

Politechnika Śląska
Wydział Automatyki, Elektroniki
i Informatyki

mgr inż. Jakub Szyguła

ADAPTACYJNE ALGORYTMY
ZARZĄDZANIA NATEŻENIEM RUCHU
W SIECI INTERNET

Rozprawa doktorska napisana pod kierunkiem
dr hab. inż. Adama Domańskiego, Prof. PŚ

Gliwice 2022

Spis treści

Lista publikacji zawartych w niniejszej rozprawie doktorskiej	2
Lista pozostałych publikacji	3
1 Wstęp	5
2 Implementacja oraz badania adaptacyjnych modeli mechanizmów AQM	11
2.1 Publikacja 1: Model mechanizmu aktywnego zarządzania kolejką wykorzystujący odpowiedź z kilku kontrolerów PI^α	11
2.2 Publikacja 2: Dobór parametrów mechanizmu AQM w oparciu o techniki uczenia maszynowego oraz uczenia przez wzmocnienie .	13
2.3 Publikacja 3: Adaptacyjny mechanizm AQM dostosowujący się do intensywności ruchu oraz stopnia samopodobieństwa	16
2.4 Publikacja 4: Adaptacyjny mechanizm aktywnego zarządzania kolejką oparty o metody uczenia nadzorowanego	19
2.5 Publikacja 5: Model mechanizmu AQM dla priorytetowych danych ruchu urządzeń Internetu Rzeczy	24
3 Podsumowanie	26
Bibliografia	27

Lista publikacji zawartych w niniejszej rozprawie doktorskiej

1. Domański, A.; Domańska, J.; Czachórski, T.; Klamka, J.; Marek, D.; Szyguła, J. AQM mechanism with the dropping packet function based on the answer of several PI^α controllers. 26th International Conference on Computer Networks (CN 2019), Communications in Computer and Information Science, Springer International Publishing, vol. 1039, s. 400-412, 2019. https://doi.org/10.1007/978-3-030-21952-9_29.
2. Szyguła, J.; Domański, A.; Domańska, J.; Czachórski, T.; Marek, D.; Klamka, J. AQM mechanism with neuron tuning parameters. 12th Asian Conference on Intelligent Information and Database Systems (ACIIDS 2020). Lecture Notes in Artificial Intelligence, vol. 12034, s. 299-311. Springer, Cham. https://doi.org/10.1007/978-3-030-42058-1_25.
3. Marek, D.; Szyguła, J.; Domański, A.; Domańska, J.; Filus, K.; Szczygieł, M. Adaptive Hurst-Sensitive Active Queue Management. Entropy 2022, vol. 24(3), 418. <https://doi.org/10.3390/e24030418>.
4. Szyguła, J.; Domański, A.; Domańska, J.; Marek, D.; Filus, K.; Mendla, S. Supervised Learning of Neural Networks for Active Queue Management in the Internet. Sensors 2021, vol. 21(15), 4979. <https://doi.org/10.3390/s21154979>.
5. Domański, A.; Domańska, J.; Czachórski, T.; Klamka, J.; Szyguła, J.; Marek, D. The IoT gateway with active queue management. International Journal of Applied Mathematics and Computer Science, vol. 31 (1), s. 165-178. <https://doi.org/10.34768/amcs-2021-0012>.

Lista pozostałych publikacji

1. Filus, K.; Domański, A.; Domańska, J.; Marek, D.; Szyguła, J. Long-Range Dependent Traffic Classification with Convolutional Neural Networks Based on Hurst Exponent Analysis. *Entropy* 2020, vol. 22(10), 1159. <https://doi.org/10.3390/e22101159>.
2. Domański, A.; Domańska, J.; Filus, K.; Szyguła, J.; Czachórski, T. Self-Similar Markovian Sources. *Appl. Sci.* 2020, vol. 10(11), 3727. <https://doi.org/10.3390/app10113727>.
3. Domański, A.; Domańska, J.; Czachórski, T.; Klamka, J.; Szyguła, J. The AQM dropping packet probability function based on non-integer order $PI^\alpha D^\beta$ controller. 9th International Conference on Non-integer Order Calculus and its Applications, Łódź, Poland. *Lecture Notes in Electrical Engineering*, Springer International Publishing, vol. 496, s. 36-48, 2019, https://doi.org/10.1007/978-3-319-78458-8_4.
4. Marek, D.; Domański, A.; Domańska, J.; Czachórski T.; Klamka J.; Szyguła J. Combined diffusion approximation - simulation model of AQMs transient behavior. *Computer Communications* 2021, vol. 166, s. 40-48. <https://doi.org/10.1016/j.comcom.2020.11.014>
5. Marek, D.; Domański, A.; Domańska, J.; Szyguła, J.; Czachórski, T.; Klamka, J. Diffusion Model of a Non-Integer Order PI^γ Controller with TCP/UDP Streams. *Entropy* 2021, vol. 23, 619. <https://doi.org/10.3390/e23050619>.
6. Czajkowski, A.; Remiorz, L.; Pawlak, S.; Remiorz, E.; Szyguła, J.; Marek, D.; Paszkuta, M.; Drabik, G.; Baron, G.; Paduch, J.; Antemijczuk, O. Global Water Crisis: Concept of a New Interactive Shower Panel Based on IoT and

Cloud Computing for Rational Water Consumption. Appl. Sci. 2021, vol. 11, 4081. <https://doi.org/10.3390/app11094081>.

7. Domański, A.; Domańska, J.; Czachórski, T.; Klamka, J.; Szyguła, J.; Marek, D. Diffusion approximation model of TCP NewReno congestion control mechanism. Conference on Modelling Methods in Computer Systems, Networks and Bioinformatics, 14-15 October 2019, Paris. Springer Nature Computer Science, vol. 1 (43). <https://doi.org/10.1007/s42979-019-0032-x>.
8. Domański, A.; Domańska, J.; Czachórski, T.; Klamka, J.; Marek, D.; Szyguła, J. The Influence of the Traffic Self-similarity on the Choice of the Non-integer Order PI^α Controller Parameters. 32nd International Symposium on Computer and Information Sciences (ISCIS), Communications in Computer and Information Science, Springer International Publishing, vol. 935, s. 76-83, 2018. https://doi.org/10.1007/978-3-030-00840-6_9.
9. Domański, A.; Domańska, J.; Czachórski, T.; Klamka, J.; Marek, D.; Szyguła, J. GPU accelerated non-integer order $PI^\alpha D^\beta$ controller used as AQM mechanism. 25th International Conference on Computer Networks (CN 2018), Communications in Computer and Information Science, Springer International Publishing, vol. 860, s. 286-299, 2018. https://doi.org/10.1007/978-3-319-92459-5_23.
10. Remiorz, L.; Czajkowski, A.; Pawlak, S.; Remiorz, E.; Szyguła, J.; Drabik, G.; Marek, D.; Antemijczuk, O.; Paduch, J.; Baron, G.; Paszkuta, M. The concept of an interactive shower panel in terms of the assumptions of Industry 4.0. Contemporary problems of power engineering and environmental protection 2020, Politechnika Śląska, s.7-14, ISBN 978-83-950087-9-5.
11. Czajkowski, A.; Remiorz, L.; Pawlak, S.; Remiorz, E.; Szyguła, J. Methodology of modeling the detection of limescale deposits in sanitary installations. 21st International Multidisciplinary Scientific GeoConference SGEM 2021, 7-10 December, 2021, International Multidisciplinary Scientific GeoConference & EXPO SGEM, vol. 3.2, 2021, Sofia, STEF92 Technology, ISBN 978-619-7603-32-3. <https://doi.org/10.5593/sgem2021V/3.2/s12.18>.

Rozdział 1

Wstęp

Rozwój technologii informatycznych oraz nieustanny wzrost liczby użytkowników posiadających nieprzerwany dostęp do Internetu, spowodował napotkanie nowych problemów. Pierwotnie sieć Internet wykorzystywana była do przesyłania danych niewielkiego rozmiaru. Problem transferu dużych zbiorów danych oraz treści multimedialnych pojawił się z początkiem XXI wieku. Jednym z podstawowych problemów, które należy rozwiązać w sieciach komputerowych jest zagadnienie zapobiegania wystąpienia przeciążeń w transmisji danych. Początkowo problem przeciążeń w sieciach komputerowych był rozwiązywany w sposób taki, w którym to węzły samodzielnie monitorowały swoje przeciążenia. W przypadku ich zaobserwowania i detekcji wysyłały o tym fakcie informację bezpośrednio do nadajników. Zgodnie z założeniem, po odebraniu takiej informacji nadajnik zmniejszał prędkość nadawania, co z kolei przyczyniało się do zniwelowania danego przeciążenia. Do najpopularniejszych rozwiązań tego typu należy zaliczyć mechanizm tzw. bitu ostrzegawczego (ang. warning bit), który stosowany był przez firmę IBM. Druga najpopularniejsza metoda oparta była o tzw. pakiety tłumiące (ang. choke packets) [1]. W przypadku szybkich sieci mechanizmy te okazały się jednak zbyt wolne i informacja o przeciążeniach docierała do nadajnika zbyt późno. Z tej przyczyny mechanizmy te przestały być powszechnie stosowane. Dlatego też w kolejnych latach zdecydowano się na stworzenie pewnych mechanizmów, w których to nadajnik sam decyduje jak prędko ma wysyłać dane. Mechanizmy mające zapewnić wspomniane postulaty zostały wbudowane w protokół warstwy transportowej sieci Internet. Są one szerzej opisane w licznych dokumentach RFC (ang. Request for Comments).

