تكامُل وتحسين الشبكات المُعرّفة برمجيّاً المُوزّعة باستخدام بُنية شبكات البيانات المُسمّاة

د.م. يمان غازي[١] م. علي مصطفى[٢]

المُلخّص

لقد تناولنا في هذا البحث تحقيق التكامُل بين الشبكات SDN والشبكات NDN للحصول على البُنية الجديدة SD-NDN في المُحاكي ndnSIM، وقُمنا هنا بالتخصيص وتبني البُنية المُوزّعة وذلك بكتابة الترميز المصدريّ الذي يُحاكي عملَها بالإضافة إلى تضمينهِ في المحاكي، كما قُمنا بتحقيق البُنية IP-Based المُشابهة لها ضمن المحاكي Mininet من أجل المُقارنة بين البنيتين في البيئة NDN-Based والبيئة IP-Based حيث سوف يتم اسقاط ذلك من خلال إنشاء طوبولوجيا مُوحّدة وتحديد وسطاء تجربة مُناسبة، ومن ثم بدأنا بالتحقيق من خِلال تنفيذ سيناريو عمليّ على كلا المُحاكيين، وفي الجزء الأخير من هذا البحث تمَّ تقييم أداء البُنى المُحقّقة ومُقارنة الشبكة IP-Based مع شبيهتِها ضِمن البيئة NDN-Based. أثبتنا بالنتائج تفوّق البُنية ضمن بيئة NDN من ناحية الإنتاجية وتقليل الرُزم المفقودة والتأخير مُقارنةً مع البيئة التقليدية IP-Based، بينما ظهر لدينا أنَّ استخدام عرض الحزمة كان أكبر في البُنية NDN-Based.

الكلمات المفتاحية: شبكات البيانات المُسمّاة، الشبكات المُعرّفة برمجيّاً، الشبكات المُعرّفة برمجيّاً المُوزّعة، المُحاكي ndnSIM، المُحاكي Mininet، الشبكات المركزيّة، إنترنت المستقبل، شبكات SDN NDN (SD-NDN).

[١] استاذ مُساعد، قسم هندسة الشبكات والنظم الحاسوبية، كلية الهندسة المعلوماتية، جامعة حمص
[٢] طالب دراسات عليا (ماجستير)، قسم هندسة الشبكات والنظم الحاسوبية، كلية الهندسة المعلوماتية، جامعة حمص

Integrating and Improving Distributed Software–Defined Networks Using Named Data Networks Architecture

Dr. Eng. Yaman Ghazi[1] Eng. Ali Mustafa[2]

ABSTRACT

In this research, we have addressed the integration of SDN and NDN networks to obtain the new SD-NDN architecture in the ndnSIM simulator. Here, we have customized and adopted the distributed architecture by writing the source code that simulates its operation and including it in the simulator. We have also implemented the similar IP-Based architecture in the Mininet simulator to compare the two architectures in the NDN-Based environment and the IP-Based environment. This will be projected by creating a unified topology and determining appropriate experiment brokers. Then, we started the investigation by implementing a practical scenario on both simulators. In the final part of this research, we evaluated the performance of the implemented architectures and compared the IP-Based network with its counterpart in the NDN-Based environment. We demonstrated that the NDN architecture outperforms the traditional IP-based architecture in terms of throughput, packet loss reduction, and latency, while bandwidth utilization was higher in the NDN-based architecture.

[1] **Assistant Professor, Department of Network Engineering and Computer Systems, Informatics Engineering Faculty, Homs University.**

[2] **Postgraduate Student (M.A), Department of Network Engineering and Computer Systems, Informatics Engineering Faculty, Homs University.**

Keywords: Named Data Networking, Software Defined Networking, Distributed Software Define Networking, Information Centric Networking, Future Internet, SDN NDN Networks (SD-NDN).

١- مُقدمة:

إنَّ شبكات البيانات المُسمّاة (NDN) هي بُنية شبكات مركزيّة للمعلومات (ICN) تُحوّل نموذج الإنترنت من مركزيّة المضيف إلى مركزيّة المعلومات، مع استخدام أسماء المحتوى على مستوى التطبيق مباشرةً في طبقة الشبكة للبحث عن البيانات واسترجاعها. تعتمد شبكات NDN على تبادل نوعين من الرُزم: رُزم الاهتمام (Interest)، الذي يستخدمها المُستهلك (Consumer) لطلب المحتوى، ورُزم البيانات (Data)، التي يستخدمها المُزوّد (Producer) للرد على الطلب. بالمقارنة مع تصميم IP، فإن مستوى بيانات NDN هو حالة، ومتكيف، وينفذ التخزين المؤقت داخل الشبكة. عندما تستقبل عقدة NDN حزمة اهتمام، فإنها تبحث أولاً عن نتيجة بيانات مخزنة مؤقتًا في مخزن المحتوى المحليّ (Content Store - CS). في حالة الفشل، تبحث عن اسم مطابق في جدول الاهتمامات المُعلّقة (PIT)، الذي يسجل الاهتمامات المرسلة التي لم يتم تلبيتها بواسطة البيانات بعد. إذا تم العثور على تطابق، يتم تجاهل الطلب لأن هناك اهتمامًا مُساويًا معلقًا بالفِعل. إذا فشل كل من فحص المخزن CS والجدول PIT، يمكن للعقدة إعادة توجيه الرُزمة، وِفقًا للقواعد في قاعدة معلومات التوجيه (FIB)، المملوءة إما ببروتوكولات التوجيه أو من قبل مسؤولي الشبكة. **[1]**

يُحاول المُصممون ضمن شبكات SDN إعادة ترتيب أجزاء وأدوار كل مُكونات البُنية التحتيّة للشبكات (infrastructure)، والتي لم تُعدّل منذ عام 1980 حيث أن أخر تعديل كان هو الانتقال من شبكات الجيل القديم NCP إلى شبكات TCP/IP الغنيّة عن التعريف، ومنذ ذلك الحين لم يطرأ أي تغيّير على مُستوى البُنية التحتيّة للشبكات لتتلاءم والتطوُر

الكبير الحاصِل في مجال تقنية المعلومات، وخصوصاً التقنية التخيليّة virtualization التي قامت بمحاكاة افتراضية لجميع طبقات تصميم الشبكات، ولكن ظلّت طبقة البُنية التحتيّة للشبكات مُستعصية على هذه التقنية. إذاً شبكات SDN هي محاولة ناجحة لفصل مُستوى البيانات Data plane أو ما يُعرف باسم Forwarding plane عن مُستوى التحكُّم control plane، ليصبح دور أجهزة الشبكات Switches/Routers مُنحصِر فقط في تمرير البيانات forwarding، أما عمليات الإدارة والتحكُّم والخدمات ستُصبِح في طبقات جديدة وهي طبقتين مُنفصلتين.

