Performance Evaluation of Application Layer Protocols in Internet Of Things IOT

Dr. Mohammed Hijazieh[*]
Noha Hammami[**]

(Received 13 / 7 / 2020. Accepted 30 / 8 / 2020)

□ ABSTRACT □

The emerging technologies of the Internet Of Things (IOT) offer a new opening in the world of applications, as it affects the quality of life in a revolutionary way, as it interferes with various fields such as health care, environmental monitoring, military and economic fields, industrial automation and many other fields. The communication everywhere is one of the critical requirements in the Internet of things, so several suitable communication protocols have recently been developed. And its study is one of the most important fields that research have been interested in recently.

Hence the importance of this research lies in the study of performance of the newly developed Coap and Mqtt protocols within the application layer, which can be used in various applications of Internet of Things. The research also draws necessary recommendations and suggestions for the selection of the most appropriate protocol in terms of performance.This has been done by proposing appropriate scenarios for IoT applications to perform experiment that is similar to the working conditions of this protocols in the applied field to study the impact of a set of different parameters on the performance and behavior of protocols, Specifically the size of sent message , reliability levels of protocols, and the time interval applied between sent messages. performance was evaluated based on the basic parameters in such protocols, such as size of the data exchanged and overhead and efficiency and end to end delay . The study found a set of recommendations regarding choosing the most suitable protocol for work.

Keywords: Internet of things, internet of things applications, Coap, Mqtt

[*]**Associate Professor, Department of Computers and Automatic Control, Faculty of Mechanical and Electrical Engineering, Tishreen University, Lattakia, Syria.**

[**]**Postgraduate Student (Master), Department of Computers and Automatic Control, Faculty of Mechanical and Electrical Engineering, Tishreen University, Lattakia, Syria.**

تقييم أداء بروتوكولات طبقة التطبيقات في إنترنت الأشياء

د. محمد حجازية*

نُهى حمّامي**

(تاريخ الإيداع 13 / 7 / 2020. قُبِل للنشر في 30/ 8 / 2020)

□ ملخّص □

تقدم التقنيات الناشئة في إنترنت الأشياء فتحاً جديداً في عالم التطبيقات، إذ أنّها تؤثر على نوعيّةِ الحياةِ بطريقةٍ ثوريّةٍ، فهي تتداخل مع مجالات مختلفة كالرعاية الصحية والمراقبة البيئية والمجالات العسكرية والاقتصادية والأتمتة الصناعية ومجالات عديدة أخرى. ويعد الاتّصال في كلِّ مكانٍ أحدَ المتطلّباتِ الحاسمةِ في إنترنتِ الأشياء لذلك طُورتْ مؤخراً عدةُ بروتوكولاتِ اتصالٍ ملائمةً، تعد دراستُها من أهم المجالات التي تهتم بها الأبحاث في الآونة الأخيرة. ومن هنا تأتي أهميةُ هذا البحث الذي يهدف إلى دراسةِ أداء البروتوكولات Coap و Mqtt المطورة حديثاً ضمن طبقة التطبيقات والتي يمكن أن تستخدم في شتى تطبيقات إنترنت الأشياء. وتم في هذا البحث استخلاص التوصيات والاقتراحات الضرورية من أجل اختيار البـروتوكول الأنسب من ناحية الأداء، ذلك من خلال اقـتراح السـيناريوهات المناسبة لتطبيقات إنترنت الأشياء، مع تنفيذ التجارب اللازمة في ظروفِ عملٍ مشابهة لعمل البروتوكولات في الحقل التطبيقي وتم دراسة تأثيرِ مجموعة من المعاملاتِ المختلفة على أداء البروتوكولات وسلوكها، وعلى وجه التحديد دراسة تأثير حجم الرسالة المرسلة ومستويات الوثوقية والفاصل الزمني المطبق بين الرسائل المرسلة. حيث تم تقييم الأداء اعتماداً على البارامترات الأساسية في مثل هذه البروتوكولات، مثل حجم البيانات المتبادلة والحمل الزائد والكفاءة والتأخير الزمني، وتم التوصل في هذه الدراسة إلى مجموعة من التوصيات بخصوص اختيار البروتوكول الأنسب للعمل.

