



SDN ASOSIDA TRAFIKNI DINAMIK BOSHQARISH ALGORITMLARINI TADQIQ QILISH

Ilmiy rahbar: Akmuradov B.U.

PhD, UMFT universiteti “K va AT” kafedra muduri

Xamirayev Sunnatullo Xojixonovich

University of management and future technologies magistranti

sunnatulloxamirayev1381@gmail.com

tel.: 949531381

Annotatsiya. Mazkur maqolada SDN — Software-Defined Networking — asosida trafikni dinamik boshqarish algoritmlarining nazariy asoslari, ishlash mexanizmlari va samaradorlik ko'rsatkichlari tadqiq qilinadi. SDN arxitekturasi boshqaruv tekisligi va ma'lumot uzatish tekisligini ajratish orqali tarmoqni markazlashgan, dasturlashtiriladigan va moslashuvchan boshqarish imkonini beradi. Maqolada Dijkstra algoritmi, maksimal oqim modeli, yuklamani balanslash, QoSga asoslangan marshrutlash, adaptiv trafik boshqaruvi va real vaqt monitoringi mexanizmlari ko'rib chiqiladi. Shuningdek, kechikish, o'tkazuvchanlik, paket yo'qotish, jitter va kanal yuklanishi kabi mezonlar asosida algoritmlarni baholash yondashuvlari yoritiladi.

Kalit so'zlar: SDN, Software-Defined Networking, trafikni boshqarish, dinamik marshrutlash, SDN-kontroller, OpenFlow, QoS, Dijkstra algoritmi, maksimal oqim, yuklamani balanslash, tarmoq monitoringi.

ANALYSIS OF DYNAMIC TRAFFIC MANAGEMENT ALGORITHMS BASED ON SDN

Abstract. This article analyzes the theoretical foundations, operating mechanisms, and performance indicators of dynamic traffic management algorithms based on Software-Defined Networking (SDN). SDN architecture separates the control plane from the data forwarding plane, enabling centralized, programmable, and flexible network management. The article examines



Dijkstra's algorithm, the maximum flow model, load balancing, QoS-based routing, adaptive traffic control, and real-time monitoring mechanisms. It also discusses approaches to evaluating algorithms based on latency, throughput, packet loss, jitter, and link utilization.

Keywords: SDN, Software-Defined Networking, traffic management, dynamic routing, SDN controller, OpenFlow, QoS, Dijkstra algorithm, maximum flow, load balancing, network monitoring.

АНАЛИЗ АЛГОРИТМОВ ДИНАМИЧЕСКОГО УПРАВЛЕНИЯ ТРАФИКОМ НА ОСНОВЕ SDN

Аннотация. В статье анализируются теоретические основы, механизмы функционирования и показатели эффективности алгоритмов динамического управления трафиком на основе SDN — Software-Defined Networking. Архитектура SDN разделяет плоскость управления и плоскость передачи данных, что позволяет централизованно, программируемо и гибко управлять сетью. Рассматриваются алгоритм Дейкстры, модель максимального потока, балансировка нагрузки, маршрутизация на основе QoS, адаптивное управление трафиком и механизмы мониторинга в реальном времени. Также освещаются подходы к оценке алгоритмов на основе задержки, пропускной способности, потерь пакетов, джиттера и загрузки каналов.

Ключевые слова: SDN, Software-Defined Networking, управление трафиком, динамическая маршрутизация, SDN-контроллер, OpenFlow, QoS, алгоритм Дейкстры, максимальный поток, балансировка нагрузки, мониторинг сети.

Kirish. Zamonaviy kompyuter tarmoqlarida trafik hajmi va xizmat turlari keskin oshib bormoqda. Bulutli hisoblash, videokonferensiyalar, elektron ta'lim,



IoT qurilmalari, ma'lumotlar markazlari, mobil ilovalar va real vaqt xizmatlari tarmoqdan yuqori o'tkazuvchanlik, past kechikish, kam paket yo'qotish va barqaror xizmat sifatini talab qiladi. An'anaviy tarmoqlarda marshrutlash qarorlari ko'pincha har bir router yoki switch darajasida alohida qabul qilinadi. Bu holat tarmoqni global holat bo'yicha boshqarish, yuklamani tez muvozanatlash va trafik oqimlarini real vaqt sharoitiga moslashtirish imkoniyatini cheklaydi.