عرَّفت مُنظمة ONF البُنية المِعماريّة لتقنية SDN على شكل نموذج (Module) مُكوّن من ثلاث طبقات، أو بمعنى أخر فإنّ SDN تقسم البنية التحتية للشبكات إلى ثلاث طبقات (Data, Control and Application). [**14**]

٢- الهدف من البحث:

ساهمنا عبر هذا البحث في تحقيق تكامُل فعليّ لبُنية شبكات NDN مع بُنية شبكات SDN باستخدام عدة مُتحكُّمات ضمن المُحاكي ndnSIM وذلك بكتابة الترميز المصدري الذي يبني ويُحاكي عمل هذه البُنية وتضمينهُ في المُحاكي، ثُمَّ استخدام المُحاكي Mininet لمُحاكاة ذات البُنية المُتكاملة. كما تتمثّل الأهداف الخاصة في تحقيق بُنية SDN مُوزّعة مع شبكات NDN وقياس مُعامِلات الأداء، وتحديد التحسينات التي سنحصل عليها عند تحقيق تكامُل هذه البُنى.

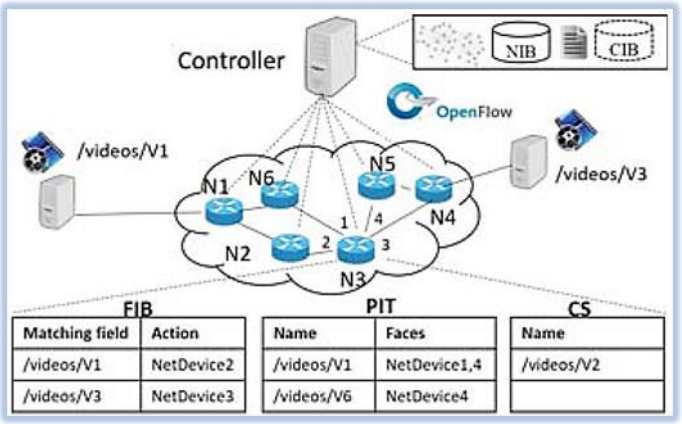
٣- البُنية SDN NDN المُتكاملة الهجينة:

يُوفّر نهج شبكات NDN مزايا عديدة مقارنةً بالنهج القائم على IP. على سبيل المثال، فهو يقلل من أعباء المعالجة ضِمن الشبكة ويُحسّن توافر المحتوى بفضل التخزين المؤقت داخل الشبكة، والذي يسمح لأي عقدة بتخزين البيانات وتقديمها. كما أنه يتيح نقل البيانات بكفاءة، بفضل التوصيل المُدار من قِبل المُستقبِل الذي يرسل رُزم البيانات فقط

استجابةً لرُزم الاهتمام (Interest) من المُستهلكين، مما يتجنّب البيانات غير المرغوب فيها أو المكررة. علاوةً على ذلك، فإنَّ أي عُقدة ضِمن الشبكة قادرة على تأمين الأمان والتشفير اللازم من تلقاء ذاتها بدون اللجوء لأطراف خارجيّة لتأمين الحماية. **[3]**

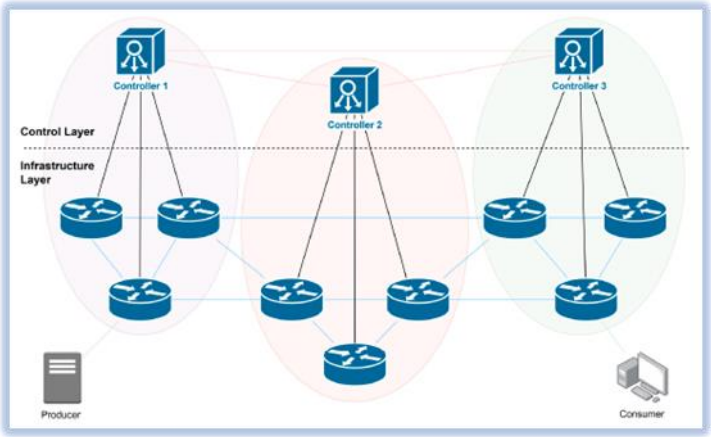
في الآوِنة الأخيرة، تم اعتبار الشبكات المُعرّفة برمجيّاً (SDN) حلاً مُحتملاً ليتم دمجه مع شبكات NDN، من أجل تحسين ميزات مستوى التحكُّم وتطبيق سياسات هندسة حركة المرور. تقوم شبكات SDN بفصل مُستوى التحكُّم عن مُستوى البيانات، ونتيجة لذلك يتم تنفيذ قرار التوجيه بواسطة كيان مركزي منطقيًا، وهو وحدة التحكُّم. في تصميم SDN NDN المُتكامُل، يتم تزويد عُقد الشبكة بجداول مستوى بيانات NDN، مع ملء FIBs بواسطة وحدة تحكُّم SDN. عندما لا يمكن التعامل مع حزمة مهمة محليًا في مستوى بيانات NDN، يتم الاتصال بوحدة تحكُّم SDN التي تنفذ قرار التوجيه وتحفُن قواعد إعادة التوجيه القائمة على الاسم في FIB وفقًا لذلك. في شبكة ذات عدد كبير من طلبات المحتوى، قد تتعرّض قواعد بيانات FIBs للتدفق الزائد بسهولة نظرًا لحجم الجدول المحدود، مع ما يصاحب ذلك من مشاكل في قابلية التوسع. في الواقع، يجب الحفاظ على حجم قاعدة بيانات FIB صغيرًا لسرعة البحث وأداء توجيه جيد. قد يؤدي التدفق الزائد في قاعدة بيانات FIB إلى حظر الطلبات الجديدة أو حذف الإدخالات الحالية لاستيعاب القواعد الجديدة التي أدخلها المتحكُّم. وبالتالي، فإن القرارات المتعلقة بالإدخالات التي يجب تخزينها في قاعدة بيانات FIB ومدة الاحتفاظ بها تؤثر بشكل كبير على أداء توجيه الرُزم. **[1]**

يُوضّح الشكل (١) البُنية المُقترحة لشبكات SD-NDN وِفق المرجِع **[1]**، وهي بُنية وحيدة المُتحكُّم وقُمنا فقط بدِراسة وفهم للبُنية انطلاقاً منها.