الكلمات المفتاحية: إنترنت الأشياء، تطبيقات إنترنت الأشياء، Coap، Mqtt.

* **أستاذ مساعد، قسم هندسة الحاسبات والتحكم الآلي، كلية الهندسة الميكانيكية والكهربائية، جامعة تشرين، اللاذقية، سورية.**

** **طالبة دراسات عليا(ماجستير)، قسم هندسة الحاسبات والتحكم الآلي، كلية الهندسة الميكانيكية والكهربائية، جامعة تشرين، اللاذقية، سورية.**

مقدمة:

تُمثل إنترنت الأشياء رؤيةً تمتدُ بها الإنترنت إلى العالم الحقيقي الذي يحتضنُ الأشياءَ اليومية حيث أنّه لم تعدْ العناصر المادية مفصولةً عن العالم الافتراضي، وإنّما يتم مكاملتها مع عالم الإنترنت بتزويدها بكيانٍ ماديٍ وآخرَ برمجيٍ لتكونَ ذكيةً وتمتلكُ القدرة على التواصل مع بعضِها البعضْ والتفاعلِ مع البيئة المحيطة والمشاركة بفعاليةٍ في جميع جوانب الحياة اليومية.[1][2]

وتتنّوعُ أنواعُ الأشياءِ من الحسّاسات إلى الحاسبات فالآلات. ويمكنُ أيضاً أن تتضمنَ مخلوقاتٍ حيةً مثل البشر والحيوانات والنباتات. على سبيل المثال؛ مشروعُ تتبُعِ البقر في مدينة Essex يَستخدمُ البيانات المجمّعة من قبل علاماتِ تحديد المواقع الراديوية Tags Positioning Radio لمراقبة البقر ومعرفة الأمراض وتتبع السلوك في القطيع. أيضاً التقنيات القابلة للارتداء وأجهزةُ الصحة الرقمية، مثل حزام Nike+ وFitbit، هي أمثلةٌ عن كيفية استخدام البشر لإنترنت الأشياء.[3]

ويعدُّ الاتّصال في كل مكان أحدَ المتطلّباتِ الحاسمةِ في إنترنت الأشياء، لذا طورت حديثاً عدة بروتوكولات اتصالٍ توسم بأنّها خفيفة الوزن lightweight وأكثرَ ملائمةً لمتطلبات الأجهزة المدمجة ذات الموارد المحدودة، وهي تعدّ مناسبة للعمل في البيئات المقيدة أكثر من البروتوكول التقليدي . [Http [2ومازال العمل مستمراً في الحقل البحثي للتوصل إلى بروتوكول إنترنت أشياء قياسي موحد لذلك يعتبر قياس وتقييم أداء بروتوكولات طبقة التطبيقات Coap و Mqtt من أهم المجالات التي تهتم الأبحاثُ والدراسات بها مؤخراً، لما للموضوع من أهمية ومردود على صعيد التوفير في الوقت والجهد والتكلفة عند تصميم وتنفيذ تطبيقات إنترنت الأشياء واختيار البروتوكول الأنسب لعملهم.

أهمية البحث وأهدافه:

تكمن أهمية هذا البحث في مساعدة مطوري تطبيقات إنترنت الأشياء على اختيار بروتوكول الاتصال المناسب للحصول على الأهداف المرجوة المتعلقة بالأداء وبالتالي تحسين تجربة المستخدم. غير أن هذا البحث يمكن أن يمثل نواة أساسية لأبحاث أخرى تهتم في تحسين أداء البروتوكولات قيد الدراسة.

يهدف البحث إلى قياس وتحليل أداء البروتوكولات Coap و Mqtt من خلال بناء وتنفيذ منظومات إنترنت أشياء تساعد بدورها في إجراء تقييم واضح لأداء البروتوكولات المدروسة ضمن سيناريوهات عمل مختلفة.