Pierwszym autorem, który opracował zasady sterowania przeciążeniami w protokole TCP był Van Jacobson w 1988 roku [2]. Natomiast pierwszą w pełni

opisaną metodą sterowania przeciążeniami oraz przygotowaną do wdrożenia w sieci Internet, był mechanizm TCP Tahoe. Rozwiązanie to było wielokrotnie modyfikowane i rozwijane w kolejnych latach. W zależności od klasy sieci przewodowych oraz bezprzewodowych powstało wiele różnych wersji algorytmów zarządzania oknem przeciążenia w protokole TCP. Na dzień dzisiejszy najpopularniejszym i najszerszej stosowanym mechanizmem jest TCP New Reno.

W sieci Internet mechanizm przeciwdziałania występowania przeciążeń realizowany jest poprzez wbudowanie w protokół transportowy TCP pewnych specyficznych mechanizmów zarządzania transmisją. Od czasu powstania pierwszego mechanizmu zarządzania oknem przeciążenia powstało wiele nowych wersji tego rozwiązania, które dedykowane są zarówno dla sieci homogenicznych, w których transmisja odbywa się za pomocą połączeń kablowych [3] lub bezprzewodowych [4], a także dla sieci heterogenicznych, gdzie trasa składa się z połączeń przewodowych oraz radiowych [5].

Na etapie projektowania protokołu TCP jego twórcy zdawali sobie sprawę z istoty zagrożenia, jakie może się pojawić gdy system odbiorcy nie będzie w stanie w odpowiednim czasie pobierać danych z bufora, do którego kierowane są pakiety wysyłane od nadawcy. Z tego powodu w protokole TCP uwzględniono mechanizm okna odbiornika (ang. Receiver Window). Określa on w precyzyjny sposób jaką ilość danych może wysłać nadawca bez informacji zwrotnej o aktualnie dostępnym miejscu w kolejce, otrzymanej od systemu odbiorcy. W przypadku braku otrzymania takiego potwierdzenia transmisja zostanie wstrzymana. Celem okna odbiornika jest dostosowanie prędkości przesyłu do jego własnych możliwości.

Mechanizm okna przeciążenia (ang. Congestion Window) pozwala odbiorcy uchronić się przed przepełnieniem kolejki, a więc przed utratą danych. Co istotne w tym przypadku nadawca nie otrzymuje wprost informacji o wolnym miejscu w buforze, a wszelkie decyzje podejmowane są tylko na podstawie liczby otrzymanych potwierdzeń. Maksymalna liczba pakietów, które mogą zostać wysłane przez nadawcę nie może przekroczyć wartości zarówno okna odbiorcy, jak też przeciążenia. Okno przeciążenia ma więc na celu dostosowanie prędkości przesyłu do możliwości sieci.

Jednym z negatywnych skutków istnienia protokołów zarządzania oknem przeciążenia może być zjawisko globalnej synchronizacji. Wartym odnotowania jest, że organizacja IETF (ang. Internet Engineering Task Force) mając na uwadze wcześniej przytoczone zagrożenia, poleca stosowanie algorytmów aktywnego zarządzania kolejką (ang. Active Queue Management). Mają one na celu zapobieganie niebezpieczeństwu występowania wspomnianego wcześniej zjawiska - glo-

balnej synchronizacji oraz utraty łączności. Mają także na uwadze zwiększenie efektywności transmisji. Koncepcja ta związana jest z prewencyjnym odrzucaniem pakietów, nawet gdy w buforze wciąż znajduje się jeszcze wolne miejsce. Z kolei samo prawdopodobieństwo odrzucenia pakietu wzrasta wraz z poziomem zajętości kolejki. Pakiety odrzucane są losowo, skutkiem czego większość użytkowników nie zaobserwuje jakichkolwiek problemów związanych z przesyłem danych.

Wyróżnić można dwa podstawowe mechanizmy zarządzania zajętością kolejki w transmisji w sieci Internet. Pierwszym z nich jest metodologia pasywnego odrzucania danych. Natomiast do drugiej grupy rozwiązań zaliczamy mechanizmy aktywnego zarządzania kolejkami.

W przypadku metod statycznego zarządzania kolejką pakiety wyrzucane są dopiero po przepełnieniu bufora. W tym przypadku jedynym sposobem zarządzania średnim czasem oczekiwania pakietu w kolejce jest dobór maksymalnego dopuszczalnego rozmiaru długości kolejki. W przypadku przepełnienia bufora dodatkową wadą jest zjawisko, w którym pakiety są gubione w większej ilości. To natomiast skutkuje pojawieniem się problemu globalnej synchronizacji, która to omówiona będzie dokładnie w dalszych rozdziałach niniejszej rozprawy.

Mechanizmy aktywnego zarządzania kolejką AQM (ang. Active Queue Management) bazują natomiast na stałej obserwacji łącza, czyli stopnia zapełnienia bufora. Na tej podstawie podejmowane są następnie decyzje o wcześniejszym - prewencyjnym odrzuceniu pakietu. Pakiety te są wyrzucane losowo zgodnie z wyliczoną wartością funkcji prawdopodobieństwa. Pozwala to zwiększyć przepustowość sieci i zapewnić sprawiedliwy dostęp do łącza.

Pierwszym pełnoprawnym i zarazem wciąż najpopularniejszym algorytmem aktywnego zarządzania kolejką AQM jest mechanizm RED (ang. Random Early Detection), zaproponowany w 1993 roku przez Sally Floyd i Van Jacobson'a [6]. Mechanizm ten pozwala uzyskać rozsądny, akceptowalny poziom długości kolejki, a liczba odrzuconych prewencyjnie pakietów rośnie wraz ze wzrostem zapełnienia bufora. Ukazały się również prace, w których starano się zwiększyć wydajność tego mechanizmu [7]. Z kolei inne podejście do tego zagadnienia zaobserwowano w artykule [8], gdzie rozważano bardziej szczegółową analizę historii zajętości kolejek oraz zmianę w sposobie implementacji średniej ważonej długości kolejki. W pracy tej wykazano, że taka koncepcja może również znacząco poprawić wydajność transmisji.

Zastosowanie mechanizmu RED oraz jego licznych modyfikacji pozwala zmniejszyć poziom opóźnień w transmisji. Konieczne jest jednak właściwe dobranie parametrów mechanizmu aktywnego zarządzania kolejką [9]. W przeciwnym

przypadku, system TCP/RED staje się niestabilny [10]. Naturalnym następstwem tej sytuacji było pojawienie się wersji adaptacyjnych mechanizmu RED, które to automatycznie dostosowują parametry algorytmu w zależności od aktualnego natężenia ruchu. Pierwszym tego typu zaproponowanym rozwiązaniem był mechanizm ARED (ang. Adaptive Random Early Detection) [11]. W tym przypadku modyfikacja polega na dostosowaniu funkcji, za pomocą której wyznaczane jest prawdopodobieństwo odrzucenia pakietu w taki sposób, aby jej wartość oscylowała pomiędzy zdefiniowanym progiem minimalnym oraz maksymalnym. Przeprowadzone badania dowiodły, że rozwiązanie to zmniejsza liczbę odrzucanych pakietów oraz poziom zmienności opóźnień występujących w transmisji [12]. Rozwiązanie to niesie za sobą jednak również pewne wady. W pracy [13] zwrócono uwagę, że dostosowanie funkcji prawdopodobieństwa odrzucenia pakietu jest niestety czasochłonne. W związku z tym adaptacja do nieustannie zmieniających się warunków w sieci Internet nie odbywa się w sposób wystarczający. Dlatego też w ostatnich latach ukazało się wiele kolejnych prac, w których autorzy starali się zaproponować różne, nowe modyfikacje algorytmu RED, mając na uwadze zwiększenie efektywności transmisji. Swoista popularność prac prowadzonych w obszarze modyfikacji mechanizmów TCP oraz AQM wynika również z faktu, że z uwagi na rozległość sieci Internet, najlepszym sposobem wprowadzania usprawnień jest stosowanie mechanizmów separowalnych, które to implementować można na jednej krańcówce sieci, czyli w pojedynczych routerach oraz stacjach nadawczych.

Z drugiej strony, wszystkie algorytmy adaptacyjne, które znane są z literatury, uzależniają się od intensywności ruchu, który to z kolei powiązany jest z poziomem zajętości kolejki. Inną zaobserwowaną i bardzo istotną własnością ruchu, wpływającą na jego dynamikę i właściwości, są cechy samopodobieństwa, które pomijane były przez wiele lat w pracach nad mechanizmami mającymi na celu zwiększenie prędkości transmisji.

Zależności długoterminowe zaobserwowano po raz pierwszy w połowie XX wieku, kiedy to Sir H.E. Hurst pracował nad budową tamy na rzece Nil. Posiadał on zapisy zawierające zmiany linii brzegowych z przeszło 800 lat. Wtedy to też po przeprowadzeniu dokładnej analizy posiadanych danych Hurst wykazał, że poziom wody w rzece nie jest procesem całkowicie losowym, niezależnym od przeszłości. Swoje wnioski oparł na całkowicie innowacyjnej koncepcji, w stosunku do metod stosowanych przez współczesnych hydrologów. Wykorzystał on opublikowaną przez Alberta Einsteina 1908 roku pracę na temat ruchów Browna.

