニットは得たデータに対して付加価値を与えていることになる。もし、すべての t に対して $R_{it} < 0$ であれば、$G_{it} < 0$ であるから、ユニットは NOOP になる。

5 Optimization

NaaA では利益を最大化するため、二つの相反する指標である収益 r_{it} とコスト c_{it} のバランスを取ることが重要になる。本研究では、この最適化にゲーム理論の一つであるメカニズムデザインを応用する。メカニズムデザインは、マルチエージェントシステムを対象にしたゲーム理論の分野であり、各エージェントが利己的であることを想定した上で、システム全体が最適になるような帰結を目指すメカニズムの設計を目的とするものである。

メカニズムデザインを導入する理由は、NaaA いくつかの既存研究と異なり、すべてのエージェントが協力的ではなく、利己的であると仮定していることに起因している。前述の問題は、そのまま最適化すると報酬額は 0 に収束するため、すべてのニューロンが NOOP になるという trivial な解が得られる。マルチエージェントシステムは無情報でアクションを選択する必要が生じ、これはランダムなアクションを取っている状況に等しい。したがって、明らかに外的環境からの報酬 R_t は小さくなる。

このように最適化を行った結果、システム全体が最適化されない現象はジレンマとして知られており、囚人のジレンマ問題をはじめとし様々な研究が行われている。しかし、一般にエージェントが利己的であると仮定した場合の最適化は難しいとされている。メカニズムデザインはこうした問題を解決することができる。

5.1 Envy-free Auction

パレート効率な仕組みを作るために、我々はオークション理論における digital goods auction からアイデアを借りる。オークション理論は、ゲーム理論におけるメカニズムデザインという分野に属しており、複数のエージェントの利害を一致させ、全体としてパレート最適を目指すことを目指している。digital goods auction は、本や音楽などの、複製可能な財を割り当てる仕組みを作っている。

Digital goods auction にはいくつかバリエーションがあるが、本研究では単純な前提のみを設けるだけでよいという理由から envy-free auction (Guruswami et al., 2005) を用いる。これは、

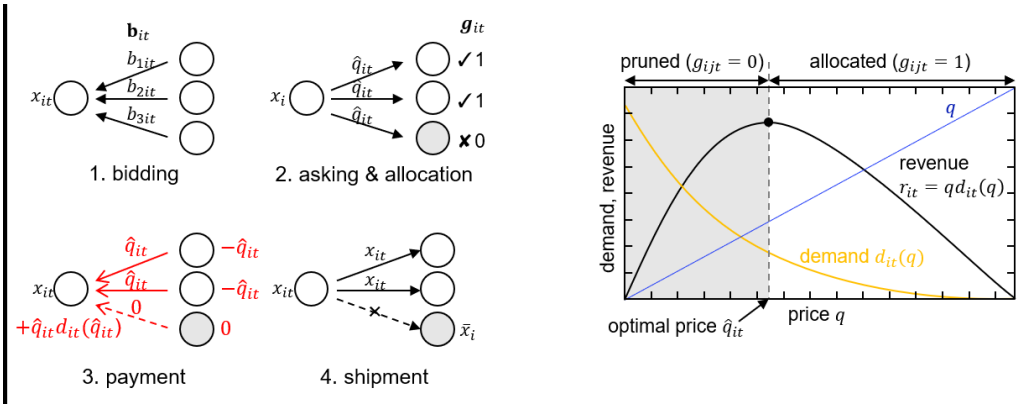

Figure 1: Left: NaaA による取引の流れ。Right: ユニットの価格決定方法。ユニットの収益は単調減少な需要と価格の積となり、これを最大化する価格が最適価格となる。

同じ時点取引において、一つのユニットの価格を同じにするというものである。NaaA において、これは次の前提によって表現できる。

N5: （一物一価）$\rho_{j_1,i,t}, \rho_{j_2,i,t} > 0$ であれば $\rho_{j_1,i,t} = \rho_{j_2,i,t}$

これは、ユニット v_i は同じ時間 (timing) t に個有の価格を持つことを意味する。この価格を q_{it} で表す。

Envy-free auction の流れを Figure 1 の左に示す。図は、信号を送信する一つのユニットと、その信号を「購入」する複数のユニットに分かれ、交渉の過程を示している。一単位の交渉は、強化学習の時間軸では 1 ステップ内に完了し、これが複数回繰り返されることになる。信号を送信する側を売り手、受信する側を買い手と呼ぶ。買い手はユニットに対して入札 b_{jit} を行う (1)。次に、入札額をもとに、売り手は価格 $\hat{q}_{it}$ を決定し、割当を行う (2)。このとき、$b_{jit} \geq \hat{q}_{it}$ であれば割当を行って $g_{jit} = 1$ とし、そうでなければ $g_{jit} = 0$ とする。割当を行った後は、$\rho_{jit} = g_{jit}\hat{q}_{it}$ として、送金を行い (3)、売り手は割当を行ったノードに対してのみ信号 x_i を送付する (4)。信号を受け取れなかったノードは、x_i の期待値 $\mathbb{E}_\pi[x_i]$ によって x_i を近似する。

