Rola i techniki eksploracji w uczeniu przez wzmacnianie

Piotr Januszewski[0000−0003−3817−3479]

Katedra Architektury Systemów Komputerowych,
Wydział Elektroniki Telekomunikacji i Informatyki,
Politechnika Gdańska, Gdańsk, Polska
piotr.januszewski@pg.edu.pl

Streszczenie Rozdział ten rozważa rolę eksploracji w uczeniu się agentów sztucznej inteligencji przez wzmacnianie. Prezentuje przegląd współczesnych technik eksploracji i rozróżnia dwie główne rodziny technik: eksplorację nieukierunkowaną i eksplorację ukierunkowaną. Praca ta powinna pomóc zrozumieć dylemat pomiędzy eksploatacją wiedzy a eksploracją środowiska, któremu poddany jest agent w każdym kroku interakcji ze środowiskiem. Opisane tutaj techniki mogą być inspiracją do dalszych prac nad rozwiązaniem tego dylematu i tym samym ulepszenia metod uczenia się przez wzmacnianie.

Słowa kluczowe: sztuczna inteligencja, uczenie przez wzmacnianie, eksploracja

1 Wprowadzenie

Gdy inteligentny agent uczy się kontrolować nieznane środowisko, musi on pogodzić dwa przeciwstawne cele. Z jednej strony musi odpowiednio zbadać środowisko w celu zidentyfikowania optymalnego sposobu zachowania (ang. policy). Dla przykładu, robot zbierający puszki musi przemierzać nieznane środowiskiem, aby zdobyć wiedzę na temat miejsc gdzie zazwyczaj znajdują się puszki. Z drugiej jednak strony, agent musi eksploatować doświadczenie zdobyte podczas nauki. Chociaż robot musi badać swoje otoczenie, powinien unikać kolizji z przeszkodami po otrzymaniu negatywnej nagrody za kolizje. Ponadto, wiedza na temat miejsc w których wcześniej znalezione zostały puszki powinna być również brana pod uwagę przez robota przy wyborze działań. W celu efektywnego uczenia się, działania powinny być dokonywane w taki sposób, że środowisko jest eksplorowane przy jednoczesnym eksploatowaniu wiedzy agenta – działania powinny maksymalizować efekty uczenia się przy minimalizacji kosztów nauki. Ten podstawowy kompromis nazywany jest dylematem eksploracji i eksploatacji (ang. exploration and exploitation dilemma [18]). Ten rozdział bada rolę eksploracji w procesie uczenia się agenta w domenach ze skończoną przestrzenią akcji.

2 O eksploracji

Jakie są podstawowe pytania dotyczące roli eksploracji w uczeniu przez wzmocnianie? Zacznijmy od pytań charakteryzujących eksplorację i eksploatację. Jeśli eksploracja ma na celu skrócenie czasu nauki, to centralne pytanie dotyczące efektywnej eksploracji brzmi: "Jak czas uczenia się może zostać zminimalizowany?". W związku z tym pytanie o eksploatację brzmi: "Jak zminimalizować koszty nauki?". Te pytania są zwykle przeciwstawne, tj. im krótszy czas nauki, tym większe koszty i odwrotnie. Jak zobaczymy, czysta eksploracja również nie minimalizuje czasu nauki. Dzieje się tak, ponieważ czysta eksploracja maksymalizuje zdobywanie wiedzy, a tym samym może marnować dużo czasu na odkrywanie części środowiska nieistotnych dla zadania. Jeśli ktoś jest zainteresowany ograniczeniem eksploracji do odpowiednich części środowiska, często ma sens jednoczesne eksploatowanie zdobytej wiedzy. Dlatego eksploatacja jest częścią efektywnej eksploracji. Z drugiej strony, eksploracja jest również częścią efektywnej eksploatacji, ponieważ kosztów nie da się zminimalizować bez poznania środowiska.

Drugie ważne pytanie, które należy zadać, to: "Jaki wpływ ma dana zasada eksploracji na szybkość i koszt uczenia się?". Czyli, ile czasu powinien poświęcić projektant, który projektuje system uczenia się, na zaprojektowanie odpowiedniej reguły eksploracji? To pytanie będzie szeroko omawiane, ponieważ wpływ techniki eksploracji zarówno na czas nauki, jak i koszty nauki może być ogromny. W zależności od struktury środowiska, nieefektywna eksploracja może skutkować nawet eksponencjalnym wzrostem czasu potrzebnego na naukę w stosunku do wielkości problemu [13].