Hurst zaproponował, by poziom lustra wody w projektowanym zbiorniku przedstawić za pomocą położenia wspomnianej cząstki Browna, natomiast sumę

całorocznych wpływów wody jako skoki i zmiany tej cząstki [14, 15]. Przyjęcie powyższych założeń pozwoliło wykazać istnienie długoterminowej pamięci zdarzeń w przypadku nieskończonej długości szeregów czasowych [16]. Pomimo faktu, że terminy samopodobieństwa oraz zależności długoterminowych bywają używane zamiennie, należy pamiętać, że nie są one dokładnie tym samym [17].

Analiza prac z zakresu samopodobieństwa ruchu sieciowego, pozwala stwierdzić, że jest ono jedną z cech, która znacząco wpływa na stopień zajętości kolejki oraz właściwości transmisji [18, 19] i koniecznym jest uwzględnianie jego wpływu podczas przeprowadzania analizy służącej ocenie mechanizmów aktywnego zarządzania kolejką. Wyniki prac przeprowadzonych przez naukowców z Bellcore [20] jednoznacznie wykazały samopodobną cechę ruchu pakietów w sieci. Praca ta stanowiła motywację pod liczne badania, w których wykazano istotny wpływ stopnia samopodobieństwa na ruch w transmisji TCP [21]. Potwierdzono również jego występowanie w sieciach rozległych WAN (ang. Wide Area Network) [22]. O cechach samopodobieństwa ruchu w łączy transmisyjnym mówimy zazwyczaj wtedy, gdy jest on traktowany całościowo. Jednak w przypadku, gdy analizie poddany zostanie wyłącznie ruch poszczególnych usług i aplikacji, to również będzie się on cechował podwyższoną wartością stopnia samopodobieństwa [23]. Analiza danych rzeczywistego ruchu sieciowego, zgromadzona przez naukowców z Bellcore [20], a także danych ze zbiorów CAIDA [24] (Centrum analizy danych sieci Internet na Uniwersytecie Kalifornijskim w San Diego), pozwala stwierdzić, że na tym samym łączy transmisyjnym obserwuje się ruch o różnym stopniu samopodobieństwa. Wszelkie próby pominięcia powyższych reguł sprawiają, że w zasadzie niemożliwe staje się poprawne oszacowanie stopnia zajętości bufora [25].

Prowadzone w tej pracy badania nad adaptacyjnymi algorytmami zarządzania natężeniem ruchu mają wykazać, czy dobór i modyfikacja parametrów tego mechanizmu w czasie rzeczywistym pozwoli na zwiększenie prędkości transmisji w sieci Internet. Pośrednim celem niniejszej rozprawy jest więc optymalizacja poziomu zajętości kolejki, a co za tym idzie średniego czasu oczekiwania pakietu w kolejce.

Pomimo ukazania się licznych prac, w których rozważane są algorytmy aktywnego zarządzania kolejką, według mojej najlepszej wiedzy nie istnieją inne adaptacyjne mechanizmy, które skalują się i adaptują do pierwszorzędnych cech ruchu, czyli do jego natężenia, a także do zależności długoterminowych i cech samopodobieństwa ruchu sieciowego.

W ramach tej pracy zostaną zaproponowane pewne mechanizmy, a ich ewaluacja odbędzie się na drodze badań symulacyjnych oraz badań analitycznych.

Do prawidłowej oceny mechanizmów aktywnego zarządzania kolejką oraz wydajności sieci komputerowej konieczne jest stworzenie odpowiednich modeli mechanizmów sieciowych, a także rzeczywiste odzwierciedlenie ruchu danych. Modele analityczne oparte o teorie kolejkowania są często spotykane w pracach badawczych z zakresu tematyki sieci komputerowych. Z drugiej strony, podążając za wnioskami autorów prac [26, 27, 28, 29] stosowanie symulacji sieciowych pozwala znacząco usprawniać rozwiązania sieciowe.

Celem niniejszej pracy jest przedstawienie nowego podejścia do adaptacyjnych mechanizmów aktywnego zarządzania kolejką. Standardowo algorytmy te zmieniają swoje parametry dostosowując się do natężenia ruchu sieciowego. Zaproponowane podejście ma na celu dobór tychże parametrów zarówno w zależności od natężenia ruchu, jak również stopnia jego samopodobieństwa wyrażonego za pomocą parametru Hursta. Wprowadzenie tej modyfikacji powinno pozwolić na optymalizację poziomu zajętości kolejki, a co za tym idzie również średniego czasu oczekiwania pakietu w buforze. Optymalizacja kolejki w routerze powinna wydatnie zwiększyć efektywność transmisji w sieciach komputerowych.

Do oceny skuteczności zaproponowanych rozwiązań w przypadku modelu pętli otwartej (ang. open loop) wykorzystano metody symulacyjne. Modelowanie zachowania protokołów TCP i ich współpracy z mechanizmami aktywnego zarządzania kolejką oparto o model aproksymacji Fluid-Flow [30, 31]. W rozpatrywanym przypadku są to badania modelu pętli zamkniętej (ang. closed loop).

Tezę niniejszej rozprawy zdefiniowano w sposób następujący:

Teza. *Stworzenie adaptacyjnych mechanizmów zarządzania pakietami, dostosowujących się nie tylko do natężenia ruchu sieciowego, ale również do zależności długoterminowych, może znacząco zwiększyć efektywność transmisji w sieciach komputerowych.*

Rozdział 2

Implementacja oraz badania adaptacyjnych modeli mechanizmów AQM

W ramach niniejszego rozdziału omówiono zaproponowane adaptacyjne mechanizmy aktywnego zarządzania kolejką. Modele te następnie zaimplementowano, a rezultaty przeprowadzonych badań eksperymentalnych opublikowano w czasopiśmie naukowym oraz wygłoszono w ramach międzynarodowych konferencji naukowych. Publikacje te, wraz z dokładnym omówieniem wkładu autorskiego przytoczono w poniższych podrozdziałach.

2.1 Publikacja 1: Model mechanizmu aktywnego zarządzania kolejką wykorzystujący odpowiedź z kilku kontrolerów PI^α

W publikacji "AQM mechanism with the dropping packet function based on the answer of several PI controllers" przedstawiono nową koncepcję mechanizmu aktywnego zarządzania kolejką, która do wyznaczenia wartości prawdopodobieństwa odrzucenia pakietu wykorzystuje odpowiedź z trzech różnych kontrolerów PI^α :

$$p(q) = \begin{cases} p_1(q) & \text{if } q < 180 \\ p_1(q) + p_2(q) & \text{if } 180 \leq q < 220 \\ p_1(q) + p_2(q) + p_3(q) & \text{if } 220 \leq q \end{cases}$$

gdzie:

p_1 - odpowiedź pierwszego kontrolera,

p_2 - odpowiedź drugiego kontrolera,

p_3 - odpowiedź trzeciego kontrolera.

Uwzględnienie odpowiedzi z kolejnych kontrolerów przy wyliczaniu wartości funkcji prawdopodobieństwa odrzucenia pakietu, pozwala na jego adaptacyjne modyfikowanie, w sytuacji gdy zanotowana zajętość bufora przekracza kluczowe progi. W zaproponowanej formule wykorzystywana jest odpowiedź wyłącznie z jednego kontrolera, w sytuacji gdy zajętość kolejki mieści się w granicach pierwszego segmentu. Gdy wartość ta znajduje się w drugim segmencie, wtedy funkcja prawdopodobieństwa odrzucenia pakietu jest sumą odpowiedzi pierwszego i drugiego kontrolera. Z kolei w ostatnim - trzecim segmencie, sumowana jest odpowiedź z wszystkich trzech kontrolerów. Wartości stanowiące granicę poszczególnych segmentów wyznaczono empirycznie, na drodze przeprowadzonych eksperymentów badawczych. Wyniki zaproponowanego rozwiązania, bazującego na powyższych regułach, zestawiono z rezultatami uzyskanymi z wykorzystaniem standardowego mechanizmu AQM, który wykorzystuje pojedynczy kontroler PI^α do zarządzania transmisją.

Przeprowadzenie analizy uzyskanych wyników pozwoliło na sformułowanie kilku wniosków potwierdzających słuszność ścieżki obranej do badań. Po pierwsze, nawet w przypadku bardzo mocno obciążonej sieci, nie odnotowano odrzucania pakietów w wyniku przepełnienia bufora, niezależnie od stopnia samopodobieństwa ruchu, co jest kluczowe z punktu widzenia płynnej transmisji oraz zapobiegania problemom związanych z globalną synchronizacją w sieci Internet. Po drugie, zajętości kolejki bufora osiągają wszystkie wyznaczone segmenty, co z kolei wyraźnie wpływa na prędkość transmisji. Dzieje się tak, pomimo, że to dwa pierwsze kontrolery odpowiadają za odrzucanie większości pakietów. Z drugiej strony w przypadku bardzo zmiennego ruchu, warto zwrócić uwagę na niewielką reakcję pierwszego i drugiego kontrolera PI^α . W przypadku ruchu cechującego się bardzo dużym stopniem zależności długoterminowych ($H = 0.9$), najwięcej pakietów odrzucanych jest przez funkcję opartą o sumę wszystkich trzech kontrolerów PI^α . Zjawisko to spowodowane jest dużą zmiennością zajętości kolejki, która zwiększa się wraz ze wzrostem parametru Hursta.