SDN texnologiyasi ushbu muammoni hal qilishga qaratilgan zamonaviy tarmoq arxitekturasi sifatida shakllandi. SDNda boshqaruv tekisligi va ma'lumot uzatish tekisligi ajratiladi. Natijada tarmoq qurilmalari oddiy forwarding funksiyasini bajaradi, marshrutlash va siyosat qarorlari esa markazlashgan SDN-kontroller tomonidan belgilanadi. Bu yondashuv tarmoqni dasturiy boshqarish, trafikni dinamik qayta taqsimlash va xizmat sifatini moslashuvchan boshqarish imkonini beradi. SDN asosida trafikni dinamik boshqarish algoritmlarini tahlil qilish tarmoq samaradorligini oshirish, kanallardagi tiqilishni kamaytirish, real vaqt xizmatlarining barqarorligini ta'minlash va resurslardan oqilona foydalanish uchun zarurdir. Ayniqsa ma'lumotlar markazlari, universitet tarmoqlari, korporativ tarmoqlar, 5G/6G transport infratuzilmalari va IoT tizimlarida bunday algoritmlar katta amaliy ahamiyat kasb etadi.

Asosiy qism. SDN arxitekturasi tarmoq boshqaruvining yangi modelini taklif etadi. An'anaviy tarmoqlarda boshqaruv va ma'lumot uzatish funksiyalari bir qurilma ichida joylashgan bo'ladi. SDNda esa boshqaruv tekisligi alohida kontroller darajasiga ko'tariladi, ma'lumot uzatish tekisligi esa paketlarni uzatish vazifasini bajaradi. Bu yondashuv tarmoqni markazlashgan tarzda nazorat qilish, global topologiyani ko'rish va oqimlar ustidan dasturiy siyosat yuritish imkonini beradi [1; 2].

SDN tizimi uch asosiy qatlamdan iborat: infratuzilma qatlami, boshqaruv qatlami va ilovalar qatlami. Infratuzilma qatlamida switchlar, routerlar va hostlar



joylashadi. Boshqaruv qatlamida SDN-kontroller tarmoq topologiyasi, link holati va trafik oqimlarini boshqaradi. Ilovalar qatlamida esa QoS, xavfsizlik, monitoring, yuklamani balanslash va trafik muhandisligi kabi funksiyalar amalga oshiriladi [1; 3].

OpenFlow SDN arxitekturasida kontroller va switch o'rtasidagi muhim protokollardan biri hisoblanadi. OpenFlow switch oqim jadvallari asosida ishlaydi, ya'ni kelgan paketlar jadvaldagi qoidalarga mos ravishda uzatiladi, o'zgartiriladi, tashlab yuboriladi yoki kontrollerga yuboriladi [3; 9]. Bu mexanizm trafikni dinamik boshqarish algoritmlarini amalda qo'llash uchun texnik asos yaratadi.

SDN asosida trafikni dinamik boshqarish algoritmlarini tahlil qilishda tarmoq odatda graf ko'rinishida ifodalanadi:

$$G = (V, E)$$

Bu yerda V — tarmoq tugunlari, E — tugunlar orasidagi kanallar majmuasi. Har bir kanal uchun sig'im, joriy yuklama, kechikish, paket yo'qotish ehtimoli va jitter kabi ko'rsatkichlar belgilanadi. Dinamik algoritmlar ana shu parametrlarni real vaqt rejimida hisobga olib, trafik oqimlari uchun eng maqbul marshrutni tanlaydi [5].

