JOURNAL
COMPUTER SCIENCES INSTITUTE

JCSI 27 (2023) 162–170
Received: 2 April 2023
Accepted: 9 May 2023

Influence of video content type on the usefulness of reinforcement learning algorithms in DASH systems

Wpływ typu treści wideo na przydatności algorytmów uczenia ze wzmocnieniem w systemach DASH

Przemysław Grzegorz Markiewicz*, Sławomir Wojciech Przyłucki

Department of Computer Science, Lublin University of Technology, Nadbystrzycka 36B, 20-618 Lublin, Poland

Abstract

The article presents the result of research on DASH (Dynamic Adaptive Streaming over HTTP) systems. In the proposed solution, the adaptive algorithm is based on the RL (Reinforcement Learning) paradigm. The Pensieve algorithm was chosen as the basis for the tests. This algorithm is widely discussed in the scientific literature and therefore the study and analysis of its properties is useful in a wide range of solutions using DASH. The main contribution of the presented test results to the development of knowledge on video streaming services consists in the analysis of the impact of the characteristics of video materials on the effectiveness of the adaptation process implemented by the developed RL model. The presented results show that this influence should not be omitted in any in-depth analyses of the characteristics of DASH systems.

Keywords: DASH; reinforcement learning; video streaming; QoE

Streszczenie

Artykuł przedstawia wynik badań nad systemami adaptacyjnego strumieniowania DASH (ang. Dynamic Adaptive Streaming over HTTP). W zaproponowanym rozwiązaniu algorytm adaptacyjny oparty jest na paradygmacie uczenia ze wzmocnieniem RL (ang. Reinforcement Learning). Jako podstawę do przeprowadzonych testów wybrany został algorytm Pensieve. Algorytm ten jest szeroko omawiany w literaturze naukowej i dlatego badanie i analiza jego własności jest przydatna w szerokiej gamie rozwiązań wykorzystujących DASH. Główny wkład zaprezentowanych wyników testów w rozwój wiedzy nad usługami strumieniowej transmisji wideo polega na analizie wpływu cech charakterystycznych materiałów wideo na efektywność procesu adaptacji realizowanego przez opracowany model RL. Przedstawione wyniki świadczą o tym, że wpływ zmienności treści wideo nie powinien być pomijany w jakichkolwiek pogłębionych analizach cech systemów DASH.

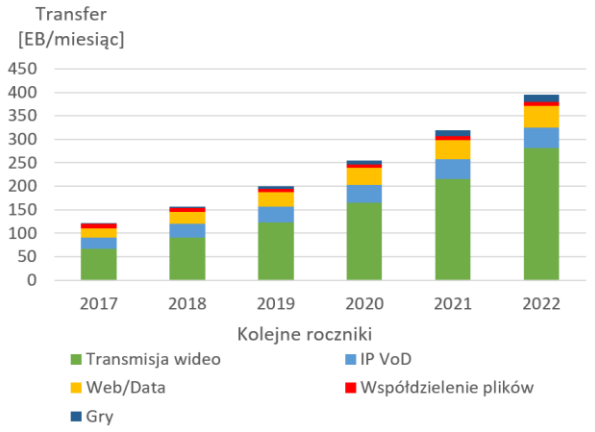
Słowa kluczowe: DASH; uczenie ze wzmocnieniem; transmisja wideo; QoE

*Corresponding author

Email address: **przemyslaw.markiewicz@pollub.edu.pl** (P. G. Markiewicz)

1. Wstęp

Od wielu lat ruch sieciowy [1] na całym świecie jest zdominowany przez platformy transmitujące wideo. Większość z nich wywodzi się z branży rozrywkowej a wśród nich są tacy potentaci jak Netflix, Apple TV, Youtube, Twitch i wiele innych. Dostarczenie dobrej jakości wideo zachowując płynność odtwarzania jest jednym z najbardziej krytycznych problemów technicznych. Popyt i oczekiwania na wysoką jakość wideo wciąż rosną, napędzane przez coraz większe rozdzielczości interfejsów urządzeń mobilnych, czy innego rodzaju odbiorników wideo. Geneza badanego zagadnienia jest złożona, stanowi konsolidację kilku problemów. Dostawcy muszą zapewnić wysoką dostępność usługi, a klienci dostosowują jakość wideo do bieżących parametrów łącza. Ze względu na losowy charakter ruchu sieciowego, możliwość gwarantowania wymaganych wartości przepustowości nie zawsze jest proste. Kolejnym aspektem jest poziom satysfakcji odbiorcy oglądającego transmisję. Użytkownik z założenia ma sprzeczne oczekiwania co do odtwarzanego wideo. Z jednej strony oczekuje on dostępu do treści wideo w możliwie najwyższe jakości wideo, a z drugiej oczekuje zachowania dużej płynności transmisji. Jeżeli dostępna przepustowość wykorzystywanego łącza nie pozwala by w danym momencie odtworzyć wideo najwyższej jakości, może wystąpić konieczność buforowania strumienia danych. W takim przypadku, aplikacja klienta zatrzymuje odtwarzanie wideo i czeka aż bufor zapełni się danymi, po czym wznawia odtwarzanie. Buforowanie wstrzymuje odtwarzanie, więc zakłóca płynność transmisji. Te zjawiska ilościowo opisuje współczynnik QoE (ang. Quality of Experience) [2]. Im wyższa wartość tego współczynnika tym wyższa satysfakcja z odbieranej treści, co oznacza jak najmniej zdarzeń buforowania i wysoką płynność transmisji. By spełnić wymienione wyżej oczekiwania użytkowników, powstały dedykowane rozwiązania transmisji strumieniowej wideo, takie jak DASH (ang. Dynamic Adaptive Streaming over HTTP), czy HLS (ang. HTTP Live Streaming) [3]. Najnowsze implementacje klienta DASH wykorzystują uczenie ze wzmocnieniem. W tej pracy badany jest wpływ parametrów strumieniowanej treści wideo na metrykę QoE w tego typu rozwiązaniach oraz analiza w jakim kierunku należałoby poszerzyć analizę.



Rysunek 1: Estymacja transferu danych w EB/miesiąc w podziale na różne typy danych/aplikacje [1].