Otrzymane rezultaty dowodzą tezie, stanowiącej że zastosowanie adaptacyjnego modelu mechanizmu AQM, reagującego na zmiany zajętości kolejki bufora, które wynikają ze zmiennej intensywności ruchu oraz cech jego samopodobieństwa, pozwala zwiększyć wydajność transmisji w sieci Internet. Badania potwierdziły istotny wpływ stopnia samopodobieństwa ruchu, wyrażonego za pomocą

parametru Hursta, na otrzymywane wartości średniej zajętości kolejki oraz liczbę odrzuconych pakietów.

Podczas gdy w rozpatrywanych przypadkach natężenie ruchu sieciowego było bardzo duże, większość pakietów odrzucana była z powodu przekroczenia maksymalnego rozmiaru kolejki. Natomiast wykorzystanie wyłącznie trzeciego kontrolera powodowało zbyt gwałtowną reakcję, która skutkowałą odrzucaniem zbyt dużej liczby pakietów. Podobnie zresztą, jak w przypadku średniego oraz niskiego obciążenia sieci, które to przypadki również charakteryzowały się stosunkowo dużymi stratami oraz słabym wykorzystaniem pasma transmisyjnego. Z drugiej strony adaptacyjny charakter zaprezentowanego modelu pozwolił dostosować sposób zarządzania buforem transmisyjnym w zależności od aktualnie napotkanych warunków w sieci Internet, umożliwił wyeliminowanie strat wynikających z przepełnienia kolejki oraz stworzył możliwość optymalnego wykorzystania dostępnego pasma transmisyjnego.

Wkład autorski [60%] polegał na:

- współudziale w przeglądzie literatury oraz analizie stanu wiedzy,
- zaproponowaniu mechanizmu aktywnego zarządzania kolejką z funkcją prawdopodobieństwa odrzucenia pakietu w oparciu o odpowiedź z kilku kontrolerów PI^α ,
- implementacji modelu wykorzystanego w badaniach,
- przeprowadzeniu eksperymentów badawczych,
- analizie uzyskanych wyników oraz sformułowaniu wniosków,
- pełnieniu funkcji autora korespondującego.

2.2 Publikacja 2: Dobór parametrów mechanizmu AQM w oparciu o techniki uczenia maszynowego oraz uczenia przez wzmocnienie

Celem badań przedstawionych w publikacji "AQM mechanism with neuron tuning parameters" było zaproponowanie rozwiązania decyzyjnego opartego

o mechanizmy aktywnego zarządzania kolejką, mającego za zadanie kontrolowanie przeciążeń w sieci Internet. Zaplanowane badania miały wykazać, czy monitorowanie natężenia transmisji danych wraz z zastosowaniem aktywnego adaptacyjnego doboru parametrów wybranej modyfikacji algorytmu aktywnego zarządzania kolejką, pozwoli wydatnie zwiększyć wpływy pakietów do węzła, a co za tym idzie poprawić przepustowość transmisji. W niniejszej pracy zaprezentowano algorytm modyfikujący współczynniki regulatora niecałkowitego rzędu PI^α . Zaimplementowane rozwiązanie zostało wykorzystane do stworzenia adaptacyjnego mechanizmu aktywnego zarządzania pakietami w węzle komunikacyjnym. Odpowiedź p regulatora PI^α wykorzystywana do obliczania prawdopodobieństwa prewencyjnego odrzucenia pakietu, wyznaczana jest za pomocą poniższej formuły:

$$p_i = \max\{0, -(K_P E_k + K_I \Delta^\alpha E_k)\} \quad (2.1)$$

gdzie: K_P oraz K_I to odpowiednio proporcjonalny i całkujący człon kontrolera, E_k to uchyb regulatora, czyli różnica pomiędzy aktualnym rozmiarem kolejki Q_k , a wartością oczekiwaną Q .

Zaproponowany mechanizm aktywnego zarządzania pakietami wykorzystuje neuron do doboru parametrów kontrolera, w zależności od aktualnych warunków ruchu sieciowego oraz jego natężenia. Sposób uczenia neuronu i wyznaczania wartości parametrów kontrolera przebiega w sposób następujący [32]:

$$K_P(t) = k_1 \frac{w_1(t)w_4(t)}{\sum_{i=1}^n w_i(t)} \quad (2.2)$$

$$K_I(t) = k_2 \frac{w_2(t)w_5(t)}{\sum_{i=1}^n w_i(t)} \quad (2.3)$$

gdzie: k_1 i k_2 są stałymi proporcjonalnymi współczynnikami.

W części eksperymentalnej przeprowadzono badania z wykorzystaniem różnych wartości parametru α . Intensywność źródła pakietów w symulacji przyjęła wartość stałą ($\lambda = 0.5$). Natomiast współczynnik czasu obsługi pakietu w kolejce ustalono kolejno dla μ : 0.25 - w celu uzyskania systemu mocno obciążonego, 0.5 - systemu średnio obciążonego oraz 0.75 - systemu słabo obciążonego. Badania przeprowadzono z uwzględnieniem czterech wartości parametru Hursta: $H = 0.5; 0.7; 0.8; 0.9$. Brano więc pod uwagę zarówno ruch niesamopodobny (gdy $H = 0.5$), jak również ruch cechujący się bardzo dużym stopniem zależności długoterminowych ($H = 0.9$). Ponadto zaproponowany mechanizm porównano z algorytmem Adaptive Neuron AQM, by określić jego zdolności adaptacyjne do

dostosowania się do aktualnych warunków w transmisji. Wcześniejsze prace pokazały, że mechanizm kontrolera PI^α bardzo dobrze nadaje się jako regulator średniej zajętości kolejki. Dość dużym problemem, podobnie jak w innych tego typu algorytmach jest prawidłowy dobór parametrów. Decyzja ta w znacznym stopniu może zależeć od właściwości ruchu sieciowego, takich jak intensywność ruchu czy zależności długoterminowe. Wyniki przeprowadzonych eksperymentów badawczych, w tym wpływ stopnia samopodobieństwa ruchu sieciowego na średnią długość kolejki, średni czas oczekiwania pakietu w buforze, liczbę pakietów odrzuconych przez zastosowany kontroler PI^α , a także zmianę wartości parametrów K_P oraz K_I przez zaimplementowany neuron przedstawiono za pomocą tabel i wykresów. Analiza uzyskanych rezultatów dowodzi, że możliwy jest adaptacyjny dobór parametrów kontrolera z wykorzystaniem technik uczenia maszynowego, sieci neuronowych oraz uczenia przez wzmocnienie. Zaproponowana metoda w odpowiedni sposób reguluje średnią zajętość kolejki, która w przeprowadzonych eksperymentach zbliżona była do wartości oczekiwanej. Z drugiej strony badania wykazały istotny wpływ stopnia samopodobieństwa ruchu, który wyrażono za pomocą parametru Hursta, na liczbę odrzuconych pakietów oraz średnią długość kolejki. Uzyskane wyniki są więc ściśle związane ze stopniem samopodobieństwa. W przypadku ruchu cechującego się dużym stopniem zależności długoterminowych (gdy $H = 0.9$), liczba odrzuconych pakietów znacznie wzrasta. To z kolei sprawia, że odnotowana wartość średniej długości kolejki jest poniżej zakładanego poziomu, czyli zmniejsza się wykorzystanie dostępnego łącza transmisji w sieci Internet. Rezultaty prac badawczych zawarte w niniejszej publikacji nagrodzone zostały wyróżnieniem: "BEST ICxS'2020 PAPER AWARD" podczas międzynarodowej konferencji naukowej Intelligent Information and Database Systems (ACIIDS 2020, 23-26 marzec 2020 Phuket, Tajlandia).

Wkład autorski [65%] związany był z:

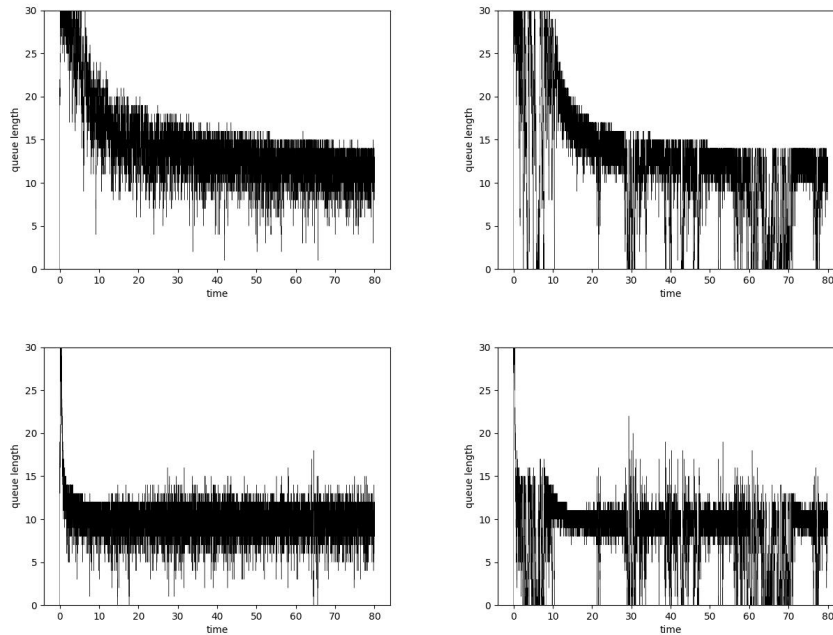
- współudziałem w analizie stanu wiedzy i wykonaniem przeglądu literatury,
- zaproponowaniem mechanizmu aktywnego zarządzania kolejką z wykorzystaniem technik uczenia maszynowego, w tym uczenia przez wzmocnienie,
- implementacją modelu wykorzystanego w eksperymentach symulacyjnych,
- współudziałem w przeprowadzonych badaniach modelu aktywnego zarządzania kolejką,
- analizą rezultatów, opracowaniem wniosków oraz zadaniami autora korespondującego.