الشكل (١) بُنية SDN- NDN المُقترحة ضِمن المرجع [1]

يُمثل الشكل (٢) مثالاً على بُنية أُخرى لكنها مُفصّلة وواضِحة أكثر وتُقسّم هذه البُنية الشبكة إلى طبقة التحكُّم وطبقة البُنية التحتيّة (العُقد).



الشكل (٢) مِثال أوضح عن بُنية SD-NDN المُوزّعة [4]

ستكون مهمة عُقد التوجيه هنا هي إعادة توجيه الرُزم، وبالتالي تنتقل أعباء التوجيه إلى أجهزة المُتحكّمات. **[4]**

سنقوم باعتماد شكل هذه البيئة (الشكل ٢) ضمن البُنية SD-NDN المُوزَعة الخاصة بنا، وسنستخدم وسطاء خاصة بالمُحاكاة وسنقوم بذِكرها لاحقاً.

٤- أدوات تحقيق البحث

لقد استخدامنا في الاختبارات والتحقيق برنامج المُحاكاة **Mininet** (المُنصّب على التوزيعة Ubuntu 20.04 من نِظام Linux) لنمذجة وتحقيق بُنية الشبكات المُعرّفة برمجيّاً حيثُ يسمح هذا المُحاكي باختبار الطوبولوجيا بأنواعها دون الحاجة إلى شبكة مادية للتحكُّم في الشبكة الافتراضيّة، كما يتضمّن Mininet واجهة سطر أوامر (CLI) كما يُوفّر واجهة برمجيّة مُحقّقة بلُغة Python تُساعِد في بِناء الشبكة واختبار الأداء. **[6]**

كما قُمنا باستخدام المُحاكي **ndnSIM** لإجراء تجارب المُحاكاة ودِراسة النتائج ضمن البيئة NDN-Based، حيثُ يُمكّننا من إنشاء طوبولوجيا الشبكة وتحديد مُعلّماتِها ومُحاكاة الاتصال بين عُقد NDN المُختلفة وتسجيل أحداث المُحاكاة بالإضافة إلى عرض النتائج بشكل واضح.

إنّ المكاتب ndn-cxx و NFD ضِمن المُحاكي ndnSIM تضمن أن تكون المُحاكاة واقعية إلى أقصى حد. **[14]**

لقد تم إجراء تجارُب وسيناريوهات المُحاكاة باستخدام بيئة ndnSIM 2.9 (وهو الإصدار الحالي).

٥- الدِراسات السابِقة:

لقد بحثنا ضِمن الكثير من الدِراسات التي تتناول موضوع التكامُل في البُنية بين شبكات NDN وشبكات SDN فمثلاً يقوم الباحثين في الورقة البحثية **[1]** بتطبيق بُنية SDN NDNباستخدام بروتوكولOpenFlow كما هو مُستخدم في المراجع **[7]** و **[8]**، حيث تمَّ هُنا نقل عمليات الإدارة واتخاذ قرارات التوجيه إلى عُقدة المُتحكّم، وتم الحصول على

تحسينات في توجيه الطلبات. سنعتمد في هذه الورقة على دِراسة هذه البُنية واحتمال استخدامها لاحقاً ضمن أي توسعات مستقبلية مع بُنية الشبكات SDN.

بينما اقترح الباحثون في الورقة البحثيّة **[2]** بُنية هجينة تُسمّى NDN–Fab، تسمح بالاتصال بين بُنى الشبكة المُعتمدة على العناوين (Ip–based) والمُحتوى (Content–based)، مثل NDN، وتم استخدام مُتحكّم SDN لتوجيه الرُزم ضمن البُنية الهجينة الجديدة. كما تمت التوصية بتطبيق البُنية NDN–Fabric على مجال أوسع وهو ما اعتبرناه دافع باتجاه تطبيق البنية الجديدة على كل طبقات الشبكة.

ركّزت الورقة البحثيّة **[5]** على دِراسة فعالية شبكات البيانات المُسمَّاة في سياق الشبكات المُتغيرة التكتيكية (Tactical networks)، وضّح الباحثون هنا أنّه على الرغم من أن بُنية ICN تُقدّم بعض الفوائد للبيئات التكتيكية (المتحرّكة) أكثر من البُنية التقليديّة TCP / IP، إلا أن تحديات التطبيق والتوسِعة كبيرة. كما قاموا بعرض السمات التي جعلت شبكات NDN مُؤهلة للتطبيق ضِمن الشبكات التكتيكيّة العسكريّة. يمكن أن نعتمد هذه السمات كتوصيف للتصميم الخاص بالبُنية الجديدة.

وقد قدّم الباحثون ضمن الورقة **[4]** عملية دمج وتكامُل شبكات SDN و NDN، وعرضوا أبرز الصعوبات التي تتحدى البُنية الجديدة، حيث يُؤدي الاندماج بين مُتحكّمات شبكات SDN وبِّنية الشبكات NDN إلى تحدٍ كبير، مما يتطلّب تطوير برامج تحكُّم متخصصة مُتوافِقة مع كِلا شبكات SDN وشبكات NDN. إنّ غالبية بُنى SD–NDN قد تُؤدي إلى إنقاص الميّزات التي تُوفّرها شبكات NDN. قمنا باعتماد جزء من البُنية المُقترحة في هذه الورقة ضمن البُنية SD–NDN المُوزّعة (مُتعددة المُتحكُّمات) والتي سنتناولها ضِمن بحثنا.

ولقد تناول الباحثون في البحث **[15]** مزايا الانتقال من الشبكات التقليديّة إلى الشبكات المُعرّفة برمجيّاً بما تحويه من وحدات تحكُّم خارجيّة مما يُوفّر بيئة أبسط لبرمجة ولإدارة

الشبكة وتحديد سُلوكها. كما قاموا بمُقارنة أداء إثنين من المُتحكِّمات مفتوحة المصدر وهي pox و Opendaylight من حيث العوامل التالية المُؤثِّرة أداء الشبكة وهي متوسط تأخير التدفق ومتوسط مُعدّل البِتات ومتوسط مُعدّل الرُزم و jitter باستخدام الأداة–D ITG لتحديد المُتحكِّم ذو الأداء الأفضل في عدد من بُنى الشبكات الأساسيّة المُضمّنة في المُحاكي Mininet وهيsingle و linear وtree. وتمت الإشارة إلى أن وِحدة التحكُّم pox توفر سهولة أكبر من ناحية تشغيلها وفهم مكوناتها وتطويرها، ولقد قمنا ضمن بحثنا هذا باستخدام هذا المُتحكِّم (POX) ضمن البُنية المُوزّعة في البيئة –IP Based.