Trzecie kluczowe pytanie brzmi: „Jak odnaleźć kompromis pomiędzy eksploracją i eksploatacją?”. Ponieważ eksploracja i eksploatacja ustanawiają kompromis, to pytanie wymaga dalszej specyfikacja ograniczeń rozwiązania. Na przykład można zapytać: "Jak mogę znaleźć najlepszy kontroler w danym czasie?" lub "Jak mogę znaleźć najlepszy kontroler, nie przekraczając określonej kwoty kosztów?”. Oba pytania ograniczają dylemat w taki sposób, że optymalną kombinację między eksploracją a eksploatacją można znaleźć, biorąc pod uwagę, że problem można w ogóle rozwiązać przy tych ograniczeniach.
Załóżmy teraz, że ktoś posiadamy już sprawną technikę eksploracja i wydajną technikę eksploatacji. Rodzi to pytanie "Jak połączyć eksplorację i eksploatację?”. Czy każde działanie powinno jednocześnie eksplorować i eksploatować środowisko, czy też agent czasami bardziej powinien skupić się na eksploracji, a czasami bardziej na eksploatacji?

Na wszystkie te pytania nie można odpowiedzieć jedną, uniwersalną odpowiedzią. Kompromis pomiędzy eksploracją a eksploatacją, optymalna strategia eksploracji, a także właściwa technika łączenia eksploracji i eksploatacji w dużym stopniu zależą od konkretnego środowiska i zadania którego wykonywania agent ma się nauczyć. Zależą również od konkretnej techniki uczenia się. Niniejszy rozdział dotyczy zagadnień eksploracji, opisuje i ocenia techniki eksploracji w kontekście uczenia się przez wzmacnianie za pomocą metody Q-Learning. Omówione zostaną różne techniki eksploracji i porównane na podstawie wspólnej

taksonomii. Przeanalizowany zostanie wpływ eksploracji na złożoność uczenia się, pokazując, jak wybór reguły eksploracji może wpłynąć na czas i koszty nauki. Ponadto zbadamy kompromis między eksploracją a eksploatacją w zadaniu nawigacji robotem.

Reszta rozdziału jest zorganizowana w następujący sposób. W sekcji 3 przedstawimy pojęcia wykorzystywane w tym rozdziale. W sekcji 4 opisujemy taksonomię technik eksploracji i analizujemy wybrane techniki eksploracji często spotykane w najnowszej literaturze na temat uczenie się przez wzmacnianie. Sekcja 5 kończy ten rozdział dyskusją.

3 Uczenie przez wzmacnianie

Typowo problem który agent uczy się rozwiązywać modeluje się jako proces decyzyjny Markowa (ang. Markov decision process, MDP). MDP jest zdefiniowany przez krotkę $(\mathcal{S}, \mathcal{A}, R, P, \gamma, p_0)$, gdzie $\mathcal{S}$ jest ciągłą wielowymiarową przestrzenią stanów, $\mathcal{A}$ oznacza ciągłą wielowymiarową przestrzeń akcji, P jest modelem przejść, $\gamma \in [0, 1]$ oznacza współczynnik dyskontowy, p_0 odnosi się do dystrybucji stanu początkowego, a R jest funkcją nagrody. Agent uczy się z sekwencji przejść $\tau = [(s_t, a_t, r_t, s_{t+1}, d)]_{t=0}^{T}$, zwanych epizodami lub trajektoriami, gdzie $a_t \sim \pi(\cdot|s_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t)$, $r_t = R(s_t, a_t, s_{t+1})$, d jest sygnałem terminalnym, a T oznacza moment czasu kroku terminalnego. Stochastyczna polityka $\pi(a|s)$ odwzorowuje każdy stan na rozkład akcji. Deterministyczna polityka $\pi : \mathcal{S} \to \mathcal{A}$ przypisuje każdemu stanowi jedną akcję. Polityka π^* stanowi optymalne rozwiązanie problemu decyzyjnego Markowa.

Jednym ze sposobów rozwiązania tak opisanego problemu jest Q-Learning. Agent uczy się funkcji Q, która ocenia jakość każdej akcji w każdym stanie. Funkcja Q może być modelowana klasycznie tablicą [18] lub np. siecią neuronową [11]. Jakość akcji jest definiowana przez oczekiwaną sumę przyszłych dyskontowanych nagród jeśli wybierzemy daną akcję a w stanie s, a później agent będzie postępował optymalnie do końca epizodu:

$$Q(s, a) = \mathop{\mathbb{E}}_{\pi^*, P}\left[\sum_{t=0}^{T} \gamma^t r_t | s_0 = s, a_0 = a\right]$$