2.3 Publikacja 3: Adaptacyjny mechanizm AQM dostosowujący się do intensywności ruchu oraz stopnia samopodobieństwa

W ramach prac przedstawionych w artykule "Adaptive Hurst-Sensitive Active Queue Management" zaproponowałem modyfikację podstawowego algorytmu Adaptive RED oraz algorytmów opartych o sieci neuronowe, które dostosowują się do intensywności oraz dodatkowo do stopnia samopodobieństwa strumienia sieciowego, wyrażonego za pomocą parametru Hursta. Metodologia badań zawarta w niniejszej publikacji stanowi rozwinięcie podejścia przedstawionego w pracy "AQM mechanism with neuron tuning parameters". W związku z tym, że operacje związane z obliczeniem wartości parametru Hursta są złożone obliczeniowo, a najpopularniejsze metody obliczania tego parametru są zbyt wolne aby można je było zastosować w rzeczywistym routerze, niniejszy artykuł przedstawia również modyfikację metody zagregowanej wariancji. Wykorzystuje ona pewne uproszczenia matematyczne, które pozwalają na wykonanie części obliczeń w tle. Wiadomości o każdym przychodzącym pakiecie gromadzone są w specjalnej strukturze, która to przechowuje informacje o liczbie pakietów w różnych skalach czasowych. Taki sposób wstępnego przetworzenia danych znacznie przyspiesza proces obliczania wartości parametru Hursta.

Zgodnie z założeniami przedstawionymi we wstępie niniejszej dysertacji, do oceny współpracy mechanizmów aktywnego zarządzania kolejką z protokołem TCP wykorzystano aproksymację Fluid-Flow (model pętli zamkniętej, ang. closed loop). Z drugiej strony dzięki zastosowaniu modelu pętli otwartej (ang. open loop), bazującego na metodach symulacyjnych, możliwe stało się dokładne przeanalizowanie specyfiki zachowania kolejki routera w przypadku ruchu o różnym natężeniu i stopniu samopodobieństwa. Do fazy badań eksperymentalnych wybrano następujące mechanizmy aktywnego zarządzania kolejką: ARED, ANRED, a także algorytmy niecałkowitego rzędu PI^α oraz $PI^\alpha D^\beta$. Dla wszystkich powyżej przytoczonych kontrolerów, zaproponowano nowe - analogiczne modele, które za pomocą struktur opartych o sieci neuronowe automatycznie dostosowują parametry mechanizmu aktywnego zarządzania kolejką, w zależności od intensywności ruchu oraz stopnia jego samopodobieństwa. Po zaimplementowaniu zaproponowanych modeli oraz przeprowadzeniu eksperymentów badawczych, uzyskano znaczącą poprawę parametrów transmisji, w porównaniu do analogicznych eksperymentów przeprowadzanych dla klasycznych rozwiązań. Wspomniana optymalizacja wyrażona była poprzez zmniejszenie wartości opóźnień transferu oraz

redukcji średniej zajętości kolejki, a także poprzez spadek sumarycznej liczby odrzuconych pakietów. Szczegółowa analiza uzyskanych wyników numerycznych pozwoliła zaobserwować zmniejszenie wartości opóźnień między 11.8% a 18.7%, w przypadku ruchu niesamopodobnego, bądź też ruchu o niskim stopniu zależności długoterminowych (odpowiednio dla wartości parametru Hursta = 0.5 oraz 0.6). Z drugiej strony dla ruchu cechującego się wysokim stopniem samopodobieństwa (dla parametru Hursta = 0.9) zanotowano spadek opóźnień między 14% a 16.1%. Podobnie do zanotowanych opóźnień zmieniała się również średnia zajętość kolejki. Wartości te w przypadku nowo zaproponowanego podejścia ulegały zmniejszeniu między 2.7% a 29%, gdy wartość parametru Hursta = 0.5 oraz między 11.8% a 18.7%, w przypadku gdy parametr Hursta = 0.9.



Rysunek 2.1: Wartości zajętości kolejki, $\mu = 0.25$ - w przypadku zastosowania klasycznego kontrolera $PI^\alpha D^\beta$, $\alpha = 0.5$, $H = 0.5$ (**lewy, górny**), $H = 0.9$ (**prawy, górny**) oraz uzyskane rezultaty dla zaproponowanego modelu kontrolera $PI^\alpha D^\beta$ dostosowującego się do intensywności oraz stopnia samopodobieństwa ruchu, $\alpha = 0.5$, $H = 0.5$ (**lewy, dolny**), $H = 0.9$ (**prawy, dolny**).

Najbardziej znaczące zyski optymalizacyjne zaobserwowano jednak w przypadku redukcji liczby retransmitowanych pakietów. Dla kontrolerów niecałkowi-

tego rzędu PI^α oraz $PI^\alpha D^\beta$ sumaryczna liczba odrzuconych pakietów na skutek przepełnienia kolejki zmniejszała się między 86% a 92% w przypadku ruchu bez zależności długoterminowych. Natomiast dla ruchu cechującego się dużym stopniem zależności długoterminowych uzyskana optymalizacja mieściła się w zakresie od 81.5% do 85.5%. Na rysunku 2.1 przedstawiono szczegółowe porównanie uzyskanych rezultatów zajętości kolejki w przypadku zastosowania klasycznego kontrolera $PI^\alpha D^\beta$ oraz dla zaproponowanego modelu kontrolera $PI^\alpha D^\beta$, dostosowującego się do intensywności oraz stopnia samopodobieństwa ruchu. Z drugiej strony warto zwrócić uwagę, że w przypadku mechanizmu ANRED wdrożenie modyfikacji pozwalającej na dostosowanie się do stopnia samopodobieństwa ruchu sieciowego nie wpłynęło znacząco na uzyskane wyniki. Mechanizm ten bowiem w każdym z rozpatrywanych przypadków cechował się zbyt dużą intensywnością reagowania na napotkane zmiany. Rezultaty niniejszej publikacji dowodzą, że stworzony mechanizm dostosowujący się do zmiennego stopnia samopodobieństwa ruchu pozwala znacznie poprawić efektywność transmisji sieciowej.

Wkład autorski [50%] w powstanie tego artykułu polegał na:

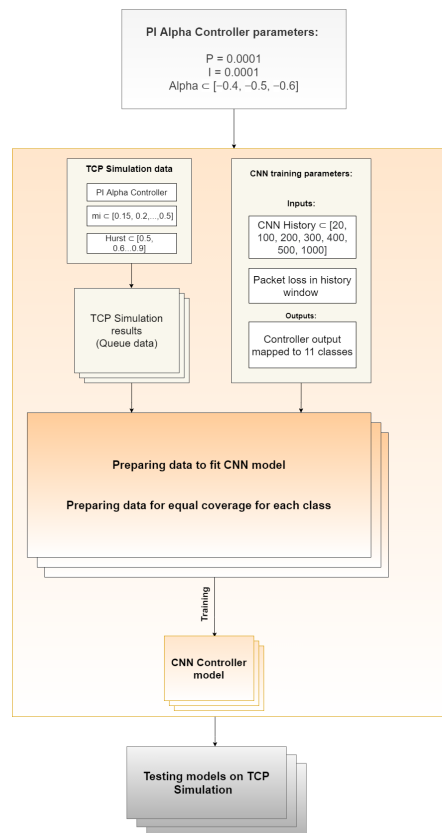
- współdziałanie w analizie stanu wiedzy i wykonaniu przeglądu literatury,
- zaproponowaniu i zaimplementowaniu modelu AQM, który dostosowuje się do stopnia samopodobieństwa oraz do intensywności ruchu,
- współdziałanie w przeprowadzaniu eksperymentów badawczych i analizie otrzymanych wyników,
- wykorzystaniu aproksymacji Fluid-Flow do oceny współpracy zaproponowanych mechanizmów AQM z protokołem TCP,
- współdziałanie w sformułowaniu wniosków i pełnieniu roli autora korespondującego.

2.4 Publikacja 4: Adaptacyjny mechanizm aktywnego zarządzania kolejką oparty o metody uczenia nadzorowanego

Publikacja "Supervised Learning of Neural Networks for Active Queue Management in the Internet" przedstawia model mechanizmu aktywnego zarządzania kolejką, który stworzono z wykorzystaniem sieci neuronowych oraz metod uczenia nadzorowanego. W celu przeprowadzenia procesu nauki wybranych struktur sieci konwolucyjnych wygenerowano dane uzyskane poprzez wykonanie eksperymentów symulacyjnych. Współczynnik natężenia ruchu wejściowego ustalono jako stałą wartość $\lambda = 0.5$. Źródło pakietów symulacji posiada więc zawsze stałą intensywność. Natomiast wartość parametru μ reprezentującego prawdopodobieństwo pobrania pakietu z kolejki wielokrotnie zmieniano. Przyjmował on wartości z zakresu od $\mu = 0.5$, dla systemu średnio obciążonego, do $\mu = 0.15$, czyli systemu bardzo mocno obciążonego. Dla każdego z wyżej wymienionych przypadków uwzględniono również zmienny stopień samopodobieństwa ruchu. Wartości te zmieniano od $H = 0.5$, co reprezentowało ruch niesamopodobny, do $H = 0.9$, czyli ruch cechujący się bardzo dużym stopniem zależności długoterminowych. Opisana w ten sposób charakterystyka ruchu sieciowego z wykorzystaniem metod symulacyjnych pozwoliła odwzorować wszystkie cechy wybranego mechanizmu aktywnego zarządzania kolejką. W tym przypadku zdecydowano się z kolei na kontroler niecałkowitego rzędu PI^α . Wartości parametrów kontrolera niecałkowitego rzędu PI^α przedstawiono w tabeli (2.1). Wartości te ustalono na podstawie wcześniejszych prac [33]. Jak wykazały wyniki wcześniej przeprowadzonych badań, dobór parametrów kontrolera znacząco wpływa na właściwości regulowania długości kolejki. Parametry kontrolerów dobrano w ten sposób, że kontroler $PI^{\alpha 1}$ jest kontrolerem najsłabszym, a kontroler $PI^{\alpha 3}$ jest mechanizmem najmocniejszym, co skutkuje dużą liczbę odrzuconych pakietów.