G_{it} は次のように書くことができる。

$$\begin{aligned} G_{it} &= r_t - c_t + \gamma G_{i,t+1} \\ &\approx r_t - c_t + \gamma V_i^{\pi_i}(s_{i,t+1}), \end{aligned} \tag{3}$$

ただし、$V_i^{\pi_i}(s_{it})$ は価値関数 (value function) であり、ユニットの状態 s_{it} に対して方策 π_i を取った場合のリターンの期待値に等しい。このため、収益、コスト、価値関数それぞれを最大化する方法について考えればよい。

Revenue: エージェントの収益は次式で与えられる。

$$\begin{aligned} r_{it} &= \sum_{j \in N_i^{\text{out}}} g(b_{jit}, q_{it}) q_{it} + R_i^{\text{ex}} = q_{it} \sum_{j \in N_i^{\text{out}}} g(b_{jit}, q_{it}) + R_i^{\text{ex}} \\ &= q_i d_{it}(q_t) + R_i^{\text{ex}}, \end{aligned} \tag{4}$$

ただし、$g(\cdot,\cdot)$ は割当 (allocation) であり、ステップ関数 $H(\cdot)$ を用いて $g(b,q) = H(b-q)$ によって定義される。q_{it} は価格、$d_{it}(q_{it})$ はユニット i の信号に対する価値を q_{it} 以上と評価しているエージェントの数であり、需要 (demand) と呼ぶ。同様の式で、右辺を最大化する a を最適価格と呼び、$\hat{q}_{it}$ で表す。第二項は q_t に対して独立であるから、最適価格 $\hat{q}_{it}$ は次のようにして与えられる。

$$\hat{q}_{it} = \underset{q \in [0,\infty)}{\operatorname{argmax}}\, q d_{it}(q). \tag{5}$$

この仕組みを、Figure 1 の右に図示する。$d_t(q_{it})$ は単調減少な関数であり、q_{it} との積によって表現される。

Cost: コストは、ユニットが他のユニットに対して払う価格である。これは次のように表示される。

$$c_{it} = \sum_{j \in N^{\text{in}}} g(b_{ijt}, q_j) q_j + \alpha_i \tag{6}$$

c_{it} 自体は $b_{ijt} = 0$ のとき最小となる。しかし、これは次の value function とトレードオフをなす。

Value Function: 価値関数は $V(s_{i,t+1})$ の値は $s_{i,t+1}$ に依存する。既に述べたようにエージェントの v_i の環境は接続されているユニット集合であり、ユニットの出力はこれらのエージェントからの評価、すなわちエッジの重みに影響を及ぼす。通常のニューラルネットワークでは、出力の精度に貢献しないニューロンの重みは小さくなることから、報酬は小さくなる。したがって、入札価格 b_{ijt} を最小化し 0 と置くとデータの購入に失敗し、将来的にエージェントが接続しているエージェントから得られる報酬が小さくなる。

今、割当を $\mathbf{g}_{it} = (g_{i1t}, \ldots, g_{iNt})^{\mathrm{T}}$ で表し、エージェントが v_j の購入に成功した場合と、そうでない場合に価値関数に及ぼす影響について考える。この時、価値関数は、状態価値関数 $Q(s_{i,t+1}, \mathbf{g}_{i,t+1})$ を用いて次式で表現できる。

$$\begin{aligned} V_i^{\pi_i}(s_{it}) &= Q_i^{\pi_i}(s_{it}, \mathbf{g}_{it}) \\ &= \sum_{j \in N_i^{\text{in}}} g_{ijt}(Q_i^{\pi_i}(s_{it}, \mathbf{e}_j) - Q_i^{\pi_i}(s_{it}, \mathbf{0})) + Q_i^{\pi_i}(s_{it}, \mathbf{0}) \\ &= \sum_{j \in N_i^{\text{in}}} g_{ijt} o_{ijt} + Q_i^{\pi_i}(s_{it}, \mathbf{0}) \\ &= \mathbf{g}_{it}^{\mathrm{T}} \mathbf{o}_{it} + Q_i^{\pi_i}(s_{it}, \mathbf{0}) \end{aligned} \tag{7}$$