يهدُف الباحثون ضمن الورقة البحثيّة [9] إلى دِراسة سياسات التخزين المُؤقّت للمحتوى وسياسات استبدال المحتوى ضِمن جهاز التوجيه ضِمن شبكات NDN، كما يهدف البحث من ناحية أُخرى إلى تَحسين أَداء سياسة استبدال ال Cash بالاعتماد على شعبيّة المُحتوى (CCP)، وعملوا عبر التجارِب على تحسين عِدّة وسطاء أداء مُرتبة بمفهوم ال Cashing ضِمن مُوجّهات الشبكات NDN. استفدنا من هذ البحث في دِراسة بُنية شبكات NDN الصافية وفهم كل ما يتعلّق بها من مفاهيم أساسيّة والتخزين المؤقت وسياساتهُ وأنواع الرُزم ووسطاء المُحاكاة المُستخدمة.

رأينا من ضمن ما سبق من دِراسات عِدة نماذج وبُنى مُقترحة للبُنية الجديدة SD-NDN. سنعمل بدورنا على تحديد وسطاء المُحاكاة المُناسبة وتحقيق البُنية المُوزّعة ضمن البيئتين IP-based و NDN-Based، وقُمنا بالمُقارنة بين أهم مُعاملات الأداء من إنتاجية وتأخير وفُقدان الرُزم واستهلاك عرض الحُزمة وإظهار التحسينات التي أضافتها شبكات NDN ضمن بُنيتنا الجديدة SD-NDN المُوزّعة.

٦- تحقيق البُنية Distributed SD-NDN في المُحاكيين Mininet و ndnSIM:

سيتم في هذا الفقرة تحقيق البُنية مُتعددة المُتحكِّمات ضمن البيئتين IP-Based (على المُحاكي Mininet) والبيئة NDN-Based (على المُحاكي ndnSIM)، لكن علينا أولاً وضع مُعلّمات التجارب.

١,٦ إعداد مُعلّمات المُحاكاة

قُمنا بأخذ وُسطاء وِفقاً لقيم نموذجيّة مأخوذة من شبكات البُنية الجديدة SDN-Based Internet2 (2014+) مع شبكات NDN.

ملاحظة: إنّ المجموعة Internet 2 عِبارة عن شبكة تعاونية للأبحاث والتعليم تم تطويرها بواسطة اتحاد Internet2 consortium، وهي منظمة غير ربحية مقرُّها الولايات المُتحدّة الأمريكيّة ويركّز عملها على دعم وتشغيل الشبكات المُتقدّمة للمؤسسات الأكاديمية والبحثية.

يُوضّح الجدول (١) القيم النموذجية لوسطاء المُحاكاة فيما يتعلّق بشبكات NDN:

جدول (١) القيم النموذجية لوسطاء المحاكاة وِفق توصيات مُنظمة Internet 2

Metric	Description	Typical Value
Simulation Time	The total duration of the simulation.	**100s - 1000s** (depends on the network size and objectives)
Average Latency (Delay)	The round-trip time (RTT) for packets to travel from consumer to producer and back.	**30ms - 180ms** (depends on network scale and path length)
Queue Size	The maximum number of packets that can be stored in a router's queue before dropping.	**50 - 200 packets** (varies with network load and traffic)
Data Rate	The rate at which data is transmitted over the network.	**100 Gbps - 10 Gbps** (depending on link speed and capacity)
Content Store Size	The maximum number of entries (packets/data chunks) that can be stored in the Content Store (CS).	**1000 - 5000 entries** (based on network caching strategy)
Interest Requests Frequency	The frequency of Interest packets sent by consumers to request data.	**1 per second to 10 per second** (depends on consumer behavior and simulation setup)

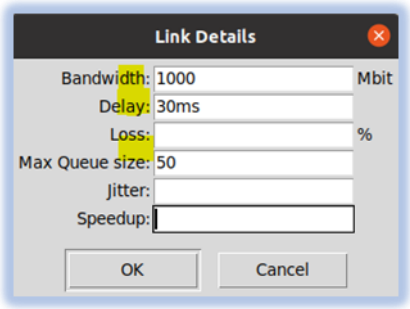
نُوضّح بالجدول (٢) الوسطاء الخاصة بالمُحاكاة ضمن سيناريو البُنية SD–NDN المُوزّعة (Distributed Controllers):

جدول(2) وسطاء مُحاكاة البُنية SD-NDN المُوزَّعة

الوسيط	القيمة
زمن المُحاكاة	100 (Sec)
عدد المُتحكِّمات	٣
عدد العقد NDN	9 (Mesh)
عدد أزواج ال Producers/Consumers	1
حجم البيانات المُرسلة (Payload Size)	10 (MB)
التأخير على الوصلات (Delay)	30 (ms)
معدّل نقل البيانات (Data Rate)	1 (Gbps)
حجم الرتل على منافذ العُقد (Queue Size)	50 (packets)

حجم ذاكرة تخزين المحتوى (Content Store)	1000 (entries)
تردد الطلبات (Interest (Request Frequency	10 (pkt per second)
عدد تكرارات المُحاكاة	5

نقوم بوضع هذه الوسطاء بنفس القيم ضمن المُحاكي Mininet، ويتم ذلك عبر واجهة المُحاكي الرسوميّة: (الشكل (٣))



الشكل (٣) تحديد وسطاء المُحاكاة ضمن المُحاكي Mininet

نضع هذه القيم في المُحاكي ndnSIM ضمن ملف الشيفرة المصدريّة الخاصة بإعدادات وطوبولوجيا الشبكة المُوزّعة وِفق التالي (الشكل (٤))

الشكل (٤) تحديد وسطاء المُحاكاة ضمن المُحاكي NDN SIM

عِلماً أنّ الشيفرة المصدريّة والملف الخاص بالطوبولوجيا الشبكيّة مُتاحَين على الرابط أَدناه:

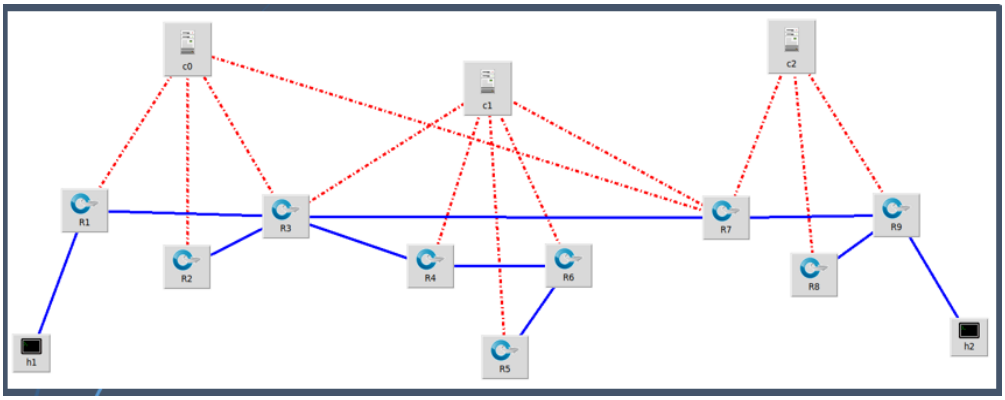
https://github.com/Ali-Mustafa96/Distributed_SD-NDN.git

٢,٦ بناء وتحقيق البُنية SD–NDN المُوزّعة

ستتضمّن البُنية المُوزّعة التي سنعتمدها ثلاث مُتحكِّمات (C0, C1, C2) وتسعة عُقد NDN Nodes (R1, R2, R3, … , R9) وعقدة المُستهلِك المُرسل للطلبات (Consumer 1) وعقدة المُنتج المُستقبِل للطلبات (Producer 1).

تم تحقيق هذه البُنية ضمن المُحاكي Mininet (بيئة IP–Based) وتشغيل الطوبولوجيا وإرسال بيانات بين عقدتي الإرسال والاستقبال كما هو مُحدّد ضِمن فقرة إعداد مُعلّمات المُحاكاة سابِقاً.

نُوضّح بالشكل (٥) هيئة الطوبولوجيا مُتعددة المُتحكِّمات (المُوزّعة) الناتجة داخل المُحاكي Mininet.



الشكل (٥) تحقيق البُنية SDN IP-Based التقليديّة المُوزّعة ضمن المُحاكي Mininet

ثمّ قُمنا بتحقيق هذه البُنية ضمن المُحاكي ndnSIM (أي ضمن البيئة NDN–Based) وتشغيل الشيفرة المصدريّة وعرض الطوبولوجيا الشبكيّة.

نُوضّح بالشكل (٦) الملف النصيّ الخاص بالطوبولوجيا والذي ستتم قراءتهُ في بداية الشيفرة المصدريّة الحاوية لإعدادات المُحاكاة. يحتوي هذا الملف ضِمن قِسمه الأول على جدول تعليمات لبِناء العُقد وتحديد إحداثياتها (x, y)، بينما يحتوي القسم الثاني على جدول تعليمات تعمل على إنشاء الوصلات بين العُقد وتحديد المواصفات الخاصة بها من سرعة وتأخيرات وحجم الأرتال.

```
# topo-tree-Distribute.txt

 * This scenario simulates a tree topology (using topology reader module)


router

#node   city    y   x   mpi-partition
leaf-1  NA  -90 -40 1
leaf-2  NA  -90 100 2
rtr-5   NA  -85 30  0
rtr-3   NA  -70 -10 1
rtr-4   NA  -70 20  3
rtr-6   NA  -70 40  2
rtr-8   NA  -70 70  4
rtr-1   NA  -60 -20 1
rtr-2   NA  -60 0   3
rtr-7   NA  -60 60  2
rtr-9   NA  -60 80  4
Cont1   NA  -45 30  0
Cont0   NA  -40 -10 1
Cont2   NA  -40 70  2

link

# from      to      capacity    metric  delay   queue
leaf-1      rtr-1   1Gbps       1   30ms    1000
rtr-3       rtr-2   1Gbps       1   30ms    1000
rtr-3       Cont0   1Gbps       1   30ms    1000
rtr-1       rtr-2   1Gbps       1   30ms    1000
rtr-1       Cont0   1Gbps       1   30ms    1000
rtr-2       Cont0   1Gbps       1   30ms    1000
Cont0       rtr-7   1Gbps       1   30ms    1000
rtr-2       Cont1   1Gbps       1   30ms    1000
rtr-2       rtr-7   1Gbps       1   30ms    1000
rtr-2       rtr-4   1Gbps       1   30ms    1000
Cont1       rtr-4   1Gbps       1   30ms    1000
Cont1       rtr-5   1Gbps       1   30ms    1000
Cont1       rtr-6   1Gbps       1   30ms    1000
Cont1       rtr-7   1Gbps       1   30ms    1000
rtr-4       rtr-6   1Gbps       1   30ms    1000
rtr-5       rtr-6   1Gbps       1   30ms    1000
rtr-4       rtr-6   1Gbps       1   30ms    1000
Cont2       rtr-7   1Gbps       1   30ms    1000
Cont2       rtr-8   1Gbps       1   30ms    1000
Cont2       rtr-9   1Gbps       1   30ms    1000
rtr-7       rtr-9   1Gbps       1   30ms    1000
rtr-8       rtr-9   1Gbps       1   30ms    1000
rtr-9       leaf-2  1Gbps       1   30ms    1000
```