1.1. Adaptacyjne strumieniowanie wideo

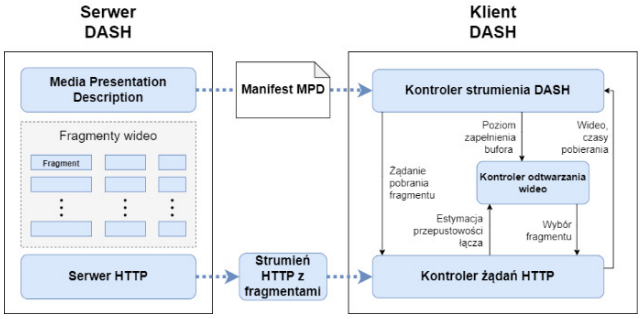
Elementy składowe systemu opartego o ten standard i ich wzajemne relacje są przedstawione na rysunku 2. Standard DASH, w najbardziej podstawowym założeniu co do sposobu komunikacji, jest w pełni oparty o protokół HTTP. W podstawowych założeniach funkcjonalnych system DASH umożliwia dynamiczne dostosowanie wyświetlanych treści adekwatnie do możliwości medialnych urządzenia, wydajności łącza oraz preferencji użytkownika. Klient DASH, próbując odtworzyć wideo, wysyła pierwsze żądania do serwera DASH. DASH w standardzie wymaga od serwera dostarczenia dla klienta definicji metadanych wideo niezbędnych do przeprowadzenia transmisji wideo. Metadane odnoszące się do treści i formatu wideo są umieszczone w pliku MPD (ang. Media Presentation Description) [4]. Jest to plik zawierający tzw. Manifest w formacie XML (ang. Extensible Markup Language). W pliku MPD [5] opisane są informacje o kodowaniu fragmentów wideo, ich rozmiarze, dostępnych jakościach. Dla każdego wideo odtwarzanego w systemie DASH jest wymagana definicja treści wideo zawarta w MPD. Odpowiedzialność za śledzenie stanu transmisji i za decyzję jaki fragment wideo pobrać jako następny oraz jakiej jakości ma być ten fragment, spoczywa na kliencie. Klient odpowiedzialny za dobór odpowiedniej jakości wykorzystuje zaimplementowany algorytm adaptacji ABR (ang. Adaptive Bitrate Streaming). Algorytm ten to zestaw instrukcji, w oparciu o które dobierana jest jakość kolejnego fragmentu wideo w zależności od parametrów łącza czy innych danych wejściowych. Istnieją czynniki zewnętrzne, które mogą zakłócić transmisję. Najbardziej istotne to [6]:

- niedostateczna wartość dostępnej przepustowości łącza internetowego,
- zbyt długi czas odpowiedzi serwera – opóźnienie.

Dostępna przepływność wpływa na to ile czasu zajmie pobranie konkretnego fragmentu wideo. Czas odpowiedzi serwera ma szczególnie istotne znaczenie w przypadku transmisji na żywo. Najważniejszym założeniem algorytmów ABR jest zapewnienie jak najlepszej jakości i płynności transmisji w ramach dostępnych zasobów sieciowych i sprzętowych urządzenia klienta. Zdarzenia negatywnie wpływające na odbiór transmitowanej treści to [2]:

- buforowanie – zatrzymanie transmisji spowodowane tym, że klient nie dysponuje w danym momencie fragmentem wideo w buforze i jest w trakcie jego pobierania,
- częste przełączanie jakości na przestrzeni kolejnych odtwarzanych fragmentów wideo w transmisji.

Zakłócenia są wypadkową parametrów sieciowych transmisji jak i sterowania doborem fragmentów wideo przez algorytm ABR.



Rysunek 2: Diagram działania systemu DASH [34].

1.2. Kierunki badań nad systemami DASH

Pierwsze badania [7] adaptacji strumienia w standardzie DASH oparły się na obserwacji stanu zapełnienia bufora BBA (ang. Buffer Based Approach). Zarządzanie bufora tak by zminimalizować zdarzenia buforowania transmisji wideo zapewnia płynność odtwarzania przez czas całej transmisji. Priorytetem algorytmu było początkowe zapełnienie bufora fragmentami niższej jakości, tak by zapewnić ciągłość w przypadku spadku przepustowości łącza i stopniowe zwiększanie jakości pobieranych kolejnych fragmentów treści wideo. W ten sposób możliwe było amortyzowanie wszelkich wariacji w przepustowości łącza. Jednakże kolejne badania wykazały, że omawiany algorytm jest rozwiązaniem suboptymalnym, ponieważ jego responsywność na dostępną wysoką przepustowość łącza była bardzo opóźniona, co uniemożliwiało poprawienie jakości. Alternatywne rozwiązania jak RB [8] (ang. Rate Based) wykorzystujące obserwację dostępnej przepływności łącza również nie dały zadowalających rezultatów, przyczyniły się do dużej fluktuacji w transmisji wideo co znacznie pogorszyły płynność. Kolejnym krokiem w rozwoju było utworzenie hybrydowych rozwiązań jak [8] MPC (ang. Model Predictive Control) wykorzystujące podejście kontrolowania strumieniem w oparciu o predykcję dwóch wymienionych wcześniej zmiennych (przepływność łącza, stan zapełnienia bufora). Dodatkowo w dalszych badaniach nad rozwojem wprowadzono pojęcie metryki QoE, która umożliwiła obiektywną ocenę algorytmów ABR w oparciu o zmienne, które mają rzeczywiste odzwierciedlenie w odczuciach odbiorcy końcowego strumieniowych usług wideo. MPC realizuje proces predykcji, w którym szacuje się wybór najbardziej opłacalnej przepływności fragmentu w perspektywie metryki QoE. Umożliwiło to uzyskanie o wiele lepszych rezultatów w stosunku do klasycznych

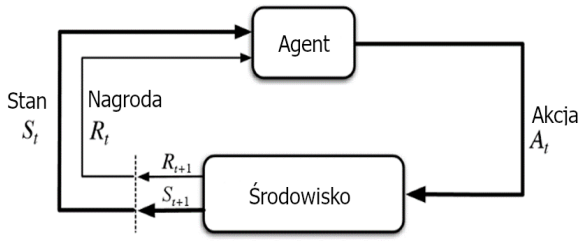
algorytmów (BBA, RB). Jednocześnie metoda oceny metryką QoE stała się podwaliną w opracowaniu kolejnych rozwiązań. Inne rozwiązania jak BOLA-E, czy DYNAMIC [9] również wykorzystują tą metodę.