Kanal yuklanish darajasi quyidagicha aniqlanadi:

$$U_e = \frac{L_e}{C_e}$$

Bu yerda U_e — kanal yuklanish koeffitsiyenti, L_e — joriy trafik hajmi, C_e — kanal sig'imi. Agar U_e yuqori bo'lsa, kanal tiqilgan deb baholanadi. Masalan, $U_e > 0.8$ bo'lsa, algoritm trafikni boshqa alternativ marshrutga o'tkazishi mumkin. Bu yondashuv yuklamani muvozanatlash va paketlar navbatini kamaytirishga xizmat qiladi.



Dinamik boshqaruv algoritmlarining umumiy ishlash tartibi bir necha bosqichdan iborat. Dastlab SDN-kontroller switchlardan statistik ma'lumotlarni oladi. Keyin linklar bo'yicha yuklama, kechikish, yo'qotish va jitter hisoblanadi. Shundan so'ng har bir link uchun og'irlik koeffitsiyenti yangilanadi. Trafik oqimlari sinflarga ajratilib, har bir oqim uchun optimal marshrut tanlanadi. Yakuniy bosqichda kontroller OpenFlow qoidalarini switchlarga yuboradi. Tarmoq holati o'zgartirganda marshrut qayta hisoblanadi.

Trafikni dinamik boshqarishda eng muhim masalalardan biri kanal og'irligini to'g'ri hisoblashdir. An'anaviy eng qisqa yo'l algoritmlarida yo'l uzunligi yoki hop soni asosiy mezon bo'lishi mumkin. SDN muhitida esa real tarmoq holati hisobga olinishi kerak. Shu sababli kanal og'irligi bir nechta ko'rsatkich asosida aniqlanadi:

$$W_e = \alpha D_e + \beta U_e + \gamma P_e + \delta J_e$$

Bu yerda W_e — kanal og'irligi, D_e — kechikish, U_e — yuklanish koeffitsiyenti, P_e — paket yo'qotish ehtimoli, J_e — jitter, $\alpha, \beta, \gamma, \delta$ esa ko'rsatkichlarning ahamiyat koeffitsiyentlari. Ushbu model trafikni faqat eng qisqa yo'l bo'yicha emas, balki eng sifatli yo'l orqali uzatishga imkon beradi.

Dijkstra algoritmi SDN asosidagi dinamik marshrutlashda keng tahlil qilinadigan algoritmlardan biridir. Klassik Dijkstra algoritmi ijobiy og'irliklarga ega grafda manba tugundan boshqa tugunlarga eng qisqa yo'llarni topish uchun ishlab chiqilgan [6]. SDN muhitida ushbu algoritmning og'irliklari real vaqt yuklama, kechikish va yo'qotish ko'rsatkichlari bilan yangilanadi. Natijada statik Dijkstra adaptiv marshrutlash vositasiga aylanadi.

Dijkstra algoritmining asosiy afzalligi uning tushunarli, tez va amaliy tarmoqlarda qo'llashga qulayligidadir. U manba va qabul qiluvchi tugun o'rtasida eng kichik og'irlikka ega marshrutni topadi. Biroq algoritmning kamchiligi shundaki, u har doim ham yuklamani teng taqsimlay olmaydi. Agar barcha trafik



eng past og'irlikka ega yo'lga yo'naltirilsa, ushbu yo'l tezda tiqilib qolishi mumkin. Shu sababli SDNda Dijkstra algoritmi ko'pincha yuklama koeffitsiyenti va QoS mezonlari bilan boyitilgan holda ishlatiladi [5; 10].

Maksimal oqim algoritmlari SDN asosida trafikni dinamik boshqarishda katta hajmdagi ma'lumotlarni bir nechta yo'l bo'yicha taqsimlash uchun tahlil qilinadi. Ford–Fulkerson usuli manba va qabul qiluvchi o'rtasidagi maksimal oqimni aniqlashga xizmat qiladi [7]. Bu yondashuv tarmoqdagi mavjud sig'imdan maksimal foydalanish imkonini beradi.