الشكل (٦) بُنية الملف النصيّ الخاص بالطوبولوجيا المُوزّعة ضمن المُحاكي ndnSIM

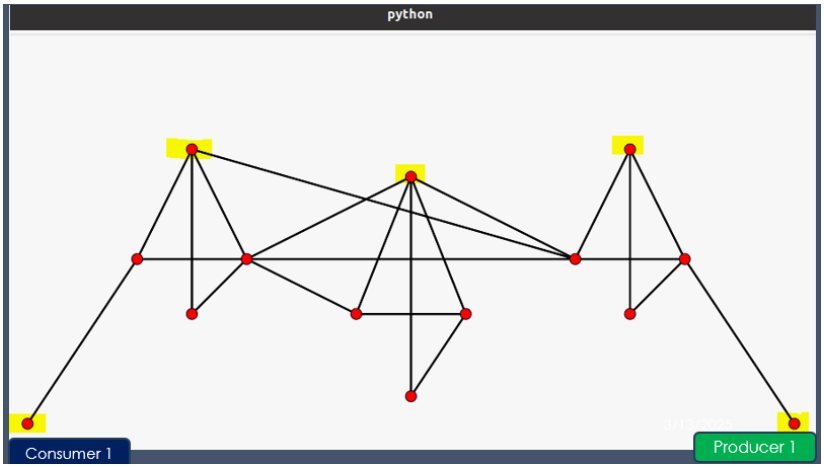
نعرُض بالشكل (٧) كيفيّة إضافة عُقد التحكُّم (لدينا ثلاث مُتحكّمات). حيث أنشأنا ثلاث أغراض من الموديول الفرعيّ "LearningController" (وهو نموذج برمجيّ يستخدم لبِناء المُتحكّمات) المُستورد من الموديول الأساسيّ "OpenFlow" ونربط هذه الأغراض مع العُقد التي قُمنا بقراءتها من ملف الطوبولوجيا مع تحديد ال Container المحتوية على العُقد الأُخرى (Consumer, Producer and NDN nodes).

```
135   Ptr<Node> switchNode1 = Names::Find<Node>("Cont0"); // The controller Cont0 node
136   Ptr<Node> switchNode2= Names::Find<Node>("Cont1");  // The controller Cont1 node
137   Ptr<Node> switchNode3= Names::Find<Node>("Cont2");  // The controller Cont2 node
138
139   // Create the switch netdevice, which will do the packet switching (Install OpenFlow )
140
141   OpenFlowSwitchHelper swtch1;
142   OpenFlowSwitchHelper swtch2;
143   OpenFlowSwitchHelper swtch3;
```

```
157     {
158       Ptr<ns3::ofi::LearningController> controller1 = CreateObject<ns3::ofi::LearningController> ();
159       if (!timeout.IsZero ()) controller1->SetAttribute ("ExpirationTime", TimeValue (timeout));
160       swtch1.Install (switchNode1, switchDevices, controller1);
161
162       Ptr<ns3::ofi::LearningController> controller2 = CreateObject<ns3::ofi::LearningController> ();
163       if (!timeout.IsZero ()) controller2->SetAttribute ("ExpirationTime", TimeValue (timeout));
164       swtch2.Install (switchNode2, switchDevices, controller2);
165
166       Ptr<ns3::ofi::LearningController> controller3 = CreateObject<ns3::ofi::LearningController> ();
167       if (!timeout.IsZero ()) controller3->SetAttribute ("ExpirationTime", TimeValue (timeout));
168       swtch2.Install (switchNode3, switchDevices, controller3);
169     }
170
```

الشكل (٧) إضافة عُقد المُتحكّمات ضِمن الشيفرة المصدريّة في المُحاكي ndnSIM

نُوضّح بالشكل (٨) شكل الطوبولوجيا المُوزّعة الناتجة ضمن المُحاكي ndnSIM بعد تشغيل الشيفرة المصدريّة الخاصة بها.



الشكل (٨) تحقيق البُنية SDN NDN-Based المُوزّعة ضمن المُحاكي ndnSIM

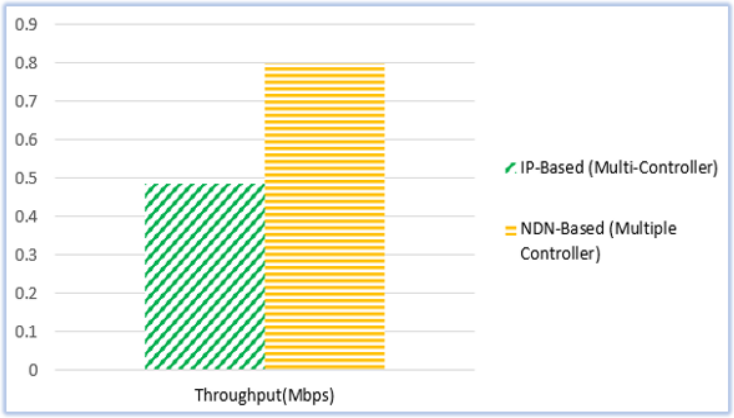
٧- اختبار أداء البُنية المُوزّعة ضمن البيئتين IP-Based و NDN-Based وتحديد التحسينات بعد تحقيق التكامُل مع بنية شبكات NDN في البُنية الجديدة:

سنقوم باختبار الأداء وِفق معايير الإنتاجية والتأخير وفقدان الرُزم واستهلاك عرض الحزمة. (قُمنا بتكرار عملية المُحاكاة عدة مرات وتمّ الحصول نتائج مُتقاربة)

تُعبّر الانتاجيّة عن كمية البيانات المُستلمة فعلياً بشكل سليم ضمن زمن مُحدّد، بينما يصِف التأخير الفترة الزمنيّة الكليّة حتى تتم معالجة وإيصال البيانات إلى وِجهتها، بينما فُقدان الرُزم (Packets Drop) تُعبر عن عدد الرُزم المُهملة (المفقودة) خلال الإرسال.

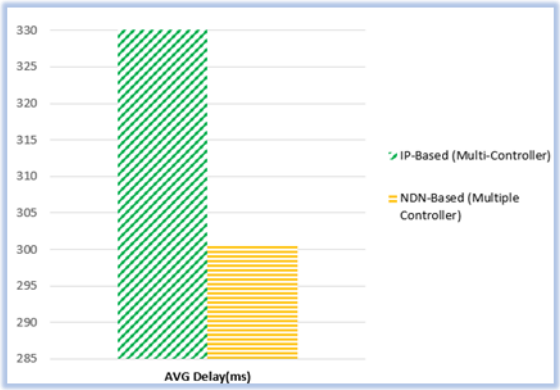
كما استخدمنا استهلاك عرض الحُزمة كمعيار أداء لإنّهُ يصِفُ مِقدار النطاق الترددي المطلوب أن تستهلكهُ البُنية الشبكية حتى تُنجز عملها. وبناءً على هذه المعايير سنُفاضِل بين البُنية SDN التقليديّة والبُنية SD-NDN.

قُمنا بإنجاز جميع التجارُب على المُحاكاة التي قُمنا بإعدادها وحصلنا على تحسُّن في الإنتاجية كما في الشكل (٩) بسبب قِلة الرُزم المفقودة التي لاحظناها في البيئة NDN– Based وسنوضّح ذلك عند عرضنا لمعيار فُقدان الرُزم.