	K_P	K_I	α
$PI^{\alpha 1}$	0.0001	0.0004	-0.4
$PI^{\alpha 2}$	0.0001	0.0004	-0.5
$PI^{\alpha 3}$	0.0001	0.0004	-0.6

Tablica 2.1: Wartości parametrów kontrolera niecałkowitego rzędu PI^α



Rysunek 2.2: Schemat przedstawiający sposób przetwarzania danych w celu przygotowania zbioru uczącego.

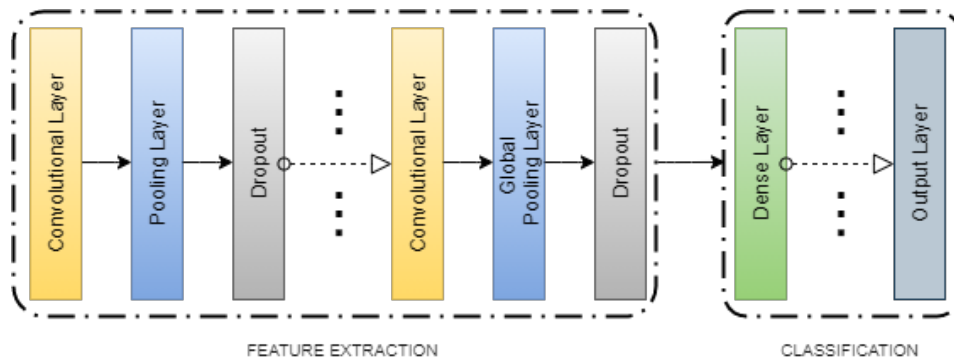
Dane uczące sieci neuronowej składają się z:

1. Danych wejściowych

- (a) Ostatnich n elementów z historii zajętości kolejki
- (b) Historii odrzucenia pakietów w n ostatnich stanach kolejki, gdzie $n \in [20; 100; 200; 300; 400; 500; 1000]$

2. Danych wyjściowych

- (a) 11 etykiet mapujących prawdopodobieństwo odrzucenia pakietu, w zależności od aktualnie występujących warunków w transmisji

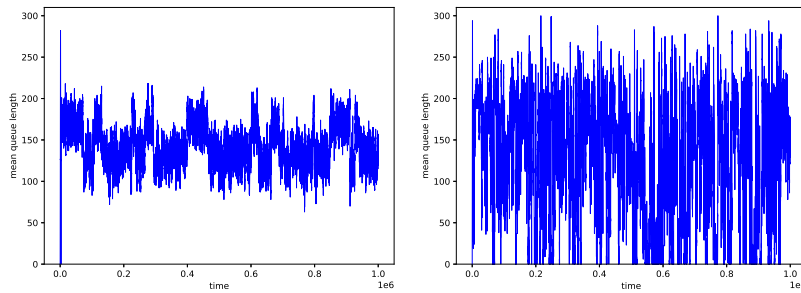


Rysunek 2.3: Koncepcja struktury modelu konwolucyjnej sieci neuronowej.

Dane uczące modelu konwolucyjnych sieci neuronowych odzwierciedlają zachowanie mechanizmu AQM. W przypadku każdego przedziału prawdopodobieństwa, przygotowano milion jednowymiarowych rekordów z danymi uczącymi. Cały zbiór uczący składa się więc z 11 milionów rekordów. Zawarte są w nich informacje dotyczące transmisji, takie jak długość kolejki w każdym kolejnym slotcie czasowym, liczba pakietów odrzuconych oraz wartość funkcji prawdopodobieństwa odrzucenia pakietu przez mechanizm aktywnego zarządzania kolejką. Metody wykorzystane w celu przetworzenia danych i przygotowania zbioru uczącego przedstawiono na rysunku (2.2). W czasie badań rozpatrywano sieci neuronowe z różną liczbą warstw konwolucyjnych, a także stosowano różne optymalizatory oraz funkcje kosztu, w celu optymalizacji tworzonego modelu AQM. Na podjęte w niniejszej pracy decyzje wpłynęły też wcześniejsze doświadczenia z prac nad klasyfikowaniem ruchu sieciowego [34], które to również potwierdziły stosowność wykorzystania konwolucyjnych sieci neuronowych w rozpatrywanej w tej pracy tematyce. Ostatecznie po przeprowadzeniu badań empirycznych zdecydowano na oparcie modelu o cztery warstwy konwolucyjne oraz dwie warstwy gęste, w których to zastosowano funkcję Relu jako funkcję aktywacji. Jest ona stosowana do obliczania wartości wyjścia z neuronu. Z drugiej strony jako funkcję aktywacji ostatniej warstwy wybrano Sigmoid, w przypadku której uzyskiwano największą wartość wskaźnika precyzji uczenia. Jest to wybór o tyle interesujący, gdyż pomimo faktu, że w literaturze [35] częściej spotykana jest funkcja Softmax w ostatniej warstwie sieci neuronowej, to w rozpatrywanym przypadku lepiej sprawdza się właśnie funkcja Sigmoid. W najgorszym przypadku, gdy sieć która uzyskiwała precyzję uczenia na poziomie 32,31%, po zmianie funkcji akty-

wacji na Sigmoid wartość ta zwiększała się do 65,65%. Natomiast w najlepszym uzyskanym przypadku wskaźnik precyzji uczenia wzrastał z 58,90% do 86,65%. Funkcja kosztu zaprojektowanej sieci neuronowej to categorical cross-entropy. Natomiast w celu wytrenowania modelu i zminimalizowania wpływu funkcji kosztu, wykorzystano optymalizator Adaptive Moment Estimation (Adam). Rysunek (2.3) ilustruje strukturę sieci neuronowej zaproponowanej jako model mechanizmu aktywnego zarządzania kolejką.

Proces uczenia przygotowanych modeli trwał 10 epok. Była to wartość zupełnie wystarczająca, gdyż wskaźniki wartości precyzji uczenia zaczynały stabilizować się już po 5-6 epokach. W trakcie badań szczególną uwagę zwrócono na wpływ stopnia samopodobieństwa ruchu oraz zależności długoterminowych na wydajność zaproponowanego mechanizmu. Bardzo ważnym aspektem było również wyselekcjonowanie liczby ostatnich n elementów historii zajętości kolejki, które stosowane są jako dane wejściowe zaproponowanego modelu. Zagadnienie to jest szczególnie istotne z punktu widzenia wydajności routera, które jest urządzeniem o niewielkich zasobach. Z tego powodu każda możliwość zmniejszenia rozpatrywanej liczby elementów historii jest sytuacją bardzo korzystną. Po przygotowaniu danych uczących, stworzeniu struktury sieci neuronowej oraz wytrenowaniu modelu, przystąpiono do fazy eksperymentów badawczych mających na celu wykorzystanie zaproponowanego adaptacyjnego modelu mechanizmu aktywnego zarządzania kolejką w warunkach symulacyjnych, a następnie porównania otrzymanych rezultatów z wynikami uzyskiwanymi w przypadku klasycznego kontrolera. W czasie trwania wspomnianego etapu ewaluacji oceniano cztery odrębne modele sieci neuronowych. Pierwsze trzy modele sieci neuronowych wyuczono zgodnie z zachowaniem kontrolerów $PI^{\alpha 1}$, $PI^{\alpha 2}$ i $PI^{\alpha 3}$. Czwarty model wytrenowano danymi reprezentującymi zachowanie trzech kontrolerów równocześnie. Szczegółowa analiza wyników z przeprowadzonych eksperymentów badawczych pozwala stwierdzić, że dla wszystkich trzech modeli aktywnego zarządzania kolejką opartych o sieci neuronowe, uzyskano lepsze rezultaty, a co za tym idzie bardziej efektywny mechanizm od standardowych rozwiązań. Nawet w przypadku modelu, który osiągnął najmniejszą wartość precyzji uczenia się - uzyskana w eksperymentach średnia długość kolejki jest większa niż dla klasycznego mechanizmu $PI^{\alpha 3}$. Sytuacja taka zachodzi zarówno dla ruchu niesamopodobnego (gdy wartość parametru $H = 0.5$), jak również w przypadku ruchu cechującego się dużym stopniem zależności długoterminowych. Jednak jako najbardziej efektywny mechanizm aktywnego zarządzania kolejką wybrano model sieci konwolucyjnych, wyuczony zachowaniem wszystkich trzech kontrolerów PI^{α} . Kolejnym istotnym wnioskiem otrzymanym po analizie uzyskanych wyników jest również fakt, że



Rysunek 2.4: Rezultaty zajętości kolejki uzyskane dla modelu wyuczonego z wykorzystaniem danych z trzech kontrolerów PI^α , dla $n = 500$ elementów historii zajętości kolejki, $H = 0.5$ (lewy), $H = 0.9$ (prawy).

wraz ze wzrostem stosowanej liczby elementów historii zajętości kolejki (n), zwiększa się również zdolność adaptacji mechanizmu do aktualnych warunków transmisji w sieci Internet. Najlepsze wyniki uzyskiwano gdy wartość n wynosiła 500 (rysunek 2.4). Z drugiej strony, minimalna akceptowalna liczba zastosowanych elementów historii zajętości kolejki, dla której uzyskano dopuszczalne rezultaty, to $n = 100$. Wyniki przeprowadzonych badań potwierdziły, że model oparty o konwolucyjne sieci neuronowe potrafi skutecznie odwzorować wyniki klasycznego algorytmu AQM oraz efektywnie zarządzać transmisją danych. Utrzymuje on założoną średnią liczbę pakietów w kolejce oraz zmniejsza całkowitą liczbę pakietów odrzuconych, niezależnie od stopnia samopodobieństwa ruchu.