$o_{ijt} = Q_i^{\pi_i}(s_{it}, \mathbf{e}_j) - Q_i^{\pi_i}(s_{it}, \mathbf{0})$ を counterfactual return と呼ぶ。これは QUICR (Agogino & Tumer, 2006) と導出は異なるが等価である。すなわち、エージェントが支払うコストは、データの購入に成功した場合は $\hat{a}_{it}$ であり、それ以外は o_{it} となる。

以上から、最適化問題は次のように書くことができる。

$$\max_{\mathbf{b}, q} G_{it} = \max_{q} q d_{it}(q) - \min_{\mathbf{b}} \mathbf{g}_{it}(\mathbf{b})^{\mathrm{T}}(\mathbf{q}_t - \gamma \mathbf{o}_{i,t+1}) + \text{const.} \tag{8}$$

では、リターンを最大化するためのエージェントの入札額 b_{it} は何か。これについては次の定理が成立する。

Theorem 5.1. *(Truthfulness)* リターンを最大化する最適な入札額は $\hat{\mathbf{b}}_{it} = \mathbf{o}_{it}$ である。

証明については Appendix を参照。

すなわち、エージェントは自身の counterfactual return のみを問題にすればよい (!) したがって、NaaA のメカニズムでは、エージェントはあたかも他のエージェントを価値評価 (valuation) し、その価値を正直に申告していることを意味する。

系として次の解が得られる。

Corollary 5.1. *The Nash equivalem of the envy-free game* $(\mathbf{b}_{it}, q_{it})$ *is* $(\mathbf{o}_{it}, \operatorname{argmax}_q q d_{it}(q))$.

5.2 VALUATION NET

残る問題は、$\mathbf{o}_t$ をいかに推定するかである。この推定には様々な方法が存在しており、多くのメソッドを使うことができるが、本論文では Q の推定に Q-learning を採用する。ただし、SARSA や actor-critic などの on-policy な方法も使うことができることを補足する。

図 2 に示す Valuation Net は、通常のニューラルネットワークのユニットに、Q-learning による valuation を組み合わせたネットワークである。まず、上部はエージェント間の通信について示したものである。ニューラルネットワークではユニットを円で表現するのが通例であるが、ここではユニットをエージェントとしてみなすことを強調して、六角形で一つのユニット

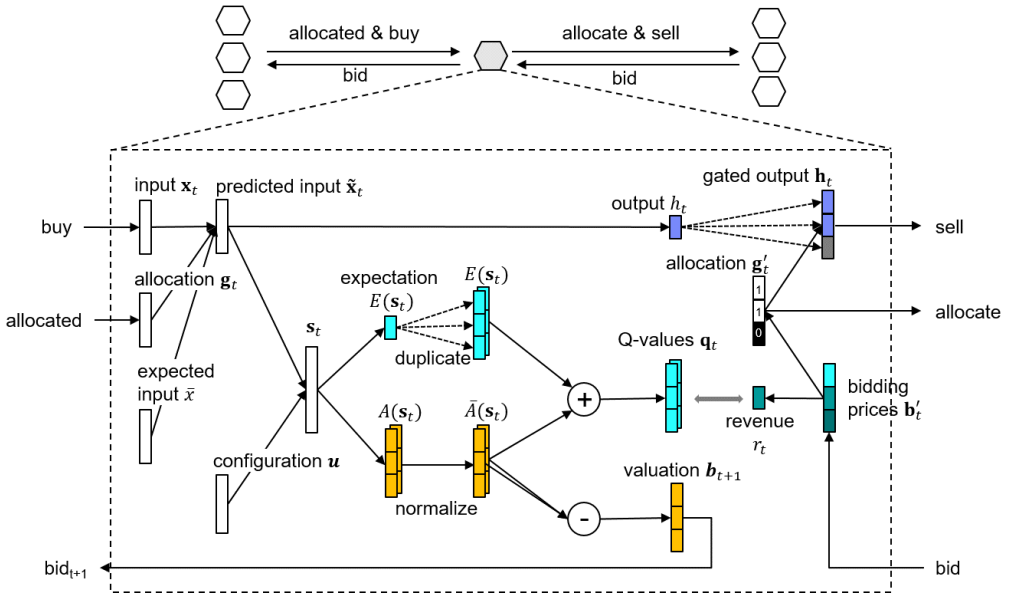

Figure 2: Valuationn Net は情報の価値を評価し、bidding price を決定する。下位のニューロンに対して入札し、信号を購入する。購入したデータを用いて、データを次のニューロンに対して売る。