طرائق البحث ومواده:

تعتمد طريقة البحث على دراسة البروتوكولات Coap وMqtt، ودراسة آلياتِ عملِ كلٍ منهم ومقاييس الأداء (بارامترات الأداء) التي يمكن استخدامها أثناء الاختبار والقياس، وكذلك دراسة إمكانية تطبيق هذه البروتوكولات في بيئاتِ عملٍ مشابهة لبيئات عملهم الحقيقية لاختبارهم تجريبياً، إذ تم اقتراح عدة سيناريوهات لتطبيقات إنترنت الأشياء من خلال بناء وتنفيذ منظومات إنترنت أشياء تعمل في ظروف مشابهة للظروف التي تعمل بها البروتوكولات المدروسة، مع دراسة تأثير مجموعة من المعاملات المختلفة مثل حجم الرسالة ومستويات الوثوقية التي تقدمها البروتوكولات والفاصل الزمني المطبق بين الرسائل المرسلة.

وتم تحديد مراحل البحث بالخطوات الآتية:

- دراسة موقع البحث ضمن الأبحاث التي تعنى بحل نفس المشكلة أو المشاكل المشابهة.

- تثبيت بيئة العمل وتطوير التطبيقات اللازمة
- اختيار السيناريوهات المناسبة لعمل البروتوكولات انطلاقاً من واقع عمل بيئات إنترنت الأشياء.
- تشغيل بيئة العمل لكل سيناريو مدروس وقياس أداء البروتوكولات قيد الدراسة بمساعدة أداة التقاط الرزم [4] وايرشارك wireshark.
- مقارنة وتحليل أداء البروتوكولات واستخلاص النتائج والتوصيات.

الدراسات المرجعية:

اهتمت العديد من الأبحاث بدراسة البروتوكولات Coap وMqtt بشكل عام وبتقييم أدائهم ومقارنته بشكلٍ خاص:

فاهتم البحث [5] بإجراء مقارنة تجريبية ما بين أداء البروتوكولين Coap و Mqtt وأظهرت التجارب أنّ البروتوكول Coap أكثر فعالية من البروتوكول Mqtt من ناحية استهلاك الطاقة الكهربائية Energy consumption وحجم البيانات المتبادلة transferred data size وذلك عند عدم احداث خسارة في الرزم وعند احداث خسارة في الرزم محددة في البحث بنسبة20%. ذلك من أجل أحجام بيانات مرسلة لا تتجاوز 2000 بايت، دون أن يتطرق البحث إلى دراسة واختبار بارامترات أداء آخرى.

بينت التجارب في [6] أنّ أداء البروتوكولات المختلف يعتمد على تغير ظروف الشبكة المختلفة إذ أن للبروتوكول Mqtt زمن تأخير نهاية إلى نهاية end to end delay أصغر مقارنةً مع البروتوكول Coap وذلك عند عدم وجود خسارة في الرزم و عند احداث خسارة قليلة في الرزم low packet loss بينما يكون أكبر عند احداث خسارة في الرزم كبيرة high packet loss غير أنّ البروتوكول Coap يولد حركة مرور أقل less extra traffic من البروتوكول Mqtt ذلك عندما يكون حجم الرسالة صغيراً ومعدل خسارة الرزم loss rate يساوي أو أقل من 25%. دون أن يتطرق البحث إلى دراسة أثر تغيرات حجم البيانات المرسلة.

وأظهرت التجارب في [7] أن البروتوكولات Coap وMqtt تبدي سلوكاً مختلفاً تبعاً لظروف الشبكة المحيطة. حيث أن للبروتوكول Mqtt زمن تأخير نهاية إلى نهاية delay end to end أقل وانتاجيةً أعلىthroughput مقارنةً مع البروتوكول Coap وذلك أثناء العمل ضمن شبكة اتصال فضائي satellite لديها سويات تأخيرdelay وخسارة في الرزم packets loss. إلا أن أداء البروتوكولات تمّ تقييمه بمساعدة برنامج محاكاة شبكي ومنصة إنترنت أشياء مطورة.