Politykę która maksymalnie eksploatuje zdobytą wiedzę funkcji Q otrzymuje się wybierając zawsze akcję o najwyższej jakości $\pi(s) = \text{argmax}_a\, Q(s, a)$. Zbieranie danych do treningu funkcji Q dokonywane jest inną polityką eksplorującą środowisko, patrz sekcja 4. Optymalne rozwiązanie π^* otrzymuje się wybierając akcję z pomocą wytrenowanej funkcji Q^*, oznaczonej gwiazdką ze względu na jej optymalność, $\pi^*(s) = \text{argmax}_a\, Q^*(s, a)$. Szczegóły trenowania funkcji Q zostały opisane w rozdziale 6.

4 Techniki eksploracji

W tym rozdziale skupimy się na politykach eksploracyjnych i rozróżnimy dwie główne rodziny technik eksploracji, eksplorację nieukierunkowaną i eksplora-

cję ukierunkowaną. Nieukierunkowane techniki eksploracji badają środowisko w oparciu o losowość. Najbardziej prymitywną techniką eksploracji nieukierunkowanej jest eksploracja losowa, w której działania są wybierane losowo za pomocą jednostajnego rozkładu prawdopodobieństwa, niezależnie od oczekiwanych kosztów. Na przykład robot korzystający z losowej eksploracji będzie raz po raz zderzał się z przeszkodami, niezależnie od wcześniejszych doświadczeń. Inne opisane tutaj techniki nieukierunkowanej eksploracji uwzględniają koszty poprzez modyfikację rozkładu prawdopodobieństwa, z jakim wybierane są działania. Stopniowo unika się kar, zmniejszając prawdopodobieństwo niewłaściwych działań. Eksploracja ukierunkowana różni się od eksploracji nieukierunkowanej tym, że wykorzystuje pewną specyficzną wiedzę do kierowania eksploracją. Zamiast wybierać działania losowo, wybiera się działania w taki sposób, aby zoptymalizować oczekiwany zysk wiedzy. Podstawową wiedzą którą można wykorzystać do kierowania eksploracją jest informacja jak dobrze poznane są konkretne części środowiska. Techniki eksploracji ukierunkowanej pozwalają zatem na bezpośredni wybór działań, które maksymalizują efekt eksploracji np. kierują agenta w najgorzej poznane regiony środowiska. Chociaż wyższość eksploracji ukierunkowanej została uznana przez kilku autorów [12, 5, 1] techniki nieskierowane są często spotykane w najnowszej literaturze [10, 7, 21].

4.1 Eksploracja nieukierunkowana

Techniki eksploracji nieukierunkowanej charakteryzują się wykorzystaniem losowości do eksploracji. Najbardziej podstawowa i pozbawiona informacji technika nieukierunkowanej eksploracji nazywa się *błądzeniem losowym*. Akcje są wybierane losowo z jednostajnego rozkładu prawdopodobieństwa. Błądzenie losowe w ogóle nie uwzględnia kosztów ani nagród. Ta technika wciąż jest wykorzystywana w najnowszej literaturze pod postacią polityki ϵ-greedy [11, 9]. Polityka ta z prawdopodobieństwem ϵ eksploruje wybierając jednostajnie losową akcję, a z prawdopodobieństwem $1-\epsilon$ eksploatuje swoją wiedzę wybierając akcję najlepszą według oceny aktualnej polityki. Zatem parametr ϵ kontroluje w jakim stopniu agent eksploruje środowisko, a w jakim stopniu eksploatuje swoją wiedzę.

Istnieją techniki eksploracji nieukierunkowanej, które uwzględniają koszty podczas nauki. Zwykle wykorzystują one wyuczoną dotychczas politykę do oceny oczekiwanych kosztów lub nagród wybieranych działań. Akcje są wybierane losowo, ale oczekiwane nagrody są wykorzystywane do modyfikowania prawdopodobieństwa z jakimi akcje są wybierane: im lepsza akcja, tym większe prawdopodobieństwo że zostanie wybrana i vice versa, gorsze akcje są częściej unikane. Jedną z technik takiej eksploracji jest strategia oparta na rozkładzie Boltzmanna nazywana polityką *softmax*.