الشكل (9) مُخطط يُوضّح الفرق بالإنتاجية ضمن البُنيتين IP-Based Multi-Controller

كما نعرِض ضمِن الشكل (١٠) التحسين الحاصِل بالتأخير (Delay) ضمن البيئة NDN-Based:



الشكل (١٠) مُخطط يُوضّح الفرق بالتأخير ضمن البُنيتين Multi-Controller

يعود التحسين من ناحية التأخير للأسباب التالية:

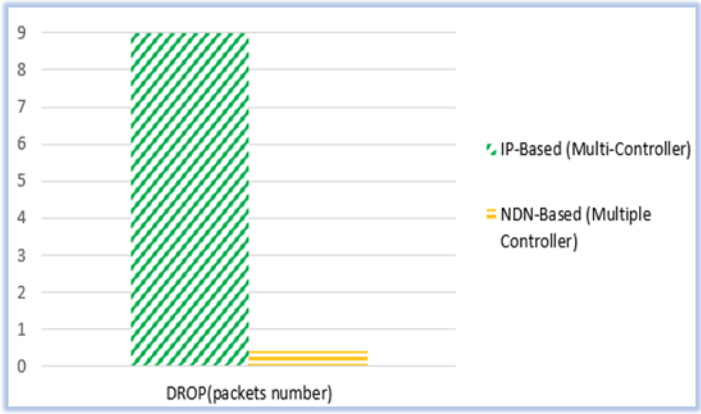
a. **السبب الأبرز هو التوجيه المُتكيّف (Adaptive Forwarding)**، حيث تدعم شبكة NDN إعادة التوجيه متعدد المسارات وفي حال حدوث أي خطب على وصلات الاتصال بين العُقد، يتم إعادة توجيه البيانات بشكل تلقائيّ عبر مسارات بديلة واختيار أفضل مسار ديناميكيًا دون انتظار وصول معلومات تحديث من مُتحكّم SDN بينما تعتمد شبكة SDN عادةً على مسارات مُعدّة مُسبقًا، مما يُسبب الزيادة في التأخير. **[10]**

b. **لا حاجة للاتصال نهاية إلى نهاية (End-to-End Connectivity)**، حيثُ تقوم شبكات NDN واعتماداً على أسماء المحتوى بعملية التوجيه بدلاً من العناوين المنطقيّة IP، مما يُلغي الحاجة إلى اتصالات نهاية إلى نهاية (End-to-End). يتم بذلك استرجاع البيانات من أي عُقدة تُخزّن المحتوى مؤقتًا، مما يُقلل من التأخيرات المرتبطة بإعدادات الاتصال. **[11]**

c. **لا يوجد ضِمن شبكات NDN أية أعباء إضافيّة مثل النموذج TCP/IP**، حيثُ يؤدي التخلّص من البروتوكول TCP/IP ضمن بيئة NDN إلى تقليل أزمنة

المُعالجة، بينما لا تزال شبكات SDN التقليديّة تستخدم بروتوكول TCP/IP، مما يتطلّب عمليات إضافية من أجل إعادة الإرسال ومعالجة الأخطاء والاختناقات والتدفّق. [12]

لاحظنا بالتجارب أيضاً قِلة فُقدان الرُزم ضمن البيئة NDN–Based كما في الشكل (١١) وقلنا أنّ هذا الأمر يُحسّن الإنتاجية.



الشكل (11) مُخطط يُوضّح الفرق بفقدان الرُزم ضمن البُنيتين Multi-Controller

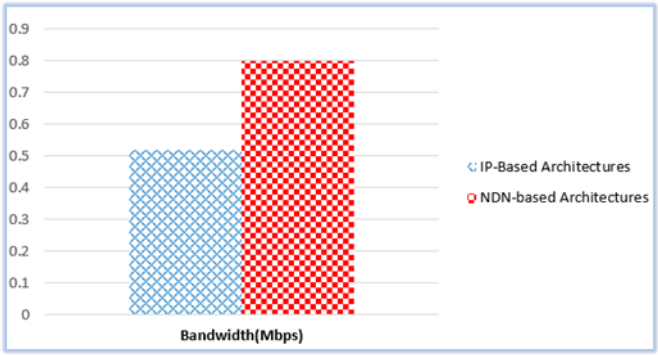
إنّ التحسُّن في إيصال الرُزم يعود لما يلي:

a) **التخزين المؤقت داخل الشبكة (In–Network Caching)**، حيثُ يُتيح التخزين المُؤقت داخل شبكات CCN استرجاع البيانات من العُقد الوسيطة، ويُقلّل من الحاجة إلى إعادة الإرسال من المصدر، الأمر الذي يُؤدي إلى تقليل فقدان الرُزم. [12]

b) **آلية الرُزم Interest/Data نفسها تُقلل من الازدحام وفقدان الرُزم**، حيثُ يتم ضمن شبكات NDN تبادل للبيانات فقط عند الطلب عن طريق رُزم ال Interest، مما يُقلل من نقل البيانات غير الضروريّ وتقليل فقدان الرُزم. بينما في شبكة SDN التقليديّة، ترسل البيانات عبر الوصلات إلى وجهتها، حتى لو لم يكُن بالإمكان

التعامل معها وتوجيهها فورًا، مما قد يؤدي إلى نفاذ مساحة التخزين المُؤقّت (Buffers) وارتفاع نسبة الفقد في الرُزم. **[11]**

على أية حال، إنّ البُنية SD–NDN تُعتبر أسوء من SDN المستند إلى IP (التقليديّة) من ناحية استهلاك النطاق الترددي (Bandwidth) ولاحظنا ذلك ضمن التجارُب ونتج لدينا الشكل (١٢) الذي يُوضّح الفرق بهذا الاستهلاك.



الشكل (١٢) مُخطط يُوضّح الفرق باستهلاك عرض الحزمة بين البيئتين IP-Based و NDN-Based

ترتفع نسبة استهلاك عرض الحُزمة ضمن بُيئة SDN المُعتمدة على NDN بسبب:

A. تستهلك حركة مرور الخاصة بعمليات مزامنة ذاكرة التخزين المُؤقت (Cache Synchronization Traffic) عرض حُومة أكبر، حيثُ تقوم شبكات NDN بآليات مُزامنة ذاكرة التخزين المؤقت (للحِفاظ على الاتّساق بالمُحتوى بين العُقد NDN) الأمر الذي يدعو لوجود بيانات تحكُّم إضافية. تستهلك تحديثات ذاكرة التخزين المؤقت في شبكةNDN عرض حُزمو إضافيّ، خاصةً في الشبكات الديناميكية والكبيرة. **[13]**

B. تزيد بادئات الأسماء الكبيرة من حجم الرُزمة (Large Name Prefixes)، حيثُ تستخدم NDN أسماء المحتوى بدلاً من عناوين IP، وبالتالي مُمكن أن تحتوي ترويسات الرُزم على مسارات تسمية هرميّة طويلة. بينما نعلم أنّ آلية العنونة تكون ذات حجم ثابت في الأنظمة IP–Based. **[13]**

تمت تقديم التوصية ضمن شبكات البُنية الجديدة SDN–Based Internet2 (٢٠١٤+) مع شبكات NDN بأن يكون عرض الحزمة كافي ومن مرتبة كبيرة تصِل إلى ١ Gbps وما فوق، وبالتالي فإنّ استهلاك عرض الحزمة لن يكون تحدٍ كبير عند تحقيق التكامُل بين البُنيتين.