を表現している。エージェント間では、通常のニューラルネットワークと同様の信号の通信以外に、取引に関する通信 (allocate, buy, sell & bid) が発生する。

Valuation Net では、状態 $\mathbf{s}_t$ として、予測後入力 $\tilde{\mathbf{x}}_t$ および入力に依存しない構成情報 $\mathbf{u}$ を横につなげたベクトル $(\tilde{\mathbf{x}}_t^{\mathrm{T}}, \mathbf{u}^{\mathrm{T}})^{\mathrm{T}}$ を用いる。構成情報の一例としてはユニットのパラメータがあげられ、たとえば重みやバイアスの情報を用いることができる。

状態からの Q 関数の予測にニューラルネットワークを用いる。エージェントが受け取った収益に基づき時間差分 (TD)-誤差 が計算され、ネットワークが訓練される。ネットワークの構成にはこれまでの deep Q-learning で用いられている二重化ネットワーク (dueling network) (Wang et al., 2015) のテクニックを用いる。オリジナルの文献 (Wang et al., 2015) で述べられている二重化ネットワークは、学習を加速するために、状態関数と、Q 関数との差分を別々に予測する手法である。Dosovitskiy & Koltun (2016) はこれに対して、差分の要素の総和が 0 になるように正規化するよう改良している。本研究では Dosovitskiy & Koltun (2016) の手法に従い、期待値 $\mathcal{E}(\mathbf{s}_t)$ と正規化差分 $\tilde{A}(\mathbf{s}_t)$ を別々に求める。

Q 関数は次のように表現される。

$$Q(\mathbf{s}_t, a_t) = \mathcal{E}(\mathbf{s}_t) + \tilde{A}(\mathbf{s}_t, a_t)$$
$$\sum_{i+1}^{k} \tilde{A}_i(\mathbf{s}_t, a_t) = 0 \tag{9}$$

第 2 式を満たすために、まず、$\mathbf{s}_t$ に基づいた予測を行い、次のような正規化を行う。

$$\tilde{A}_i(\mathbf{s}_t, a_t) = A_i(\mathbf{s}_t, a_t) - \frac{1}{k}\sum_{j=1}^{k} A_j(\mathbf{s}_t, a_t) \tag{10}$$

次に、valuation を行い、bidding price $\mathbf{b}_t$ を求める。$\hat{b}_{ijt} = o_{ijt}$ 式 9 より、最適な入札価格 $\hat{b}_{it}$ は次のように計算できる。

$$\hat{b}_{ijt} = \tilde{A}(\mathbf{s}_t, 1) - \tilde{A}(\mathbf{s}_t, 0) \tag{11}$$

Valuation Net ではこの式に基づき、advantage の出力を引き算することで入札価格を計算している。

6 EXPERIMENT

7 DISCUSSION

7.1 DISADVANTAGE

Disdvantage としてまず挙げられるのは計算量である。Envy-free auction では需要の計算にソートの演算が入るために、直列化しなければならない箇所があるため、これらについては近似を行うなどして改善していく必要がある。

個別の最適化技術について述べると、Envy-free auction は、買い手のエージェント同士の価格がわからない sealed な状態であれば、正直性 (truthfulness) が成り立つが、一方で買い手同士がコミュニケーションを行い価格を共有し合う状態においては、買い手が自由に価格を偽装できることが知られている。これについては、Goldberg et al. (2006) によって解決方法が示唆されている。

Valuation Net は、用いるニューラルネットワークによっては実装が困難であることがある。これは著者らの GitHub に Linear と CNN は公開しているが、RNN などについては今後の研究課題となる。

7.2 APPLICATION

NaaA は、ネットワークが分散されている環境での学習や、サブモジュールでの制御に有用である。具体的に、以下の技術に応用が可能である。

- ハイパーパラメータチューニング。Neuroevolution など、遺伝的アルゴリズムを用いてハイパーパラメータチューニングを用いるアルゴリズムがすでにいくつか提案されている。このとき、fitness 関数として利益を用いることで、より強化学習の目的に特化したニューラルネットワークを得ることができると考えられる。
- pruning, dilusion などのネットワークの規模の縮小。
- アテンション制御。一部のアテンションの研究では、強化学習を用いてアテンションの制御を行っている。
- アンサンブル。複数のモデルの混合に今回の技術を用いることができる。