كما بينت التجارب في [8] أن للبروتوكولCoap أداءً أفضل من البروتوكول Http من ناحية الآلية المعتمدة لنقل البيانات ومن ناحية تأخير الارسال Transmissions delay. لذلك يعدّ البروتوكول Coap مناسباً أكثر للعمل في الشبكات ذات الموارد المحدودة حيث أنّ ذلك من شأنه تحسين الأداء. إلا أنّ الدراسة تركز على مكاملة أجهزة إنترنت الأشياء في التجربة ودراسة سلوكها في سيناريوهات عدّة، فاقتصرت على قياس أداء البروتوكولات من ناحية تأخير الإرسال فقط وتحليل آلية نقل البيانات المعتمدة. ولم يتطرق البحث إلى دراسة أداء البروتوكول Mqtt.

وأظهرت النتائج في [9] أنَّ لطريقة إعداد الشبكة تأثير ملحوظ على أداء البروتوكول من ناحية زمن الذهاب والإياب Round Trip Time(RTT). على النقيض من ذلك لا يكون لها أية تأثير ملحوظ على كفاءة البروتوكولefficiency. وأظهرت النتائج أيضاً أن للبروتوكول Coap الكفاءة الأعلى مقارنةً مع بقية البروتوكولات المدروسة وأن للبروتوكول Mqtt ذو مستوى الوثوقية Qos1 الكفاءة الأصغر وهو الأعلى من ناحية زمن الذهاب والإياب. وذلك من أجل أحجام بيانات مرسلة لاتتجاوز100 بايت، أي لم يناقش البحث أثر حجم البيانات المرسلة أكبر من 100 بايت وأثر مستويات الوثوقية كافة على الأداء.

أعطى البحث [10] اهتماماً خاصاً لتنفيذ البروتوكولات ضمن منظومة للرعاية الصحية، وبينت التجارب أنَّ للبروتوكول Coap أداءً أفضل من البروتوكول Mqtt من ناحية زمن الذهاب والإياب ذلك عند عدم وجود خسارة في الرزم وعند وجودها بنسبة مئوية تتراوح ما بين]0 30] وفي ذات السياق إن لـ Coap الأداء الأفضل من ناحية عدد الرزم Packets في المناقلة transaction المرسلة. إلا أن البحث يقتصر على دراسة الأداء بالاعتماد على رسالة طبية ذات حجم محدد ومستويات وثوقية مناسبة لخدمات الرعاية الصحية، أي لم يتم التطرق إلى دراسة أثر حجم البيانات المرسلة وأثر مستويات الوثوقية كافة على الأداء.

قام الباحثون في [2] بإجراء دراسة تحليلية لتقانة إنترنت الأشياء لاسيما بروتوكولات الاتصال وتم التوصل في سياق البحث إلى أن البروتوكولات Coap وMqtt تعدُّ بروتوكولات طبقة تطبيقات في إنترنت الأشياء وأن لكل منهم معمارية برمجية مختلفة والعديد من الخصائص التي تميز أحدهما عن الآخر.

قام الباحثون في [11] بدراسة استهلاك البروتوكولMqtt للطاقة الكهربائية حيث تم مراقبة الاختلاف في الطاقة المستهلكة عند استخدام معايير مختلفة للبروتوكول. وبينت النتائج أنّ وجود عدد متزايد من الناشرين publishers الذين تتم خدمتهم بواسطة الوسيط(المخدم) نفسه يفرض عليه عبءً أكبر من ناحية المعالجة و استهلاك الطاقة.

أعطى البحث [12] اهتماماً خاصاً لدراسةً نموذج استهلاك البيانات في البروتوكول Mqtt. وتم التوصل نظرياً من خلال مراجعة الأدب إلى أن البروتوكول Mqtt أكثر فعالية من البروتوكول Http ذلك من ناحية استهلاكه لعرض حزمة الشبكة Bandwidth consumption حيث أنَّه يتميز باستهلاكه لعرض حزمة أقل بكثير. وتم التحقق تجريبياً في البحث من مزايا البروتوكول Mqtt وذلك بتحليل استهلاكه للبيانات أثناء عمله في تجربة منفذة ضمن البحث.