٨- الخاتِمة والأعمال المُستقبليّة

قدّمنا في هذا البحث نموذج وتحقيق فعليّ للبُنية الجديدة SD-NDN مُتعدّدة المُتحكّمات وقُمنا بتحديد ما يُناسِبُها من مُعلّمات مُحاكاة، وقد ضمَّ هذا التحقيق كِلا البيئتين IP-Based التقليديّة و NDN-Based الجديدة، كما قُمنا بتحديد كل التحسينات التي سبّبتها البُنية NDN ضمن هذه البُنية الجديدة وعمِلنا على قياس أهم معايير الأداء ووضع التفسيرات المُناسبة لكل تحسّن تم إيجادُه.

نتطلّع مُستقبلاً إلى التطوير على الشِيفرة المصدريّة الخاصة بالبُنية SD-NDN المُوزّعة بحيث تُصبح داعِمة لتخصيص خوارزميات مُوازنة الحِمل بين المُتحكُّمات وتحديد أي وسطاء تتعلّق بذلك. يمكن أن تُجرى أعمالاً على هذا البحث مثل تطبيق البُينة-SD NDNعلى بُنية مُوزّعة تشابكية هرميّة، وتطوير الشيفرة المصدريّة للبُنية المُوزّعة حتى تدعم (بالإضافة إلى خوارزميات مُوازنة الحِمل) تحديد نسبة تدفّقات الرُزم ضِمن كل مُتحكُّم.

٩- جدول المُختصرات:

CCN	Content-Centric Network
CCP	Cache replacement policy based-on Content Popularity
CS	Content Store
FIB	Forward Information Base
ICN	Information-Centric Network
NCP	Network Control Protocol
NDN	Named Data Network
PIT	Pending Interest Table
SDN	Software-Defined Network
SD-NDN	Software-Defined Named Data Network

١٠- المراجِع:

[١] A. Marica ،C. Claudia ،R. Giuseppe ،M. Antonella و I. Antonio ، "Understanding Name-based Forwarding Rules in Software-Defined Named Data Networking "،*University of New South Wales. Downloaded on September 26,2020 at 18:20:29 UTC from IEEE Xplore* .٢٠٢٠ ،.

[٢] L. R. M. André ،R. C. Francisco ،B. A. Guilherme و N. S. Leobino ، "NDN Fabric: Where the Software-Defined Networking Meets the Content-Centric Model "،*IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, VOL. 18, NO. 1* ،March 2021 .

[٣] Y. Sembati ،N. Naja و A. Jamali ،"A review of SDN-enabled routing protocols for Named Data Networking "،*Engineering Reports published by John Wiley & Sons Ltd* ٦ ،.March 2024 .

[٤] A. Albatool و B. Abdelfettah ،"Software-Defined Named Data Networking in Literature: A Review "،*Future Internet 2024, 16, 258* ،. ٢٣July 2024 .

[٥] C. Lorenzo ،H. Mariann ،L. Lars ،S. Niranjan و T. Mauro ، "Considerations on the Adoption of Named Data Networking (NDN) in Tactical Environments "،*International Conference on Military Communications and Information Systems (ICMCIS)* ، ٢٠١٩.

[٦] M. Marion ،Development of a performance measurement tool for SDN ،٢٠١٥.

[٧] S. S. M.Micheal و Dr.K.Kumar. ،"A SURVEY ON NAMED DATA NETWORKING "،*IEEE SPONSORED 2ND INTERNATIONAL CONFERENCE ON ELETRONICS AND COMMUNICATION SYSTEM* ، ٢٠١٥.

[٨] J. Zain ul Abidin ،A. Zeeshan و T. Muhammad ،"A Named Data Networking (NDN), New Approach to Future Internet Architecture Design: A Survey "،*International Journal of Informatics & Communication Technology (IJ-ICT) Vol.2, No.3* ،Desemper 2013 .

[٩] د. ي. غازي و م. ع. الحمصيه، "سياسة مُحسّنة لاستبدال ذاكرة التخزين المُؤقت في شبكات البيانات المسمَّاة،" *مجلة جامعة حمص المجلد ٤٦ العدد ٥*، ٢٠٢٤.

[١٠] S. Kalafatidis ،S. Sotiris ،D. Vassilis ،M. Lefteris و T. Vassilis ، "Logically-Centralized SDN-Based NDN Strategies for Wireless Mesh Smart-City Networks "،*Future Internet 2023, 15, 19. https://doi.org/10.3390/fi15010019* ٢٩ ،December 2022 .

[١١] "Named Data Networking،" Wikipedia article.

[١٢] R. S. Nana ،T. Harashta ،M. Alvin و P. Fajar ،"Performance Comparison of Named Data and IP-based Network—Case Study on the Indonesia Higher Education Network "،*Journal of Communications Vol. 13, No. 10* ،October 2018 .

[١٣] A. Manar ،A.-A. Dawood ،A. Medhat و K. Firdous ،"Leveraging ICN and SDN for Future Internet Architecture: A Survey "،*Electronics 2023, 12, 1723. https://doi.org/10.3390/electronics12071723* ٤ ، April 2023 .

[١٤] X. Wenfeng ،W. Yonggang ،S. M. IEEE ،H. F. Chuan ،S. M. IEEE ،N. Dusit ،M. IEEE ،X. Haiyong و M. IEEE. ،"A Survey on Software-

Defined Networking "،*.COMMUNICATION SURVEYS & TUTORIALS, VOL. 17, NO. 1* ،FIRST QUARTER 2015 .

[١٥] و POXم. الضاهر، "تحليل أداء وحدات تحكم الشبكات المعرفة برمجياً: Opendaylight .٢٠٢١ ،*مجلة جامعة حمص المجلد ٤٣ العدد ١٧*، "،

[١٦] "https://ndnsim.net/current/intro.html .[متصل] "،