これらの応用に関しては、今後の研究の方向性である。

8 CONCLUSION AND FUTURE WORKS

本論文では、POMDP の問題設定において良質な特徴表現を得るために、ニューラルネットワーク上の各ユニットをエージェントとして扱うフレームワーク、NaaA について述べた。NaaA のフレームワークでは、ジレンマ問題を解決し、それぞれのエージェントの持つ付加価値がナッシュ均衡として得られ、全体としてパレート最適になることを示した。入札価格の決定アルゴリズムの一つとして、Q-learning に基づくネットワーク Valuation Net を示した。評価実験では、Atari と VizDoom を用いた実験を行い、実験結果が既存手法よりもよくなることを示した。

今後の方向性として、高速化、Valuation Net を A3C などの on-policy な手法で置き換えるといった方向性の他、神経科学的な説明を可能にしていくといった方法、遺伝的アルゴリズムとの組み合わせが挙げられる。

APPENDIX

A.1 定理 5.1 の証明

買い手の獲得する生涯報酬 G は次で与えられる。

$$G(b, q) = g(b, q) \cdot (v - q) + G_0, \tag{12}$$

ただし、g は割当 (allocation) であり、G_0 はユニットを購入しなかった場合の生涯報酬である。割当は、$g(b,q) = H(b-q)$ が成立する。H はステップ関数である。

買い手にとって売り手が提示する asking price q は未知であるため、q を台 $[0,\infty)$ の上の確率変数であるとして扱い、期待値 $\mathbb{E}_q[G(b,q)]$ を最大化することを考える。このとき、

$$\begin{aligned}\frac{\partial}{\partial b}\mathbb{E}_q[G(b,q)] &= \frac{\partial}{\partial b}\int_0^\infty (H(b-q)\cdot(v-q)+G_0)p(q)dq \\ &= \frac{\partial}{\partial b}\left[\int_0^b (v-q)p(q)dq + G_0\int_0^\infty p(q)dq\right] \\ &= \frac{\partial}{\partial b}\int_0^b (v-q)p(q)dq \\ &= (v-b)p(q=b)\end{aligned}$$

したがって、 $\mathbb{E}_q[G(b,q)]$ が最大となるための条件は $b=v$ ある。

REFERENCES

A. K. Agogino and K. Tumer. QUICR-learning for multi-agent coordination. AAAI'06, 2006.

F. Bonomi, R. Milito, J. Zhu, and S. Addepalli. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pp. 13–16. ACM, 2012.

A. Dosovitskiy and V. Koltun. Learning to act by predicting the future. *ICLR'17*, 2016.

G. M Edelman. *Neural Darwinism: The theory of neuronal group selection.* Basic books, 1987.

J. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson. Counterfactual multi-agent policy gradients. *arXiv:1705.08926*, 2017.

D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous robots*, 8(3):325–344, 2000.

Andrew V Goldberg, Jason D Hartline, Anna R Karlin, Michael Saks, and Andrew Wright. Competitive auctions. *Games and Economic Behavior*, 55(2):242–269, 2006.

Venkatesan Guruswami, Jason D Hartline, Anna R Karlin, David Kempe, Claire Kenyon, and Frank McSherry. On profit-maximizing envy-free pricing. In *ACM-SIAM symposium on Discrete algorithms*, 2005.

Shivaram Kalyanakrishnan, Yaxin Liu, and Peter Stone. Half field offense in robocup soccer: A multiagent reinforcement learning case study. In *Robot Soccer World Cup*, pp. 72–85. Springer, 2006.

T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous control with deep reinforcement learning. *ICLR'16*, 2015.

V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *ICML'16*, pp. 1928–1937, 2016.

A. Nair, P. Srinivasan, S. Blackwell, C. Alcicek, R. Fearon, A. De Maria, V. Panneershelvam, M. Suleyman, C. Beattie, S. Petersen, et al. Massively parallel methods for deep reinforcement learning. *ICML'15*, 2015.

D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

I. Sorokin, A. Seleznev, M. Pavlov, A. Fedorov, and A. Ignateva. Deep attention recurrent q-network. *arXiv:1512.01693*, 2015.

S. Sukhbaatar, R. Fergus, et al. Learning multiagent communication with backpropagation. In *NIPS'16*, 2016.

R. S. Sutton and A. G Barto. *Reinforcement learning: An introduction*. A Bradford Book, 1998.

A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente. Multiagent cooperation and competition with deep reinforcement learning. *PloS one*, 12(4):e0172395, 2017.

Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Freitas. Dueling network architectures for deep reinforcement learning. *arXiv:1511.06581*, 2015.

T. Zahavy, N. Ben-Zrihem, and S. Mannor. Graying the black box: Understanding DQNs. In *ICML'16*, 2016.