2. Uczenie ze wzmocnieniem

Uczenie ze wzmocnieniem jest jedną z 3 podstawowych technik uczenia. Większość środowisk uczenia ze wzmocnieniem jest zdefiniowana w postaci procesu decyzyjnego Markov'a MDP (ang. Markov Decision Process), który jest rozbudowaną koncepcją równania Bellmana. Podstawowymi elementami w stochastycznym modelu MDP są [10]:

- środowisko (ang. Environment) – jest to odizolowana skończona przestrzeń w ramach, której agent operuje,
- agent (ang. Agent) – jest to algorytm odpowiedzialny za podejmowanie działań w celu zrealizowania narzuconego zadania/wytycznych,
- cel – są to wytyczne/wytyczna, która mają być finalnie zrealizowane przez agenta,
- stan (ang. State) – agent może przyjąć określony stan po podjęciu określonej akcji, są to etapy przejściowe na drodze do realizacji celu,
- akcja (ang. Action) – jest to działanie/zestaw działań, które może podjąć agent w celu przejścia z jednego stanu do kolejnego. Gama możliwych działań jest narzucona przez dostępne stopnie swobody w środowisku,
- nagroda (ang. Reward) – agent za podjęcie akcji i przejście do kolejnego stanu otrzymuje odpowiednią nagrodę.

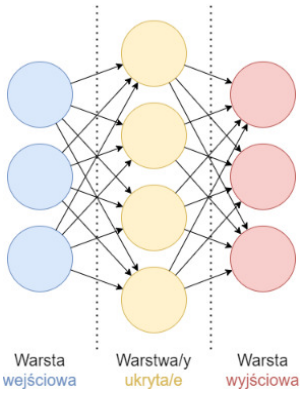
Agent w środowisku, podejmując akcje przechodząc z jednego stanu do kolejnego, otrzymuje nagrodę. Nagroda jest przydzielana adekwatnie do tego czy uzyskany stan przez agenta przybliża go do końcowego celu, czy oddala. W dużym stopniu to w jaki sposób jest przydzielona nagroda jest silnie uzależnione od warunków środowiska jak i implementacji agenta. Każda akcja podjęta przez agenta, która wiąże się z przejściem z pierwotnego stanu do kolejnego wybranego stanu, opisana może być z wykorzystaniem pojęcia prawdopodobieństwa. Każdy stan prowadzący coraz bliżej do zrealizowania celu jest wyżej nagradzany. W ten sposób agent wie jaki jest priorytet w realizacji celu. Tą metodą można znaleźć najlepszą sekwencję akcji (politykę). Znalezienie optymalnej polityki w środowisku o skończonej ilości stanów jest kluczowym celem.



Rysunek 3: Interakcja elementów składowych w modelu uczenia ze wzmocnieniem [11], gdzie t stanowi krok w ciągłej iteracji zmiany stanów.

2.1. Algorytm Pensieve

Algorytm Pensieve został zaproponowany w pracy [12]. Jego fundamentem jest metoda A3C (ang. Asynchronous Advantage Actor-Critic). Jest to asynchroniczny wariant metody A2C (ang. Advantage Actor-Critic) [13, 14]. Podstawowa metoda AC (ang. Actor-Critic) obejmuje założenia co do formy samego agenta i metody maksymalizacji wynagrodzenia. W procesie AC funkcja agenta podejmujące akcje są podzielone na dwie sieci neuronowe. Sieć neuronowa jest to zagadnienie silnie powiązane z głębokim uczeniem [15]. W dużym uogólnieniu sztuczne sieci neuronowe to systemy, które mogą „nauczyć się" rozpoznawać wzorce i relacje między danymi. Nazywa się to siecią neuronową, ponieważ jest inspirowana strukturą i funkcją ludzkiego mózgu. Proste sieci są to modele składające się typowo z 3 warstw powiązanych węzłów/neuronów. Pierwsza warstwa jest wejściowa, a ostatnia pełni rolę warstwy wyjściowej. Poszczególne węzły sieci (sztuczne neurony) operują na wartościach skalarnych i każdy z węzłów ma przypisaną wagę, odpowiedzialną za jego względną istotność w procesie przetwarzania danych w sieci [10]. Agent zawiera sieć neuronową aktora i sieć neuronową krytyka. Sieć neuronowa aktora jest odpowiedzialna za przyjmowanie parametrów wejściowych z środowiska i podejmowanie akcji. Każde podjęte działanie aktora jest oceniane przez sieć krytyka, a jego ocena przekazana jest dalej do aktora jako informacja zwrotna. Proces uczenia nie następuje na końcu każdej iteracji jak przy polityce gradientowej, a już w trakcie danego przebiegu. Dzięki temu również możliwe jest wykorzystanie tej techniki w środowiskach ciągłych i wymaga ona mniejszej liczby iteracji w celu wytrenowania polityki. W przypadku gradientu polityk mamy również do czynienia z dużą wariancją w rezultatach generowanych polityk, co jest również powiązane z ilością iteracji potrzebnych do osiągnięcia konwergencji. Natomiast metoda AC stanowi rodzinę algorytmów, stąd A2C funkcjonuje z tymi samymi założeniami, a różnica wynika w funkcji estymacji sieci krytyka. Tak samo wariant A3C jest rozbudowanym A2C, gdzie środowiska agenta są powielone w procesie uczenia i operują na wspólnym zestawie parametrów [14].



Rysunek 4: Poglądowa struktura głębokiej sieci neuronowej [11].

2.2. Uczenie ze wzmocnieniem w systemach DASH

Obecnie uczenie wzmocnieniem jest dominującym zagadnieniem i kierunkiem badań w celu usprawnienia adaptacji strumienia wideo. Przykładem takich rozwiązań jest system DeepLive [16] do generowania algorytmu ABR wykorzystujący metodę optymalizacji wartości metryki QoE. Jednakże DeepLive różni się od MPC nie tylko samą implementacją, a dodatkowo przeznaczeniem rozwiązania w systemach DASH i inaczej sformułowaną definicją QoE. DeepLive jest zaprojektowany z myślą o platformach z transmisjami na żywo. Oznacza to, że sposób fragmentacji wideo i priorytety odtwarzania transmisji różnią się np. od standardowego wideo na platformie Netflix. Dodatkowym usprawnieniem jest mechanizm omijania klatek. Oznacza to, że jeżeli czas odpowiedzi przekroczy narzucony, dopuszczalny limit to pomijane są klatki wideo by zredukować opóźnienie i pobierane są kolejne. W implementacji Deeplive wykorzystany jest algorytm Double-DQN, czyli złożenie dwóch głębokich sieci neuronowych. Wprowadzono też usprawnienia takie jak użycie algorytmu typu RB na początku wyświetlania transmisji, czy w momencie przewinięcia wideo do określonego miejsca. Okazało się one korzystniejsze ponieważ algorytm Pensieve przyjmował domyślne ustawienia, które okazywały się nieadekwatne do bieżącego stanu systemu. Opracowane rozwiązanie zostało zestawione w testach z innymi algorytmami z grup BBA, RB oraz referencyjnym algorytmem w dash.js DYNAMIC i podobnym rozwiązaniem Pensieve. Z testów zrealizowanych dla różnych symulowanych warunków sieciowych wynika, że Deeplive pod każdym względem zyskuje po około 16% w stosunku do najlepszego ówcześnie rozwiązania jakim był Pensieve.