Maksimal oqim modeli quyidagi shartlarga asoslanadi:

$$0 \leq f_e \leq C_e$$
$$\sum_{e \in out(v)} f_e - \sum_{e \in in(v)} f_e = 0, v \neq s, t$$

Bu yerda f_e — kanal orqali uzatilayotgan oqim, C_e — kanal sig'imi, s — manba, t — qabul qiluvchi tugun. Birinchi shart kanal sig'imidan oshmaslikni, ikkinchi shart esa oraliq tugunlarda oqim saqlanishini bildiradi.

Yuklamani balanslash algoritmlari SDN tarmoqlarida muhim ahamiyatga ega. Bunda kontroller trafikni faqat bitta marshrutga emas, balki bir nechta alternativ yo'lga taqsimlaydi. Bu quyidagicha ifodalanadi:

$$F = \sum_{i=1}^k f_i$$
$$f_i \leq C_i$$

Bu yerda F — umumiy trafik oqimi, f_i — i -yo'lga ajratilgan trafik, C_i — ushbu yo'lning sig'imi. Yuklamani balanslash algoritmlari kanal resurslaridan tengroq foydalanish, tiqilish ehtimolini kamaytirish va umumiy throughputni oshirishga xizmat qiladi [5; 8].



QoSga asoslangan algoritmlar trafik turlarini farqlashga tayanadi. Barcha trafik bir xil talabga ega emas. Masalan, VoIP past kechikish va past jitter talab qiladi, video streaming yuqori bandwidthga muhtoj, backup trafik esa kechikishga nisbatan kam sezgir bo'ladi. SDN-kontroller trafik turiga qarab marshrut tanlasa, tarmoq resurslaridan samarali foydalanish kuchayadi [5; 10].

Trafik ustuvorligini quyidagi formula orqali baholash mumkin:

$$Priority(f) = \lambda_1 SLA_f + \lambda_2 DelayReq_f + \lambda_3 LossReq_f$$

Bu yerda $Priority(f)$ — oqim ustuvorligi, SLA_f — xizmat darajasi talabi, $DelayReq_f$ — kechikishga sezgirlik, $LossReq_f$ — paket yo'qotishga sezgirlik. Ushbu model yuqori ustuvorlikdagi oqimlarni sifatliroq yo'llarga yo'naltirish imkonini beradi.

Trafikni dinamik boshqarish algoritmlarini tahlil qilishda samaradorlik mezonlari muhim o'rin tutadi. Eng asosiy ko'rsatkichlardan biri throughput hisoblanadi:

$$T = \frac{B}{t}$$

Bu yerda T — o'tkazuvchanlik, B — uzatilgan ma'lumot hajmi, t — vaqt. Throughput qancha yuqori bo'lsa, tarmoq resurslaridan foydalanish shuncha samarali hisoblanadi.

Kechikish quyidagi tarkibiy qismlar orqali ifodalanadi:

$$D_{total} = D_{proc} + D_{queue} + D_{trans} + D_{prop}$$

Bu yerda D_{proc} — qayta ishlash kechikishi, D_{queue} — navbatda kutish kechikishi, D_{trans} — uzatish kechikishi, D_{prop} — tarqalish kechikishi. SDN algoritmlari odatda D_{queue} qiymatini kamaytirishga yordam beradi, chunki trafik yuqori yuklangan kanallardan bo'shroq kanallarga yo'naltiriladi.



Paket yo'qotish koeffitsiyenti quyidagicha hisoblanadi:

$$PLR = \frac{N_{lost}}{N_{sent}} \times 100\%$$

Bu yerda N_{lost} — yo'qolgan paketlar soni, N_{sent} — yuborilgan paketlar soni. Paket yo'qotish darajasi real vaqt xizmatlari, sanoat IoT tizimlari va multimedia uzatish uchun muhim ko'rsatkich hisoblanadi.