Wiele algorytmów uczenia przez wzmacnianie oparte jest na metodzie iteracyjnego ulepszania polityki (ang. policy iteration) [18]. Q-Learning jest przykładem takiej metody. Q-Learning uczy się funkcji $Q(s, a)$, która pozwala na oszacowanie jakości – oczekiwanej sumy przyszłych nagród – każdego działania z osobna w konkretnym stanie. Na jej podstawie wybierana jest najlepsza akcja przez politykę zachłanną kiedy zależy nam na eksploatowaniu wiedzy. Zamiast czarnej

skrzynki która zwraca nam akcję do podjęcia, dostajemy narzędzie do oceny użyteczności działań. Polityka *softmax* wykorzystuje te szacunki i za pomocą dystrybucji Boltzmanna waży użyteczność eksploracyjną każdej akcji, biorąc pod uwagę oceny jakości wszystkich innych akcji:

$$\pi(a_i|s) = \frac{e^{Q(s,a_i)/T}}{\sum_j e^{Q(s,a_j)/T}}$$

Tutaj $\pi(a_i|s)$ oznacza prawdopodobieństwo wybrania akcji a_i w stanie s, a T jest dodatnią wartością stałą, często nazywaną temperaturą, która kontroluje w jakim stopniu wiedza agenta jest eksploatowana. Jeśli T jest duże, akcje są bardziej losowe, niezależnie od oceny funkcji Q. Jeśli T zbiega do zero, wtedy prawdopodobieństwo najlepszej akcji staje się coraz większe i zbiega do 1 – polityka staje się coraz bardziej zachłanna i bardziej eksploatuje wiedzę agenta. Kiedy zastanowimy się jakim działaniom polityka *softmax* przydziela wysokie prawdopodobieństwa, dojdziemy do wniosku iż im wyższa jakość akcji, $Q(s, a)$, w porównaniu do wszystkich pozostałych akcji, tym większe prawdopodobieństwo że zostanie ona wybrana.

Innym przykładem techniki eksploracji nieukierunkowanej, która uwzględnia koszty podczas nauki, jest *stochastyczna polityka* w algorytmach Actor-Critic np. Vanilla Policy Gradient [19]. Polityka ta wybiera akcje losowo (jest stochastyczna), ale jest trenowana za pomocą gradientów [22] w taki sposób, aby lepsze akcje pod względem oczekiwanych nagród były bardziej prawdopodobne, a te które powodują wysokie koszty były mniej prawdopodobne.

Podsumowując, nieskierowane techniki eksploracji wybierają działania stochastycznie, podczas gdy działania o lepszej jakości są wybierane z równym lub wyższym prawdopodobieństwem niż działania o gorszej jakości. Techniki eksploracji nieukierunkowanej obejmują szerokie spektrum stochastycznych mechanizmów selekcji działań, które monotonicznie odwzorowują użyteczność eksploatacyjną akcji na prawdopodobieństwa wyboru działań. Poszukiwania w oparciu o niejednostajne rozkłady prawdopodobieństwa, takie jak rozkłady Boltzmanna, nadal opierają się na losowości, ale częściowo unika się kosztownych akcji na rzecz tych które agent wie że przynoszą wysokie nagrody. Techniki te są często spotykane w domenach z dyskretną i skończoną przestrzenią akcji. Znacznie trudniej jest zastosować te metody w dziedzinach w których akcje przyjmują wartości rzeczywiste. W takim wypadku prym wiodą metody oparte na deterministycznych politykach do których dodaje się losowy szum eksploracyjny [10, 7, 21], chociaż są pewne wyjątki [8].

Techniki eksploracyjne oparte na modyfikowaniu rozkładu prawdopodobieństwa akcji uzyskują eksplorację jedynie poprzez losowość, a zatem eksploracja jest nieukierunkowana. Eksploracja nieukierunkowana może być nieefektywna pod względem czasu nauki, co oznacza, że oczekiwany czas nauki może skalować się wykładniczo z rozmiarem przestrzeni stanów [13].