W innych publikacjach [17] badane jest również zaganienie rywalizacji o dostęp do zasobów. Tradycyjnie główną odpowiedzialność za optymalizację metryki QoE w systemie DASH ponosi klient. Poszerzyło to założenia co do optymalizacji transmisji dla wielu klientów współdzielących to samo łącze internetowe. Rozwiązania klientów korzystających z uczenia ze wzmocnieniem np. DeepLive, czy Pensieve, w swojej ograniczają się do samolubnej alokacji łącza w celu maksymalizacji metryki QoE. W przypadku wielu klientów wykorzystujących powyższe rozwiązania może to spowodować, że klienci będę rywalizować co negatywnie wpłynie na jakość odbieranych transmisji (na wartości metryki QoE każdego klienta) i spowoduje niezadowolenie użytkowników dostrzegających brak płynności w wyświetlanej treści. Rozwiązanie FBAC [12] korzysta z pomocy serwera sterując przepływnością łącza dla klientów. Analogicznie jak na potrzeby klienta w politykach ABR wygenerowanych metodami uczenia ze wzmocnieniem korzysta się z mechanizmów optymalizacji metryki QoE, tak i dla serwera opracowana została nowa metryka optymalizacji QoE. Natomiast do realizacji kontroli przepływności łącz klientów wykorzystany został algorytm generowany metodą DRL (ang. Deep Reinforcement Learning). Nowo zaproponowana metryka QoE jest nazwana metryką uczciwości, czyli pomaga ustalić na ile sprawiedliwie jest przydzielany dostęp do przepływności łącza w kontekście maksymalizacji wartości metryk QoE uzyskiwanych u poszczególnych klientów. FBAC korzysta z mechanizmu kontrolowania przeciążeń. W FBAC również wykorzystana jest metoda uczenia ze wzmocnieniem, ale w tym wypadku autorzy zdecydowali się na zastosowanie metody opartej na niezależnej proksymalnej polityce optymalizacji IPPO (ang. Independent Proximal Policy Optimization). Przedstawione rozwiązanie systemowe zostało zweryfikowane w środowisku testowym. Środowisko to obejmowało dedykowany serwer i wielu klientów wykorzystujących algorytm MPC, dla których serwer przeliczał swoją globalną wartość metryki w oparciu o wartości metryk QoE uzyskane od klientów. Mechanizm sterujący przeciążeniami został oparty o mechanizm QUIC (ang. Quick UDP Internet Connections) zaimplementowany w serwerze DASH. Zrealizowano serię testów zestawiając ze sobą różne treści wideo w różnych jakościach dostosowane do standardu DASH. Scenariusze testowe obejmowały wiele klientów z różnymi implementacjami polityki ABR: MPC, Pensieve, BOLA, FBAC z serwerem Nginx i protokołem QUIC. Wyniki tych testów wykazały, że metryka uczciwości jak i średnia wartość metryk QoE klientów w przypadku FBAC była wyższa o 9-13% w stosunku do pozostałych polityk ABR. To oznacza, że odgórne sterowanie łączem, tak by zoptymalizować metrykę uczciwości, pozytywnie wpłynęło na wartości metryki QoE po stronie klientów.

3. Metodyka badań

Opracowane środowisko testowe, do weryfikacji własności algorytmów ABR generowanych w oparciu o uczenie ze wzmocnieniem, składa się z dwóch podstawowych modułów. Pierwszy z nich to moduł odpowiedzialny za proces przygotowania modelu, natomiast drugi pozwala na przeprowadzenie symulacji wykorzystujących wytrenowany model i akwizycję danych pomiarowych. Korzystając z doświadczeń innych zespołów badawczych [12, 18] zostało zaprojektowane oraz wdrożone środowisko testowe o opisanej poniżej strukturze.

3.1. Środowisko testowe

Algorytm ABR został wygenerowany metodą uczenia ze wzmocnieniem zaktualizowanym systemem Pensieve. Platforma testowa observing-rl-agents-netai2019 [18] posłużyła do ewaluacji modelu w testowym środowisku symulacyjnym oraz do wizualizacji zgormadzonych rezultatów. System Pensieve wymaga zbiorów danych wejściowych o określonym szczegółowo formacie w celu wytrenowania modelu. Jednym z podstawowych składników tego zbioru są zapisy parametrów transmisji w sieciach komputerowych, które są wykorzystywane w procesie uczenia agenta jako informacja o dostępnej w danej chwili przepustowości łącza. Kolejnym składnikiem zbioru danych wejściowych są fragmenty wideo. Każdy z tych fragmentów reprezentuje określoną przepływność/jakość fragmentu w danym

przedziale czasu. Drugi moduł realizuje dwa etapy badań. Pierwszy z nich to ewaluacja wytrenowanego modelu w środowisku symulacyjnym wykorzystującym zmodyfikowane zapisy transmisji w sieci komputerowej [19] oraz generowanych sztucznie zapisy transmisji na łączach o stałej przepustowości. Drugim etapem jest analiza zebranych rezultatów ewaluacji modelu z użyciem narzędzia lime [20]. Jednym z podstawowych celów zastosowania tego narzędzia jest sprawdzenie i ocena wpływu poszczególnych cech wejściowych na jakość odpowiedzi generowanych przez badany model. Ocena stanowi wartość metryki QoE [8]:

$$QoE_1^K = \sum_{k=1}^{K} q(R_k) - \lambda \sum_{k=1}^{K-1} |q(R_{k+1}) - q(R_k)| - \mu \sum_{k=1}^{K} \left(\frac{d_k(R_k)}{c_k} - B_k \right) - \mu_s T_s \quad (1)$$

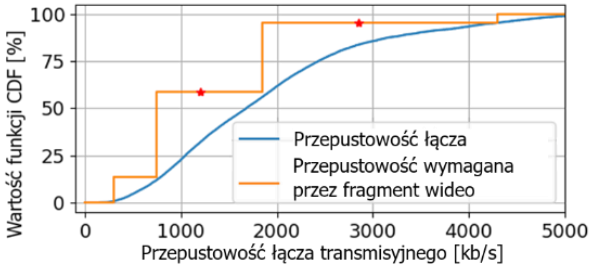
gdzie:

- $\sum_{k=1}^{K} q(R_k)$ – średnia wartość jakości fragmentu wideo w perspektywie całego wideo,
- $\sum_{k=1}^{K-1} |q(R_{k+1}) - q(R_k)|$ – obserwowalny zakres zmian w jakości między kolejnymi fragmentami wideo,
- $\sum_{k=1}^{K} \left(\frac{d_k(R_k)}{c_k} - B_k \right)$ – jest to łączny czas jaki użytkownik spędził w oczekiwaniu na załadowanie kolejnych fragmentów wideo. Buforowanie zachodzi wtedy, gdy czas pobierania fragmentu wideo zajmuje dłużej niż czas pobieranego fragmentu wideo,
- T_s – początkowe opóźnienie,
- λ, μ, μ_s – to współczynnik istotności (wagi) dla odpowiednio: średniej wariancji jakości fragmentów wideo, łącznego czas buforowania całego wideo i początkowego opóźnienia.