Wkład autorski [65%] w powstanie tego artykułu związany był z:

- współudziałem w przeglądzie literatury,
- zaproponowaniem modelu mechanizmu aktywnego zarządzania kolejką w oparciu o wykorzystanie technik uczenia nadzorowanego i sieci neuronowych,
- zaimplementowaniem modelu wykorzystanego w badaniach,
- współudziałem w procesie przygotowania zbioru uczącego,
- przeprowadzeniem eksperymentów badawczych,
- analizą rezultatów i sformułowaniem wniosków,
- pełnieniem funkcji autora korespondującego.

2.5 Publikacja 5: Model mechanizmu AQM dla priorytetowych danych ruchu urządzeń Internetu Rzeczy

Publikacja "The IoT gateway with active queue management" przedstawia adaptacyjny mechanizm aktywnego zarządzania kolejką, który pozwala wyeliminować opóźnienia w przypadku transmisji priorytetowych pakietów ruchu urządzeń Internetu Rzeczy (ang. Internet of Things). Model ten umożliwia transfer zarówno priorytetowych, jak również wszystkich pozostałych pakietów z wykorzystaniem jednego bufora transmisyjnego. Ma więc stanowić ekonomiczne i wartościowe rozwiązanie, zwłaszcza z punktu widzenia niezwykle dynamicznie rozwijającego się rynku urządzeń Internetu Rzeczy [36, 37]. Stworzony mechanizm dedykowany jest dla bramy urządzeń Internetu Rzeczy, gdzie poprzez zastosowanie odpowiednich metod umożliwiających oznaczenie pakietów priorytetowych (odpowiednio pole Type of Service w nagłówku IPv4 oraz Traffic Class w IPv6), a następnie za pomocą technik klasyfikacji danych, umożliwić ma spełnienie rygorystycznych wymogów jakości usług transmisji w sieci Internet (ang. Quality of Service). By lepiej zilustrować zasady zaproponowanej koncepcji, można przyjąć, że zadaniem stworzonego mechanizmu jest zarządzanie pakietami o standardowym poziomie uprzywilejowania w taki sposób, by pasmo transmisyjne zawsze zapewniało możliwość przesyłu pakietów priorytetowych zgodnie z regułami FIFO. Kryterium to zapobiega ryzyku retransmisji krytycznych danych oraz wystąpienia opóźnień w transmisji, a przez to pozwala spełnić umowy o gwarantowanym poziomie świadczenia usług między usługodawcą a klientem. W czasie przeprowadzania eksperymentów badawczych rozpatrywano trzy różne zestrojone mechanizmy aktywnego zarządzania kolejką, oparte o kontroler niecałkowitego rzędu PI^α . Jego parametry do badań dobierano w taki sposób, by mechanizm cechował się kolejno bardzo szybką, szybką oraz wolną reakcją na zmiany zachodzące w buforze. Intensywność źródła pakietów przyjęła wartość stałą ($\lambda = 0.5$). Prawdopodobieństwo pobrania pakietu z kolejki, czyli czas obsługi danych, wyrażony za pomocą parametru μ , przyjmował wartości od 0.2 do 0.8, co pozwoliło odzwierciedlić zarówno bardzo mocno, jak również bardzo słabo obciążone pasmo transmisyjne. W badaniach szczególną uwagę zwrócono na wpływ stopnia samopodobieństwa ruchu na końcowe rezultaty. Stąd wielokrotnie zmieniane wartości parametru Hursta (0.5 - 0.9) reprezentowały zarówno ruch niesamopodobny, jak również ruch cechujący się bardzo dużym stopniem zależności długoterminowych. Ponadto we wszystkich powyżej rozpatrywanych przypadkach zmieniano pro-

centowy udział pakietów priorytetowych w paśmie transmisyjnym, chcąc określić ich maksymalną dopuszczalną wartość. W tym przypadku głównym kryterium był brak strat pakietów priorytetowych. Natomiast do oceny wpływu zaproponowanego mechanizmu na pojedynczy strumień TCP wykorzystano aproksymację Fluid-Flow. Analiza uzyskanych rezultatów dowiodła, że w zależności od doboru zestawów parametrów kontrolera PI^α , przedstawione rozwiązanie można elastycznie dostosowywać do zmiennego udziału pakietów priorytetowych. Jednak ich maksymalna wartość wyraźnie maleje wraz ze wzrostem stopnia zależności długoterminowych ruchu. Co jednak istotne z punktu widzenia praktyczności zaproponowanego modelu, sprawdza się on nawet w przypadku bardzo mocno obciążonej sieci. Brak utraty pakietów priorytetowych, przy bardzo intensywnym odrzucaniu pozostałych danych, pozwala wysnuć stwierdzenie, że przedstawiony mechanizm może być również bardzo wartościowy w przypadku prób odpiernia ataków DDoS (ang. Distributed Denial of Service), polegających na zajęciu wszystkich wolnych zasobów (a tym samym zablokowaniu dostępu do łączy), poprzez atak przeprowadzany jednocześnie z wielu komputerów [38].

Wkład autorski [60 %] polegał na:

- współudziale w analizie stanu wiedzy,
- przedstawieniu koncepcji modelu aktywnego zarządzania kolejką, który pozwala oznaczyć pakiety priorytetowe oraz zapewnić ich transmisję w krytycznym czasie,
- implementacji modelu badawczego,
- przeprowadzeniu eksperymentów badawczych,
- wykorzystaniu aproksymacji Fluid-Flow do oceny wpływu zaproponowanego modelu na pojedynczy strumień TCP,
- współudziale w analizie wyników i sformułowaniu wniosków,
- pełnieniu roli autora korespondującego.

Rozdział 3

Podsumowanie

Analiza uzyskanych wyników z przeprowadzonych eksperymentów badawczych, które w trakcie prac nad niniejszą dysertacją opublikowano w uznanych czasopismach naukowych oraz które przedstawiono na międzynarodowych konferencjach naukowych, dowodzi prawdziwości postawionej w tej pracy tezie, którą udowodniono poniżej:

Wniosek. *Stworzenie adaptacyjnych mechanizmów zarządzania pakietami, dostosowujących się nie tylko do natężenia ruchu sieciowego, ale również do zależności długoterminowych, może znacząco zwiększyć efektywność transmisji w sieciach komputerowych.*

Jednakże, pomimo zaproponowanych uproszczeń, wprowadzonych w celu ograniczenia liczby wykonywanych operacji, adaptacyjne mechanizmy zarządzania kolejką przedstawione w niniejszej rozprawie są bez wątpienia bardziej złożone obliczeniowo, niż najprostsze algorytmy z rodziny RED. Z drugiej strony oferują one znaczący poziom optymalizacji parametrów transmisji oraz zarządzania zajętością kolejki bufora. Jestem jednak przekonany, że wraz z dalszym rozwojem technologii, zwiększaniem dostępności wysoko wydajnego sprzętu komputerowego oraz dostępem do coraz mocniejszych i wydajniejszych obliczeniowo routerów, w niedalekiej przyszłości możliwe będzie pełne wykorzystanie rozwiązań zaproponowanych w niniejszej rozprawie.

Kierunki dalszych prac badawczych powstały w pewien sposób samodzielnie i stanowią odpowiedź na sugestie otrzymane od recenzentów, które pojawiły się przy okazji procesu publikowania artykułów przedstawionych w niniejszej rozprawie. Pierwszym możliwym kierunkiem będzie implementacja zaproponowanych mechanizmów aktywnego zarządzania kolejką w środowisku Linux,

a także w routerach kompatybilnych z systemem OpenWrt. Praca ta umożliwi przeprowadzenie badań z wykorzystaniem rzeczywistego ruchu pakietów z sieci Internet, a także będzie stanowiła pierwszy krok do wdrożenia przedstawionych mechanizmów aktywnego zarządzania kolejką, co też pozwoli na ich praktyczne wykorzystanie.

W przyszłych pracach dostrzegam również potrzebę dalszej optymalizacji zaproponowanych rozwiązań, tak aby dostosować je do wydajności i możliwości dzisiejszych urządzeń sieciowych. Z drugiej jednak strony zaproponowane mechanizmy już teraz są możliwe do praktycznego wykorzystania. Wbrew pozorom to właśnie sam proces uczenia modelu opartego o sieci neuronowe jest najbardziej złożony obliczeniowo i wymaga dostępu do najbardziej wydajnego sprzętu. Jednak samo jego wykorzystanie nie jest już tak kosztowne, co niweluje obawy związane z wydajnością współcześnie dostępnych i powszechnie stosowanych urządzeń sieciowych.