4.2 Eksploracja ukierunkowana

Techniki eksploracji ukierunkowanej wykorzystują pewną wiedzę specyficzną dla eksploracji do prowadzenia poszukiwań w środowisku. Zamiast losowego wybierania akcji, schemat eksploracji bezpośrednio określa, którą akcję należy podjąć w następnej kolejności aby jak najlepiej poznać środowisko. Ostatecznym celem ukierunkowanej eksploracji jest wybranie działań, które maksymalizują poprawę agenta w czasie. Jest to jednak niemożliwe do ustalenia ponieważ nie można z góry wiedzieć, w jaki sposób akcja poprawi wydajność agenta w nieznanym lub częściowo nieznanym środowisku. Z tego powodu techniki ukierunkowanej eksploracji opisane w niniejszej sekcji mają charakter heurystyczny – wykorzystują heurystyki do optymalizacji zdobywania wiedzy. W szczególności eksplorację można osiągnąć poprzez wybór działań i/lub stanów, które były wybierane rzadziej [3], lub dawniej [1], lub zakłada się że mają wysoki błąd predykcji [17], lub wcześniej wykazywały wysoki błąd predykcji [16]. Wszystkie te metody można wspólnie nazwać optymistycznymi w obliczu niepewności, w skrócie OFU (od ang. optimistic in the face of uncertainty). Eksplorację ukierunkowaną można również uzyskać metodami Bayesowskimi, za pomocą próbkowania a'posteriori (ang. posterior sampling)[14].
Techniki ukierunkowanej eksploracji są zwykle bardziej efektywne niż dowolna technika nieukierunkowana zarówno pod względem czasu nauki, jak i kosztów nauki. [20] przedstawia twierdzenie i dowodzi wyższość ukierunkowanej eksploracji dla wielu skończonych dziedzin deterministycznych. Niemniej jednak, wiele problemów nie wymaga ukierunkowanej eksploracji do ich rozwiązywalnych – wystarczają prostsze techniki nieukierunkowane opisane w poprzedniej podsekcji. Poniżej opisujemy kilka przykładów technik ukierunkowanej eksploracji.

Eksploracja oparta na licznikach Ta technika eksploracji oparta się na adaptacyjnej mapie $N(s, a)$, zliczającej wykonania akcji a w każdym stanie s. Mapa ta służy do kierowania agenta do mniej zbadanych stanów. Prostym przykładem polityki wykorzystującej ten schemat eksploracji jest reguła źa każdym razem idź do najrzadziej odwiedzanego sąsiada". Bardziej wyrafinowana metoda eksploracji będzie łączyła zdobytą wiedzę z informacją o tym jak często poszczególne akcje były wybierane np. w postaci bonusu eksploracyjnego dodawanego do funkcji Q:

$$U(s, a) = Q(s, a) + c \cdot \frac{\sqrt{\sum_b N(s, b)}}{1 + N(s, a)}$$

Tutaj c jest stałą kontrolującą jak bardzo agent eksploatuje swoją wiedzę, inna dla każdego środowiska, $Q(s, a)$ to funkcja oceniająca jakość akcji, agent uczy się przybliżać tę funkcję, a $U(s, a)$ ocenia użyteczność eksploracyjną akcji a w stanie s. Agent wybiera akcję która maksymalizuje użyteczność eksploracyjną: $\text{argmax}_a\, U(s, a)$. Akcja zostanie wybrana jeśli dotychczas przynosiła duże nagrody lub była rzadko wybierana. Warto zauważyć, iż nawet jeśli akcja nie była ostatnio wybierana, ale jej jakość jest bardzo niska, wciąż nie zostanie wybrana.

Jeśli przestrzeń stanów i akcji jest zbyt duża, aby możliwe było wielokrotne odwiedzenie każdej pary stan-akcja, stosuje się metody przybliżające licznik

(ang. pseudo-count), które generalizują wartość licznika pomiędzy podobnymi parami stan-akcja [2].

Eksploracja poprzez optymistyczną inicjalizację W książce [18, s. 34-35] możemy znaleźć sposób prowadzenia ukierunkowanej eksploracji poprzez czystą eksploatację. Kluczowy pomysł opiera się na optymistycznym zainicjalizowaniu tablicy w Q-Learningu np. maksymalną możliwą sumą nagród do zdobycia. Wstępne przeszacowanie jakości każdej pary stan-akcja, $Q(s, a)$, sprawi iż agent będzie preferował niezbadane działań, a tym samym stany, ze względu na ich dużą ocenę jakości. W miarę uczenia się i estymaty jakości zbliżają się do rzeczywistych wartości i ukierunkowana eksploracja stopniowo się zmniejsza. Technika ta może być łączona z inną techniką eksploracyjną, aby utrzymać eksplorację.

Eksploracja oparta na oszacowaniu górnym Technika ta również opiera się na przeszacowaniu wartości funkcji Q. W tym przypadku, wartości jakości są wyrażone jako górna granica oczekiwanej sumy nagród, inaczej niż w Q-Learningu, gdzie funkcja Q ma przybliżyć średnią oczekiwaną sumę nagród. Wykorzystana statystyka zapewnia poprawność tego ograniczenia górnego z pewnym z góry określonym prawdopodobieństwem. Uzyskujemy więc maksymalną, ale wciąż prawdopodobną, jakość jaką może osiągnąć dana akcja na podstawie dotychczas zebranych danych. Statystyka ta jest po części funkcją liczby doświadczeń, a wartości jakości są stopniowo pomniejszane w miarę zdobywania wiedzy. Wynika to z faktu, iż im więcej doświadczeń, tym węższy staje się przedział ufności na wartości jakości akcji i tym mniejsze górne ograniczenie.