3.2. Zbiory danych i ich charakterystyka

Zbiór wszystkich zapisów parametrów transmisji został podzielony na zbiór testowy i zbiór treningowy. Zbiór testowy jest skomponowany z udostępnionego zbioru HSDPA [19] i generowanych sztucznie danych o transmisji na łączu o stałej przepustowości. Dane generowane sztucznie reprezentują warunki transmisji dla różnych (ale o stałej wartości) przepływności łącza, od 300 kb/s do 4700 kb/s. Ich uwzględnienie w zbiorze danych służy odpowiedzi na pytanie na ile wytrenowany model jest skłonny do dobierania lepszej jakości fragmentów kosztem większej częstotliwości zmian jakości wideo, czy też zapewnienia jak najpłynniejszy obraz i unika pobierania fragmentów wyższej jakości. Zbiór treningowy jest kompozycją różnych źródeł danych, w tym zapisów z HSDPA oraz danych z bazy FCC [21]. Ten zbiór pokrywa się w pełni ze zbiorem danych, które został wykorzystany na potrzeby procesu uczenia modelu Pensieve.

Oba powyższe zbiory danych zostały wykorzystane do analizy własności opracowanego modelu w testowym środowisku symulacyjnym (moduł drugi) i pozwoliły na obserwację zachowania tego modelu w systemie transmisji strumieniowej wideo opartym o zasady DASH. Testy zostały wykonane z wykorzystaniem łącza sieciowego o zestawie parametrów, które były wykorzystywane w procesie uczenia jak i całkowicie nowych (nie wykorzystywanych w procesie uczenia). Dzięki temu uzyskane rezultaty testów są bardziej obiektywne z punktu widzenia jego wykorzystania w



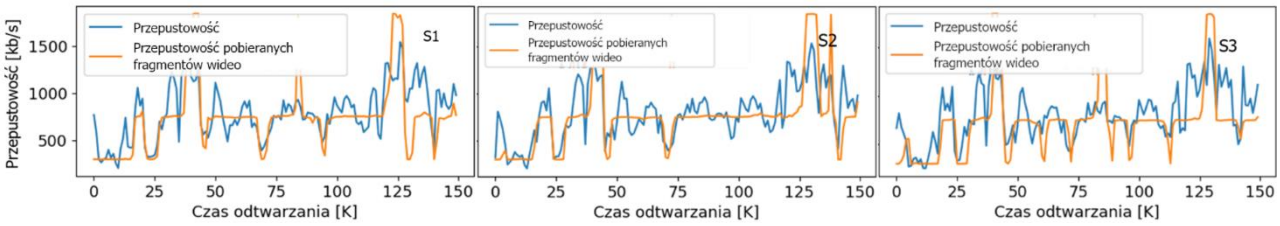
Rysunek 5: Wartość funkcji CDF dla przepustowości w fragmentach wideo. Materiał wideo: „Red Bull Playstreets".

rzeczywistym systemie sieciowych usług strumieniowych. Innymi słowami, takie podejście stawia na jak największą zbieżność symulacji z oczekiwanymi warunkami rzeczywistymi a nie na testowanie warunków krytycznych (specyficznych dla danej technologii transmisji danych i topologii infrastruktury sieciowej). Treści wideo zostały przygotowane w postaci list rozmiarów kolejnych fragmentów wideo w bajtach. Zostały sporządzone oddzielnie dla każdej dostępnej przepływności wraz z plikiem manifestu. Przepływności wideo w użytych zbiorach wideo dobrano tak, by były jak najbliższe przepływnościom zdefiniowanym w zbiorze treningowym oraz zostały one dostosowane do zaprezentowanego wcześniej formatu danych: 300, 750, 1200, 1850, 2850, 4300 Kb/s.

4. Wyniki testów

Na potrzeby zbadania zagadnienia użyteczności uczenia ze wzmocnieniem w generowaniu algorytmów ABR opracowano poszczególne etapy wraz z scenariuszami testowymi:

1. Wytrenowanie modelu.
2. Ewaluacja modelu w emulowanym środowisku dla utworzonego zbioru parametrów łącza transmisyjnego:
   a. scenariusz 1 – dalej nazywany S1, emulacja odtwarzania 10 minut wideo „Big Buck Bunny" [22, 23]– film animowany z dużą ilością statycznych ujęć,
   b. scenariusz 2 – dalej nazywany S2, emulacja odtwarzania 10 minut wideo „Red Bull Playstreets" [22] – film sportowy z dużą ilością dynamicznych ujęć,
   c. scenariusz 3 – dalej nazywany S3, emulacja odtwarzania 10 minut wideo „Valkaama" [22] – film obyczajowy z różnym zakresem ujęć.
3. Analiza statystyczna zebranych pomiarów.



Rysunek 7: Zależność pomiędzy procesem decyzyjnym, a zmianami dostępnej przepustowości łącza transmisyjnego.