Bibliografia

- [1] Pan, R., Prabhakar, B., Psounis, K.: CHOKe, A Stateless AQM scheme for Approximating fair Bandwidth Allocation. IEEE INFOCOM, s. 942-952, 2000.
- [2] Jacobson V.: Congestion Avoidance and Control. ACM SIGCOMM, Vol. 18(4), 1988.
- [3] Hellal, O. A., Altman, E.: Analysis of TCP Vegas and TCP Reno. Telecommunication Systems, 2000.
- [4] Sarolahti, P., Kojo, M., Raatikainen, K.: F-RTO: An Enhanced Recovery Algorithm for TCP Retransmission Timeouts. ACM SIGCOMM Computer Communication Review, 2003.
- [5] Gerla, M., Sanadidi, M. Y., Wang, R., Zanella, A.: TCP Westwood: Congestion Window Control Using Bandwidth Estimation. IEEE Global Telecommunications Conference, 2001.
- [6] Floyd, S., Jacobson, V.: Random early detection gateways for congestion avoidance. IEEE/ACM Transactions on Networking, Vol. 1(4), 1993. doi:10.1109/90.251892.
- [7] Hassan, M., Jain, R.: High Performance TCP/IP Networking. Pearson Education Inc., 2004.
- [8] Domańska, J., Domański, A., Augustyn, D., Klamka, J.: A RED modified weighted moving average for soft real-time application. International Journal of Applied Mathematics and Computer Science, Vol.24(3), 697-707, 2014. doi:10.2478/amcs-2014-0051.

- [9] Tan, L., Zhang, W., Peng, G., Chen, G.: Stability of TCP/RED systems in AQM routers. *IEEE Transactions on Automatic Control*, Vol. 51(8), s. 1393-1398, 2006.
- [10] Unal, H., Melchor-Aguilar, D., Ustebay, D., Niculescu, S.I., Ozbay, H.: Comparison of PI controllers designed for the delay model of TCP/AQM. *Computer Communications*, Vol. 36(10), s. 1225-1234, 2013.
- [11] Xu, Y.-D., Wang, Z.-Y, Wang, H.: ARED: a novel adaptive congestion controller. *International Conference on Machine Learning and Cybernetics*, Vol. 2, s. 708 - 714, 2005. doi:10.1109/ICMLC.2005.1527036.
- [12] Floyd, S., Gummadi, R., Shenker, S.: Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management, 2001.
- [13] Lin, D., Morris, R.: Dynamics of Random Early Detection. *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*. Association for Computing Machinery, s. 127–137, 1997. doi:10.1145/263105.263154.
- [14] Zacharewicz, A.: *Metody Analizy Długozasięgowej*. Hugo Steinhaus Center (1999-2002).
- [15] Borys P.: *Sztuczki karciane, wylewy Nilu i wykładnik Hursta*. Wydział Chemiczny, Wydawnictwo Politechniki Śląskiej, 2011.
- [16] Kiłyk, A., Wilimowska, Z.: Wykorzystanie wykładnika Hursta do prognozowania zmian cen na Giełdzie Papierów Wartościowych. *Innowacje w zarządzaniu i inżynierii produkcji*. Oficyna Wydawnicza Polskiego Towarzystwa Zarządzania Produkcją, Vol. 1, s. 455-463, 2016.
- [17] Domańska, J., Domański, A., Czachórski, T.: Estimating the Intensity of Long-Range Dependence in Real and Synthetic Traffic Traces. *International Conference on Computer Networks*, Springer International Publishing, Vol. 522, s. 11-22, 2015. doi:10.1007/978-3-319-19419-6_2.
- [18] Domańska, J., Domański, A., Czachórski, T., Klamka, J.: Self-similarity Traffic and AQM Mechanism Based on Non-integer Order $PI^\alpha D^\beta$ Controller. *Communications in Computer and Information Science*. Springer International Publishing, Vol. 718, s. 336-350, 2017. doi:10.1007/978-3-319-59767-6_27.

- [19] Willinger, W., Taqqu, M. S., Wilson, D. V.: Lessons from "On the Self-Similar Nature of Ethernet Traffic". SIGCOMM Comput. Commun. Rev. Association for Computing Machinery, Vol. 49(5), s. 56-62, 2019, New York, NY, USA. doi:10.1145/3371934.3371955.
- [20] Leland, W. E., Taqqu, M. S., Willinger, W., Wilson, D. V.: On the self-similar nature of Ethernet traffic. IEEE/ACM Transactions on Networking, Vol. 2 (1), s. 1-15, 1994. doi:10.1109/90.282603.
- [21] Paxson, V., Floyd, S.: Wide area traffic: the failure of Poisson modeling. IEEE/ACM Transactions on Networking, Vol. 3(3), s. 226-244, 1995.
- [22] Feldmann, A., Gilbert, A. C., Willinger, W., Kurtz, T. G.: The Changing Nature of Network Traffic: Scaling Phenomena. SIGCOMM Comput. Commun. Rev., Vol. 28(2), s. 5–29, 1998, New York, USA. doi:10.1145/279345.279346.
- [23] Suchacka, G., Dembczak, A.: Verification of Web Traffic Burstiness and Self-similarity for Multiple Online Stores. Information Systems Architecture and Technology: Proceedings of 38th International Conference on Information Systems Architecture and Technology (ISAT), Springer International Publishing, s. 305-314, 2018. doi:10.1007/978-3-319-67220-5_28.
- [24] CAIDA: Center for Applied Internet Data Analysis based at the University of California's San Diego Supercomputer Center. <https://www.caida.org/>
- [25] Kim, Y., Min, P.: On the prediction of average queueing delay with self-similar traffic. IEEE Global Telecommunications Conference, GLOBECOM '03, Vol. 5, s. 2987 – 2991, 2004. doi:10.1109/GLOCOM.2003.1258782.
- [26] Swami, N.; Bairwa, A.; Choudhary, M.: A LITERATURE SURVEY OF NETWORK SIMULATION TOOLS. International Conference On Communication & Computational Technologies by RIET, Proceeding ICCCT, Vol. 5, s. 206–208, 2017.
- [27] Borboruah, G.; Nandi, G.: A Study on Large Scale Network Simulators. International Journal of Computer Science and Information Technologies, Vol. 5, s. 7318–7322, 2014.

- [28] Dou, Y.; Liu, H.; Wei, L.; Chen, S.: Design and simulation of self-organizing network routing algorithm based on Q-learning. 21st Asia-Pacific Network Operations and Management Symposium (APNOMS), s. 357–360, 2020. doi:10.23919/APNOMS50412.2020.9237020.
- [29] Guo, X.; Guo, B.; Li, K.; Fan, C.; Yang, H.; Huang, S. A.: SDN-enabled Integrated Space-Ground Information Network Simulation Platform. 18th International Conference on Optical Communications and Networks (ICOON), s. 1–3. doi:10.1109/ICOON.2019.8934091.
- [30] Misra, V.; Gong, W.-B.; Towsley, D.: Fluid-based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED. ACM SIGCOMM, 2000. doi:10.1145/347059.347421.
- [31] Domański, A.: Wpływ mechanizmów protokołu TCP oraz algorytmów kolejkiowania na transmisję danych w sieci Internet. *Studia Informatica*, Vol. 38(1A), s. 3-245, 2017.
- [32] Ping, Y.D., Wang, N.: A PID controller with neuron tuning parameters for multi-model plants. *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.04EX826)*, Vol. 6, s. 3408 - 3411, 2004. doi:10.1109/ICMLC.2004.1380375.
- [33] Domański, A., Domańska, J., Czachórski, T., Klamka, J., Szyguła, J., Marek, D.: AQM mechanism with the dropping packet function based on the answer of several PI^α controllers. *Communications in Computer and Information Science*, Springer International Publishing, Vol. 1039, s. 400-412, 2019. doi:10.1007/978-3-030-21952-9_29.
- [34] Filus, K.; Domański, A.; Domańska, J.; Marek, D.; Szyguła, J. Long-Range Dependent Traffic Classification with Convolutional Neural Networks Based on Hurst Exponent Analysis. *Entropy*, vol. 22, 1159. doi:10.3390/e22101159.
- [35] Mercioni, M. A., and Holban, S.: The Most Used Activation Functions: Classic Versus Current. *International Conference on Development and Application Systems (DAS)*, s. 141-145, 2020. doi:10.1109/DAS49615.2020.9108942.
- [36] Stoica, I., Morris, R., Karger, D., Kaashoek, F. M., Balakrishnan, H.: Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications.

SIGCOMM '01 Proceedings of the 2001 conference on Applications, Technologies, Architectures and Protocols for Computer Communications, Vol. 31, s. 149-160, 2001. doi:10.1145/383059.383071

[37] Miao, D., Liu, L., Xu, R., Panneerselvam, J., Wu, Y., Xu, W.: An Efficient Indexing Model for the Fog Layer of Industrial Internet of Things. IEEE Transactions on Industrial Informatics, Vol. 14(10), s. 4487-4496, 2018. doi:10.1109/TII.2018.2799598.

[38] Geng, X., Whinston, A.B.: Defeating distributed denial of service attacks. IT Professional, Vol. 2(4), s. 36-42, 2000. doi:10.1109/6294.869381.