Przykładami algorytmów wykorzystujących tę technikę eksploracji są Upper Confidence Bound (UCB) [18] oraz Optimistic Actor-Critic (OAC) [5]. W pierwszym przypadku, wykorzystywany jest licznik odwiedzeń par stan-akcja do oszacowania górnego ograniczenia jakości akcji. W drugim natomiast autorzy trenują kilka sieci neuronowych przybliżających funkcję Q i na tej podstawie oszacowują ograniczenie górne jakości akcji (technika nazywana w statystyce "bootstrap").

Eksploracja oparta na pamięci W odróżnieniu od technik opartych na liczniku, techniki oparte na pamięci nie zliczają odwiedzeń stanów, natomiast zapisują ostatnio odwiedzone stany w buforze. Ponieważ stany mogą być duże, np. obrazki środowiska, metody te często trenują sieci neuronowe, lub wykorzystują inną metodę przetwarzania np. skalowanie obrazka, do kodowania stanów w mniejszej przestrzeni. Pozwala to również generalizować pomiędzy podobnymi stanami. Następnie, tak utworzony bufor odwiedzonych stanów, może być wykorzystywany do kierowania agenta w rejony środowiska dotychczas rzadziej odwiedzane [1] lub prowadzenia agenta z powrotem do stanów w których otrzymał on wcześniej wysokie nagrody i rozpoczęcie eksploracji z tamtego miejsca [6].

Eksploracja oparta na błędzie predykcji Podobnie jak liczniki, błąd predykcji również może zostać wykorzystany do kierowania agenta w stronę niepoznanych dotychczas rejonów środowiska. Niezależnie od wybranego modelu, jego

wysoki błąd predykcji oznacza, że agent wcześniej nie miał szansy nauczyć się jak reagować w danej sytuacji. Jednym ze sposobów implementacji tej techniki jest uczenie dedykowanej sieci neuronowej, $\hat{f}(s)$, przewidywać wyjścia bliźniaczej sieci, $f(s)$, ale inaczej zainicjalizowanej. Obie sieci jako wejście dostają stan środowiska s. Technika ta nazywa się destylacją wiedzy losowej sieci neuronowej (ang. Random Network Distillation) [4], ponieważ w miarę treningu, destylujemy predykcje losowej sieci do sieci trenowanej. Trenowana sieć będzie lepiej przewidywała wyjścia z losowej sieci, czyli będzie miała niższy błąd predykcji, w stanach które były często odwiedzane, ponieważ miała wiele okazji do nauczenia się wyjść sieci losowej. Natomiast błąd predykcji będzie wysoki, w nowych, niespotykanych wcześniej stanach. W ten sposób uzyskujemy nowy, "wewnętrzny sygnał uczący $r_i(s) = \left\| \hat{f}(s) - f(s) \right\|^2$. Ten wewnętrzny sygnał jest mieszany z nagrodą środowiska, sygnałem zewnętrznym $r_e(s)$, dla zaobserwowanego stanu i agent jest uczony na podstawie tego sygnału $r(s) = r_e(s) + c \cdot r_i(s)$. Stała c waży stosunek sygnału eksploracyjnego, r_i, do sygnału eksploatacyjnego, r_e, i pozwala kontrolować agresywność eksploracji agenta. W miarę treningu, eksploracja będzie wygaszana, ponieważ trenowana sieć będzie lepiej przewidywać wyjścia sieci losowej.

Eksploracja oparta na próbkowaniu a'posteriori W artykule [12], autorzy wykorzystują statystyczną metodę bootstrap w jeszcze inny sposób. Autorzy trenują kilka sieci neuronowych przybliżających funkcję Q i wybierają jedną z nich na pewien okres, zazwyczaj jednego epizodu środowiska, do wybierania akcji. Ponieważ sieć neuronowa zwróci jakąś wartość dla każdej pary stan-akcja, nawet takiej której nigdy wcześniej nie zaobserwowała, to różne inicjalizacje sieci neuronowych sprawią, iż sieci będę miały różne uprzedzenia dotyczące przestrzeni stanów i akcji – będą wstępnie optymistyczne lub pesymistyczne w stosunku do różnych rejonów środowiska. To natomiast sprawi, że agent będę kierował się w różne rejony środowiska, w zależności która sieć jest aktualnie wybrana do zbierania danych. Przypomina do metodę optymistycznej inicjalizacji z wcześniejszego paragrafu, ale zastosowaną do sieci neuronowych. W miarę postępu treningu, sieci lepiej przewidują rzeczywiste jakości akcji i ich uprzedzenia stają się mniej istotne przy wyborze akcji. Metoda ta osiąga dobre empiryczne rezultaty, a autorzy w innym artykule przedstawiają teoretyczne dowody na optymalność tej techniki eksploracji [15].