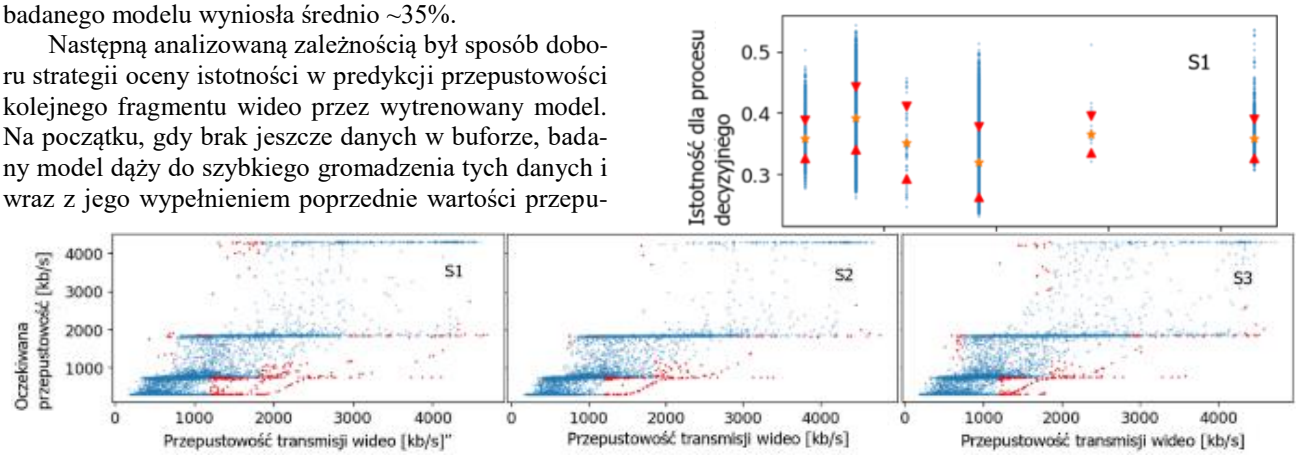
4. Wizualizacja kluczowych metryk na potrzeby interpretacji wyników.

Dla każdego ze scenariuszy emulację środowiska przeprowadzono dla tych samych warunków łącza internetowego. Analiza statystyczna obejmuje badanie zależności między danymi wejściowymi oraz tym jaki mają wpływ na metrykę QoE.

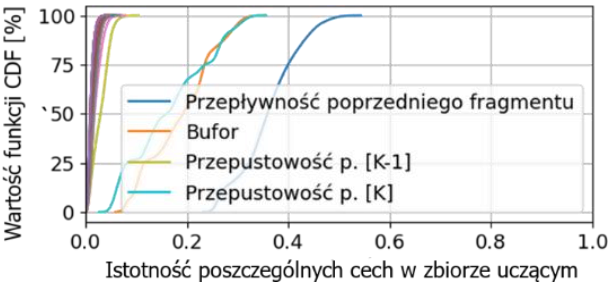
Pierwszą badaną cechą jest rozkład wybieranych przepustowości. Dla wszystkich scenariuszy badawczych S1, S2, S3 uzyskano bardzo podobne rezultaty. Na przykładzie rysunku 5 można zauważyć, że mimo dostępności przepustowości 1200 kb/s, 2850 kb/s badany model unika wyboru fragmentów o tych przepustowościach, co oznacza, że ich wybór jest suboptymalny. Można to potwierdzić na podstawie obserwacji danych na rysunku 7. Przedstawia on wynik analizy istotności przepustowości ostatniego fragmentu i dostępnej przepustowości w trakcie wyboru przepustowości kolejnego fragmentu. Dla przepustowości 1200 kb/s i 2850 kb/s widoczne jest znacznie mniejsze zagęszczenie lub całkowity brak istotnych relacji, natomiast dla pozostałych przepustowości widać zdecydowanie większą istotność relacji. Jednocześnie należy podkreślić, że we wszystkich przeprowadzonych testach zauważono że poprzednia przepustowość fragmentu wideo jest jednym z kluczowych elementów w doborze kolejnego fragmentu tego wideo.

Widoczna jest tendencja, że dla niższych przepustowości (<=2850 kb/s) oczekiwana jest coraz wyższa przepustowość kolejnego fragmentu i ma istotny wpływ na wartości metryki QoE. Natomiast dla wyższych przepustowości (>2850 kb/s) słabnie i oczekiwana przepustowość jest taka sama lub niższa. Ilustruje to rysunek 6. Generalny wniosek ze wszystkich zrealizowanych testów jest taki, że istotność wartości przepływności poprzedniego fragmentu wideo w procesie decyzyjnym badanego modelu wyniosła średnio ~35%.

Następną analizowaną zależnością był sposób doboru strategii oceny istotności w predykcji przepustowości kolejnego fragmentu wideo przez wytrenowany model. Na początku, gdy brak jeszcze danych w buforze, badany model dąży do szybkiego gromadzenia tych danych i wraz z jego wypełnieniem poprzednie wartości przepustowości fragmentów stają się coraz mniej istotne. Krytyczny moment następuje po przekroczeniu zapełnienia objętości bufora treścią wideo o czasie odtwarzania powyżej 20 sekund. Wtedy, ponownie wartość przepustowości poprzednio pobranego fragmentu staje się znaczący w predykcji. Należy jednak zaznaczyć, że ta zaobserwowana istotność jest mniej zależna od zapełnienia bufora i maleje wraz z rosnącą ilością buforowanych danych. Istotna część testów została poświęcona zagadnieniu wpływu zmian w dostępnej przepustowości łącza transmisyjnego na dobór przepływności kolejnych fragmentów wideo. Wyniki przedstawione na rysunku 7 dowodzą, że w trakcie testów zaobserwowano zbliżone działanie badanego modelu dla wszystkich trzech scenariuszy. W przypadkach gdy treść wideo wymagała łącza o dużych wartościach przepływności, model podejmował decyzje maksymalizujące wykorzystanie łącza. Następnie, po osiągnięciu stanu wypełnienia łącza, następowała decyzja o dość gwałtownym zmniejszeniu wymagań odnośnie przepustowości dla kolejnego fragmentu. Ta strategia pozwalała na zachowanie stałych wartości metryki QoE dzięki danym zgromadzonym w buforze. Zaobserwowano też, że przez większość czasu transmisji wideo średnia wartość przepustowości fragmentów wideo wynosiła 750 kb/s. Wartość ta wynika z faktu, że przepustowość łącza w zakresie 0-750 kb/s spełniała zapotrzebowanie dla większości pobieranych fragmentów. Dodatkowo, na podstawie rysunku 7, można stwierdzić, że w przypadku filmu w scenariuszu 2, wykorzystywana była największa przepustowość łącza przy jednocześnie najmniejszej liczbie zmian przepustowości pomiędzy kolejnymi fragmentami wideo. Natomiast, dla scenariusza 3 transmisja wideo miała najmniejsze zapotrzebowanie na dostępną przepustowość łącza ale charakteryzowała się największą fluktuacją przepustowości pomiędzy poszczególnymi, po-



Rysunek 8: Relacja pomiędzy przepustowością pobieranych fragmentów wideo a oczekiwaną wartością przepustowości transmisji.