Jitter ketma-ket paketlar kechikishidagi farqni bildiradi:

$$J_i = | D_i - D_{i-1} |$$

Bu yerda D_i — i -paket kechikishi. Jitter qiymati katta bo'lsa, audio va video xizmatlarda uzilishlar, sifat pasayishi yoki sinxronizatsiya muammolari yuzaga kelishi mumkin.

SDN asosida trafikni dinamik boshqarish algoritmlarini eksperimental tahlil qilish uchun Mininet, Ryu, OpenDaylight, ONOS, Floodlight yoki POX kabi platformalardan foydalanish mumkin. OpenDaylight SDN va NFV yechimlarini ishlab chiqish uchun ochiq platforma bo'lib, tarmoqni dasturiy boshqarish va monitoring qilish imkonini beradi [4]. Ayrim tadqiqotlarda POX, Floodlight va OpenDaylight kontrollerlari throughput va RTT bo'yicha taqqoslangan [8].

Eksperimental tahlilda statik marshrutlash, oddiy Dijkstra algoritmi, adaptiv SDN algoritmi, maksimal oqim modeli va QoS asosidagi model o'zaro solishtiriladi. Statik marshrutlash sodda va barqaror bo'lsa-da, tarmoq holati o'zgarganda moslashuvchan emas. Oddiy Dijkstra algoritmi eng qisqa yo'lni tanlaydi, lekin yuklamani har doim ham samarali muvozanatlashtirmaydi. Adaptiv SDN algoritmi link yuklamasi va kechikishni hisobga olgani sababli real vaqt sharoitida samaraliroq natija beradi. Maksimal oqim modeli katta hajmdagi



trafikni taqsimlashda foydali, QoS asosidagi model esa turli trafik sinflari uchun xizmat sifatini ta'minlashga qaratiladi [5; 7; 10].

SDN asosida algoritmlarni tahlil qilishda ularning afzallik va kamchiliklarini alohida ko'rsatish zarur. Dijkstra algoritmi sodda, tez va tushunarli. Maksimal oqim algoritmlari kanal sig'imidan samarali foydalanadi. Yuklamani balanslash tiqilishni kamaytiradi. QoSga asoslangan algoritmlar real vaqt xizmatlari sifatini yaxshilaydi. Adaptiv algoritmlar esa tarmoq holatiga moslashib boradi.

Biroq ushbu algoritmlarning barchasida muayyan cheklovlar mavjud. Real vaqt monitoringi qo'shimcha xizmat trafikini hosil qiladi. Kontrollerga ortiqcha yuk tushishi mumkin. Har bir oqim uchun tez-tez qayta marshrutlash flow setup time qiymatini oshiradi. Markazlashgan kontroller ishdan chiqsa, boshqaruv barqarorligi pasayishi mumkin. SDN xavfsizligi ham alohida muammo bo'lib, kontrollerga DDoS, soxta flow-rule yoki statistik ma'lumotlarni buzish kabi tahdidlar mavjud [5; 10].

SDNda dinamik trafik boshqaruvining yana bir muhim jihati prediktiv yondashuvdir. Trafik tarixi asosida keyingi vaqt oralig'ida qaysi kanal yuklanishini oldindan aniqlash mumkin. Buning uchun mashinali o'rganish, regressiya, neyron tarmoqlar yoki reinforcement learning usullaridan foydalanish mumkin. Bunday yondashuv tiqilish yuzaga kelgandan keyin emas, balki u yuzaga kelishidan oldin javob chorasi ko'rish imkonini beradi.

SDN asosidagi algoritmlarni tahlil qilish quyidagi umumiy xulosaga olib keladi: sodda tarmoqlar uchun Dijkstra algoritmi yetarli bo'lishi mumkin; trafik ko'p va kanallar xilma-xil bo'lgan tarmoqlarda yuklamani balanslash va maksimal oqim modellari samaraliroq; real vaqt xizmatlari ustuvor bo'lgan tarmoqlarda QoS asosidagi adaptiv algoritmlar afzal; katta va murakkab tarmoqlarda esa SDN-kontrollerlar klasteri, mashinali o'rganish va prediktiv monitoring bilan boyitilgan algoritmlar talab etiladi [5; 8; 10].