5 Dyskusja

Celem tego rozdziału było zilustrowanie podstawowych heurystyk używanych do efektywnej eksploracji. Opisując niektóre z najbardziej popularnych heurystyk, rozróżniliśmy dwie rodziny technik eksploracji: nieukierunkowane i ukierunkowane. Zajęliśmy się fundamentalnym kompromisem między eksploracją a eksploatacją i przedstawiliśmy parametry kontrolowania tego kompromisu. Jednakże, istnieje kilka ograniczeń przyjętego podejścia, z którymi należy się zmierzyć przy

zastosowaniu przedstawionych w tym rozdziale idei do bardziej złożonych dziedzin.

Złożone domeny Wszystkie przedstawione schematy ukierunkowanej eksploracji mają na celu zbadanie całej przestrzeni stanów i akcji, zakładając, że możliwe jest wyczerpujące zbadanie tej przestrzeni lub przynajmniej części tej przestrzeni. W wielu problemach jest to rozsądne założenie, ale istnieje wiele problemów, w tym te typowo badane w kontekście rzeczywistych zastosowań sztucznej inteligencji, w których przestrzenie stanów i akcji są zbyt duże, aby można je było wyczerpująco zbadać. W takich problemach, zamiast szukać najlepszego rozwiązania, celem jest często skuteczne znalezienie dobrego, nieoptymalnego rozwiązania. W takich przypadkach inteligentny agent musi odróżnić istotne i nieistotne części problemu i odciąć eksplorację w nieistotnych regionach środowiska. Do pewnego stopnia połączenie eksploracji i eksploatacji może ograniczać eksplorację nieistotnych części problemu, ponieważ eksploatacja sprawia, iż agent koncentruje się na bardziej lukratywnych regionach środowiska. Wykluczanie nieistotnych części problemu można również uzyskać, dostarczając wiedzę zewnętrzną do prowadzenia eksploracji, która może pomagają ocenić użyteczność eksploracyjną działań bez faktycznego ich doświadczania.

Wiedza domenowa W wielu podejściach do uczenia się, dostępna jest pewna wiedza domenowa. Ta wiedza może być reprezentowana za pomocą: inicjalizacji polityki, ograniczenia na akcje i/lub stany, wiedzę domenową w postaci reguł środowiska, wcześniej podanych strategii eksploracji i tak dalej. Wiedza domenowa może drastycznie zmniejszyć złożoność uczenia się. Rozdział ten, nie brał pod uwagę roli eksploracji, kiedy dostępna jest dodatkowa wiedza o problemie. Nie mniej jednak, ukierunkowane techniki powinny nadal się sprawdzić, zwłaszcza przy wykorzystaniu tejże wiedzy domenowej.

Eksploracja i generalizacja Optymalna eksploracja mocno zależy od wyboru uczonego modelu (np. sieci neuronowej) i możliwości generalizacji tego modelu. Generalizacja przenosi wiedzę z pojedynczego przykładu do zwykle nieskończonego zbioru powiązanych sytuacji. Wpływ generalizacji na optymalną eksplorację jest oczywisty. Na przykład, jeśli agent ma wybór między zupełnie nowym działaniem, a drugim działaniem podobnym do niektórych działań wybrany wcześniej, powinien spróbować nowej akcji, aby maksymalizować zdobywanie wiedzy, chociaż oba działania mogły nie być nigdy wcześniej testowane. Optymalna technika eksploracji musi uwzględniać, w jaki sposób model ekstrapoluje wiedzę z przykładów w celu oceny użyteczności eksploracyjnej każdej z akcji.