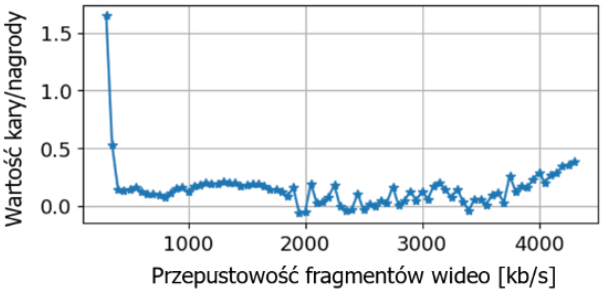
Rysunek 9: Wartość funkcji CDF dla istotności dla średniej wartości metryki QoE w relacji do poszczególnych cech zbioru uczącego. Symbol K oznacza k-tą decyzję modelu. Materiał wideo: „Big Buck Bunny".

bieranymi fragmentami.

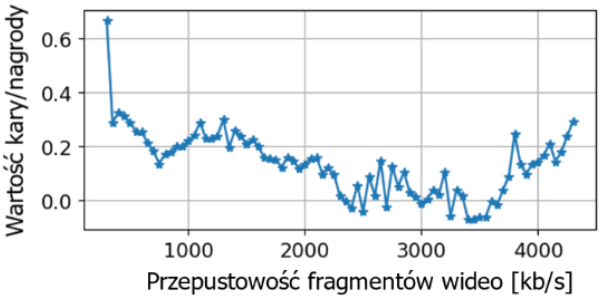
Omówione wyżej zależności zostały dodatkowo przeanalizowane pod kątem relacji pomiędzy przepustowością pobieranych fragmentów a oczekiwaną wartością przepustowości transmisji. Relacje te dla wszystkich trzech scenariuszy badawczych przedstawia rysunek 8. Na tym rysunku, kolorem niebieskim oznaczono relacje optymalne dla chwilowej wartości dostępnej przepustowości łącza a kolorem czerwonym, oznaczano przeciwną sytuację. Dla wszystkich scenariuszy, w zdecydowanej większości przypadków, model dokonywał doboru optymalnej przepustowości. Jednakże widoczne są też różnice między scenariuszami, co wynika z cech charakterystycznych transmitowanej treści wideo. Dla scenariuszy S1 i S2 w zakresie mniejszych, oczekiwanej przepływności można zaobserwować większe zagęszczenie sub-optymalnych wyborów. Jak wspomniano wyżej, może to wynikać z charakteru wideo ponieważ scenariusz 1 oraz 2 obejmują odpowiednio, kategorie filmów animowanych i sportowych. Na podstawie wyników analizy, które zostały zebrane na rysunku 9, można stwierdzi, że kluczowymi elementami w ocenie wyboru przepływności kolejnego fragmentu wideo są:

- przepływność poprzedniego fragmentu wideo,
- dostępna przepustowość łącza,
- stan zapełnienia bufora.

Ten wniosek dodatkowo potwierdza przydatność definicji metryki QoE [12], w której powyższe parametry są głównymi składnikami. Należy się zastanowić czy parametr – stan zapełnienia bufora nie jest zbyt dużym uproszczeniem dla rzeczywistej zmienności treści wideo. Opracowane rozwiązania monitorują jedynie jego stan zapełnienia a nie z jaką dynamiką zmienia się jego objętość. Zakładając, że różnica w przyrostach objętości



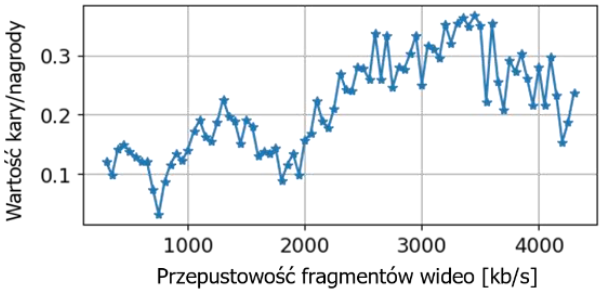
Rysunek 10: Relacja pomiędzy wartościami kary/nagrody związanej z wartością metryki QoE a przepustowością fragmentów wideo. Materiał wideo: „Valkaama".



Rysunek 11: Relacja pomiędzy wartościami kary/nagrody związanej z wartością metryki QoE a przepustowością fragmentów wideo. Materiał wideo: „Big Buck Bunny".

może być zależna od dynamiki wideo, świadczy to o pominięciu tej zależności w dotychczasowych rozwiązaniach, jak i braku jej uwzględnienia w badanym modelu. Stanowi to istotny wniosek prze definiowaniu kolejnych etapów rozwoju tego modelu.

Ostatnim analizowanym aspektem działania wytrenowanego modelu była jego zdolność do adaptacji w sytuacji sterowania transmisją zróżnicowanych materiałów wideo. Wyniki dla tej części badań są przedstawione na rysunkach 10, 11 oraz 12. Można zaobserwować, że wytrenowany model, którego proces uczenia oparty był jedynie na pojedynczym zbiorze danych wideo, próbuje dostosować się do nowych warunków środowiska (do materiałów o zdecydowanie innych cechach, np. dynamiki zmian przepustowości). Dla tych samych parametrów łącza sieciowego, ale dla różnych kategorii wideo otrzymano znacząco różne wyniki testów. Można zatem wysunąć wniosek, że trenowanie modelu na tylko pojedynczym zbiorze wideo jest istotnym uproszcze-



Rysunek 12: Relacja pomiędzy wartościami kary/nagrody związanej z wartością metryki QoE a przepustowością fragmentów wideo. Materiał wideo: „Red Bull Playstreets".

niem. Stanowi to cenną informację, że w przypadku przyszłych działań zmierzających do udoskonalenia modelu należałoby poszerzyć proces uczenia o treści wideo jak najbardziej zróżnicowanych parametrach (o różne kategorie wideo). Przedstawione wyżej relacje prowadzą do jeszcze jednej, istotnej obserwacji. Generalizacja znajomości własności środowiska jaką posiadł badany model na podstawie zbioru uczącego o niewystarczająco zróżnicowanych parametrach wideo prowadzi do zachowawczych decyzji odnośnie doboru przepływności kolejnego fragmentu wideo.

5. Podsumowanie

Idea realizacji algorytmów ABR z wykorzystaniem technik jak uczenie ze wzmocnieniem może być trakto-

wana jako istotna alternatywa dla rozwiązań klasycznych. W trakcie wykonanych testów potwierdzono, żetego typu rozwiązania mogą być w prosty sposób implementowane w klientach DASH, takich jak chociażby referencyjny klient dash.js. Algorytmy z rodziny Pensieve, w swojej pierwotnej postaci, opierają się jedynie o dane o przepustowości łącza, rozmiary pobieranych fragmentów wideo oraz informacje o dostępnych poziomach jakości. Zaprezentowane wyniki badań zależności wpływu różnych kategorii treści wideo odbieranej przez klienta DASH na jakościowe parametry wchodzące w skład metryki QoE wykazano ich silne, wzajemne relacje. W praktyce oznacza to, że w zależności na jakim zestawie danych wideo model został wytrenowany, będzie w konsekwencji inaczej radził sobie dla treści wideo innej kategorii. Na podstawie uzyskanych rezultatów zauważono, że są to subtelne różnice w wartościach metryki QoE pomiędzy takimi kategoriami wideo jak film animowany, akcji i sport. Jako przyszły, niezwykle istotny kierunek badań zaproponowanego modelu, należy wskazać testy wzajemnego wpływ kilku klientów DASH uruchomionych w tym samym segmencie sieci. W takim przypadku scenariusze testowe należałoby rozbudować o dodatkowy element pozwalający na ewaluację wpływu wzajemnej konkurencji o tą samą pulę dostępnej przepustowości.