Xulosa. SDN asosida trafikni dinamik boshqarish algoritmlarini tahlil qilish zamonaviy tarmoqlarni samarali, moslashuvchan va dasturiy boshqarish imkoniyatlarini chuqurroq tushunishga yordam beradi. SDN arxitekturasi boshqaruv va ma'lumot uzatish tekisliklarini ajratish orqali kontrollerga tarmoq holatini global ko'rish va trafik oqimlarini real vaqt sharoitiga mos boshqarish imkonini yaratadi. Tahlil shuni ko'rsatadiki, Dijkstra algoritmi eng qisqa yo'lni topishda qulay, lekin yuklamani to'liq muvozanatlashtirishda cheklangan. Maksimal oqim algoritmlari kanal sig'implaridan samarali foydalanish uchun muhim. Yuklamani balanslash algoritmlari tarmoq tiqilishini kamaytiradi. QoSga asoslangan adaptiv algoritmlar esa real vaqt xizmatlari uchun kechikish, jitter va paket yo'qotish ko'rsatkichlarini yaxshilashga xizmat qiladi.

SDN asosidagi dinamik boshqaruv algoritmlarini baholashda throughput, kechikish, paket yo'qotish, jitter, link utilization va flow setup time kabi ko'rsatkichlar muhim hisoblanadi. Algoritm qanchalik ko'p parametrni hisobga olsa, uning moslashuvchanligi ortadi, lekin hisoblash murakkabligi va kontroller yuklamasi ham oshadi. Kelgusida SDN asosidagi trafik boshqaruvini mashinali o'rganish, prediktiv monitoring, ko'p kontrollerli arxitektura va xavfsiz OpenFlow aloqasi bilan rivojlantirish maqsadga muvofiqdir. Bu yondashuvlar tarmoq samaradorligini oshirish bilan birga, uning barqarorligi va xavfsizligini ham mustahkamlaydi.

Foydalanilgan adabiyotlar

1. Haleplidis E., Pentikousis K., Denazis S., Salim J.H., Meyer D., Koufopavlou O. Software-Defined Networking (SDN): Layers and Architecture Terminology. – RFC 7426. – IETF, 2015.
2. Open Networking Foundation. Software-Defined Networking (SDN) Definition. – Menlo Park: ONF.



3. Open Networking Foundation. OpenFlow Switch Specification. Version 1.5.1. – Menlo Park: ONF, 2015.
4. OpenDaylight Documentation. Getting Started Guide: Introduction. – OpenDaylight Project.
5. Kreutz D., Ramos F.M.V., Veríssimo P.E., Rothenberg C.E., Azodolmolky S., Uhlig S. Software-Defined Networking: A Comprehensive Survey // Proceedings of the IEEE. – 2015. – Vol. 103, No. 1. – P. 14–76.
6. Dijkstra E.W. A Note on Two Problems in Connexion with Graphs // Numerische Mathematik. – 1959. – Vol. 1. – P. 269–271.
7. Ford L.R., Fulkerson D.R. Maximal Flow through a Network // Canadian Journal of Mathematics. – 1956. – Vol. 8. – P. 399–404.
8. Altangerel G., Chuluuntsetseg T., Yamkhin D. Performance Analysis of SDN Controllers: POX, Floodlight and OpenDaylight // arXiv preprint. – 2021. – arXiv:2112.10387.
9. McKeown N., Anderson T., Balakrishnan H., Parulkar G., Peterson L., Rexford J., Shenker S., Turner J. OpenFlow: Enabling Innovation in Campus Networks // ACM SIGCOMM Computer Communication Review. – 2008. – Vol. 38, No. 2. – P. 69–74.
10. Nunes B.A.A., Mendonca M., Nguyen X.N., Obraczka K., Turletti T. A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks // IEEE Communications Surveys & Tutorials. – 2014. – Vol. 16, No. 3. – P. 1617–1634.