Literatura

1. Badia, A.P., Sprechmann, P., Vitvitskyi, A., Guo, D., Piot, B., Kapturowski, S., Tieleman, O., Arjovsky, M., Pritzel, A., Bolt, A., Blundell, C.: Never give up: Learning directed exploration strategies. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=Sye57xStvB

2. Bellemare, M.G., Srinivasan, S., Ostrovski, G., Schaul, T., Saxton, D., Munos, R.: Unifying count-based exploration and intrinsic motivation. In: Lee, D.D., Sugiyama, M., von Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. pp. 1471–1479 (2016), https://proceedings.neurips.cc/paper/2016/hash/afda332245e2af431fb7b672a68b659d-Abstract.html
3. Brafman, R.I., Tennenholtz, M.: R-MAX - A general polynomial time algorithm for near-optimal reinforcement learning. The Journal of Machine Learning Research **3**, 213–231 (2002), http://jmlr.org/papers/v3/brafman02a.html
4. Burda, Y., Edwards, H., Storkey, A., Klimov, O.: Exploration by random network distillation. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=H1lJJnR5Ym
5. Ciosek, K., Loftin, R., Vuong, Q., Hofmann, K.: Better exploration with optimistic actor-critic. In: Advances in Neural Information Processing Systems (2019)
6. Ecoffet, A., Huizinga, J., Lehman, J., Stanley, K.O., Clune, J.: First return, then explore. Nature **590**(7847), 580–586 (2021). https://doi.org/10.1038/s41586-020-03157-9, https://doi.org/10.1038/s41586-020-03157-9
7. Fujimoto, S., van Hoof, H., Meger, D.: Addressing function approximation error in actor-critic methods. In: Dy, J.G., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018. Proceedings of Machine Learning Research, vol. 80, pp. 1582–1591. PMLR (2018), http://proceedings.mlr.press/v80/fujimoto18a.html
8. Haarnoja, T., Zhou, A., Hartikainen, K., Tucker, G., Ha, S., Tan, J., Kumar, V., Zhu, H., Gupta, A., Abbeel, P., Levine, S.: Soft actor-critic algorithms and applications. CoRR **abs/1812.05905** (2018), http://arxiv.org/abs/1812.05905
9. Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., Horgan, D., Piot, B., Azar, M., Silver, D.: Rainbow: Combining improvements in deep reinforcement learning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 32 (2018)
10. Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D.: Continuous control with deep reinforcement learning. In: Bengio, Y., LeCun, Y. (eds.) 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings (2016), http://arxiv.org/abs/1509.02971
11. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529–533 (2015). https://doi.org/10.1038/nature14236, https://doi.org/10.1038/nature14236
12. Osband, I., Blundell, C., Pritzel, A., Van Roy, B.: Deep exploration via bootstrapped DQN. In: Advances in Neural Information Processing Systems (2016)
13. Osband, I., Roy, B.V., Russo, D.J., Wen, Z.: Deep exploration via randomized value functions. J. Mach. Learn. Res. **20**, 124:1–124:62 (2019), http://jmlr.org/papers/v20/18-339.html
14. Osband, I., Van Roy, B.: Why is posterior sampling better than optimism for reinforcement learning? In: 34th International Conference on Machine Learning, ICML 2017 (2017)

15. Osband, I., Van Roy, B.: Why is posterior sampling better than optimism for reinforcement learning? In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 2701–2710. PMLR (06–11 Aug 2017), http://proceedings.mlr.press/v70/osband17a.html
16. Schmidhuber, J.: Adaptive confidence and adaptive curiosity. Tech. rep., Institut fur Informatik, Technische Universitat Munchen, Arcisstr. 21, 800 Munchen 2 (1991)
17. Simmons-Edler, R., Eisner, B., Yang, D., Bisulco, A., Mitchell, E., Seung, H.S., Lee, D.D.: Reward prediction error as an exploration objective in deep RL. In: Bessiere, C. (ed.) Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020. pp. 2816–2823. ijcai.org (2020). https://doi.org/10.24963/ijcai.2020/390, https://doi.org/10.24963/ijcai.2020/390
18. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MIT press (2018)
19. Sutton, R.S., McAllester, D., Singh, S., Mansour, Y.: Policy gradient methods for reinforcement learning with function approximation. In: Proceedings of the 12th International Conference on Neural Information Processing Systems. p. 1057–1063. NIPS'99, MIT Press, Cambridge, MA, USA (1999)
20. Thrun, S.: The role of exploration in learning control. In: White, D., Sofge, D. (eds.) Handbook for Intelligent Control: Neural, Fuzzy and Adaptive Approaches. Van Nostrand Reinhold, Florence, Kentucky 41022 (1992)
21. Wang, C., Wu, Y., Vuong, Q., Ross, K.: Striving for simplicity and performance in off-policy DRL: Output normalization and non-uniform sampling. Proceedings of the 37th International Conference on Machine Learning **119**, 10070–10080 (13–18 Jul 2020), http://proceedings.mlr.press/v119/wang20x.html
22. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine Learning **8**(3), 229–256 (1992). https://doi.org/10.1007/BF00992696, https://doi.org/10.1007/BF00992696