Literatura

[1] Cisco Visual Networking Index: Forecast and Methodology 2016-2021, High Efficiency Video Coding (HEVC) Algorithms and Architectures (2017).

[2] K. u. R. Laghari, O. Issa, F. Speranza, T. H. Falk, Quality-of-Experience perception for video streaming services: Preliminary subjective and objective results, Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference, Hollywood, CA, USA, (2012) 1-9.

[3] A. Reuban, MPEG-DASH Enhanced Multimedia Streaming, International Journal of Advanced Research in Computer Science and Software Engineering, 4 (2014) 848-851.

[4] D. You, S. -H. Kim, D. H. Kim, ATSC 3.0 ROUTE/DASH Signaling for Immersive Media: New Perspectives and Examples, in IEEE Access, 9 (2021) 164503-164509, https://dx.doi.org/10.1109/ACCESS.2021.3133626.

[5] I. Sodagar, The MPEG-DASH Standard for Multimedia Streaming Over the Internet, IEEE MultiMedia, 18 (2011) 62-67, https://doi.org/10.1109/MMUL.2011.71.

[6] O. Izima, R. de Fréin, A. Malik, A Survey of Machine Learning Techniques for Video Quality Prediction from Quality of Delivery Metrics, Electronics, 10 (2021) 2851, https://dx.doi.org/10.3390/electronics10222851.

[7] T-Y. Huang, R. Johari, N. McKeown, M. Trunnell, W. Mark, A buffer-based approach to rate adaptation: Evidence from a Large Video Streaming Service, SIGCOMM Computer Communication Review, New York, NY, USA, 44 (2014) 187-198, https://dx.doi.org/10.1145/2619239.2626296.

[8] X. Yin, A. Jindal, V. Sekar, B. Sinopoli, A Control-Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP, SIGCOMM Computer Communication Review, New York, NY, USA, 45 (2015) 325-338, https://doi.org/10.1145/2829988.2787486.

[9] K. Spiteri, R. Sitaraman, D. Sparacio, From Theory to Practice: Improving Bitrate Adaptation in the DASH Reference Player, ACM Transactions on Multimedia Computing, Communications, and Applications, New York, NY, USA, 15 (2019) 1-29, https://doi.org/10.1145/3336497.

[10] M. Otterlo, M. Wiering, Reinforcement Learning and Markov Decision Processes, Reinforcement Learning: State of the Art, (2012) 3-42, https://doi.org/10.1145/2829988.2787486.

[11] S. Bhatt, Reinforcement Learning 101, Learn the essentials of Reinforcement Learning!, https://towardsdatascience.com/reinforcement-learning-101-e24b50e1d292, [03.11.2022].

[12] H. Mao, R. Netravali, M. Alizadeh, Neural Adaptive Video Streaming with Pensieve, Association for Computing Machinery, Los Angeles, CA, USA, (2017) 197–210, https://doi.org/10.1145/3098822.3098843.

[13] I. Grondman, L. Busoniu, G. Lopes, R. Babuska, A Survey of Actor-Critic Reinforcement Learning: Standard and Natural Policy Gradients, IEEE Transactions on Systems, Man and Cybernetics Part B-Cybernetics, 42 (2012) 1291-1307, https://doi.org/10.1109/TSMCC.2012.2218595

[14] H. -C. Jang, Y. -C. Huang, H. -A. Chiu, A Study on the Effectiveness of A2C and A3C Reinforcement Learning in Parking Space Search in Urban Areas Problem, International Conference on Information and Communication Technology Convergence, Jeju, South Korea, (2020) 567-571, https://doi.org/10.1109/ICTC49870.2020.9289269.

[15] J. Schmidhuber, Deep learning in neural networks: An overview, Neural Networks, 61 (2015) 85-117, https://doi.org/10.1016/j.neunet.2014.09.003.

[16] Z. Tian, L. Zhao, L. Nie, P. Chen, S. Chen, Deeplive: QoE Optimization for Live Video Streaming through Deep Reinforcement Learning, IEEE 25th International Conference on Parallel and Distributed Systems, Tianjin, China, (2019) 827-831, https://doi.org/10.1109/ICPADS47876.2019.00122.

[17] Y. Liu, D. Wei, C. Zhang, W. Li, Distributed Bandwidth Allocation Strategy for QoE Fairness of Multiple Video Streams in Bottleneck Links, Future Internet, 14 (2022) 152, https://doi.org/10.3390/fi14050152.

[18] A. Dethise, M. Canini, S. Kandula, Cracking Open the Black Box: What Observations Can Tell Us About Reinforcement Learning Agents, Association for Computing Machinery, New York, NY, USA, (2019) 29-36, https://doi.org/10.1145/3341216.3342210.

[19] H. Riiser, P. Vigmostad, C. Griwodz, P. Halvorsen, Commute Path Bandwidth Traces from 3G Networks: Analysis and Applications, Association for Computing Machinery, New York, NY, USA, (2013) 114-118, https://doi.org/10.1145/2483977.2483991.

[20] M. Ribeiro, S. Singh, C. Guestrin, "Why Should I Trust You?": Explaining the Predictions of Any Classifier, Association for Computational Linguistics: Demonstrations, San Diego, California, (2016) 97-101, http://dx.doi.org/10.18653/v1/N16-3020.

[21] Raw Data - Measuring Broadband America 2016, Federal Communications Commission, https://www.fcc.gov/reports-research/reports/measuring-broadband-america/, [03.11.2022].

[22] ITEC - Dynamic Adaptive Streaming over HTTP, https://dash.itec.aau.at/contact/, [03.11.2022].

[23] C. Müller, S. Lederer, C. Timmerer, H. Hellwagner, Dynamic Adaptive Streaming over HTTP/2.0, IEEE International Conference on Multimedia and Expo, (2013) 1-6, http://dx.doi.org/10.1109/ICME.2013.6607498.