قام الباحثون في [13] بدراسة تأثير مستويات الوثوقية للبروتوكول Mqtt على زمن التأخير latency ودراسة تأثير قوة إشارة شبكة الواي فاي WIFI على الزمن أيضاً. وأظهرت النتائج أنَّ البروتوكول Mqtt ذو مستوى الوثوقية الأدنى لديه زمن التأخير الأدنى مقارنة مع مستويات الوثوقية الأعلى وأظهرت النتائج أيضاً أن البروتوكولين Coapو Mqtt يبديان زمن تأخير أقل عند العمل في بيئة تكون قوة الإشارة فيها مرتفعةً مقارنةً مع بيئة تكون قوة الإشارة فيها منخفضة أو أقل ارتفاعاً وأنه ليس من الواضح ما إذا كانت هنالك أية عوامل غير معروفة أثرت على النتائج. لذلك فإن أهمية هذا البحث تكمن في اعتماده على التنوع في ماهية تدفق البيانات فيما إذا كانت مرسلة بشكل مستمر أو بشكل متقطع منتظم وكذلك باستخدامه لكافة طرائق الارسال التي تزودها البروتوكولات؛ الموثوقة منها والغير الموثوقة، مع تركيزه على تغطية طيف من السيناريوهات تغطي أحجاماً مختلفة للبيانات المرسلة، ودراسة أداء البروتوكولات باستخدام مقاييس أداء متعددة.

بروتوكولات إنترنت الأشياء:

طُور مؤخراً عدة بروتوكولات اتصال تعمل ضمن طبقة التطبيقات وتعد ملائمة لتطبيقات إنترنت الأشياء ذات الموارد المحدودة، ومن أشهرها البروتوكولات Coap وMqtt

1- بروتوكول التطبيقات المقيدة CoAP (Protocol Application Constrained):هو بروتوكول لنقل البيانات عبر الشبكة العنكبوتية يعمل ضمن طبقة التطبيقات وهو مصمم للعمل ضمن الأجهزة والشبكات المقيدة.[14]

2-البروتوكول(MQTT(Transport MQ Telemetry: هو بروتوكول تراسل publish/subscribe بواسطة الإنترنت، يعمل ضمن طبقة التطبيقات من الموديل المرجعي OSI وهو مناسب للعمل ضمن الأجهزة والشبكات المقيدة [15]. يبين الجدول (1) مقارنة موجزة بين البروتوكولين Coapو Mqtt:

اسم البروتوكول:	Coap	Mqtt
بروتوكول طبقة النقل:	UDP	TCP
نموذج الارسال Messaging model:	1. طلب استجابة Request/ Response 2. مراقبة المورد Resource Observe (مشابه للنمط نشر/ اشتراك)	نشر/اشتراك Publish/Subscribe
الوثوقية reliability:	اختيارية: 1-المستوى غير الموثوق Non confirmable(Non) 2-المستوى الموثوق confirmable(Con)	مستويات للوثوقية ثلاثة: Qos level0-1(التسليم مرة واحدة على الأكثر At most once) Qos level1 -2(التسليم مرة واحدة على الأقل At least once) Qos level2 -3(التسليم مرة واحدة فقط Exactly once)
تنسيق الرسالة message format	ثنائي Binary	ثنائي Binary
حجم Header الرسالة:	4 bytes	2 bytes

الجدول(1): مقارنة بين البروتوكولات Coap وMqtt

البنية العامة لنظام إنترنت الأشياء:

تسمح منظومات إنترنت الأشياء للأجهزة المرتبطةِ مع بعضها بالتفاهم فيما بينها عبر بروتوكولِ اتصالٍ محدد. إذ تشمل هذه الأجهزة الحساسات والمشغلات وغيرهم. ولكن تتطلب تطبيقات إنترنت الأشياء الوصول إلى بيانات الحساسات، وفقاً لاحتياجاتها. لذلك تعد البوابة gateway أحد أهم المكونات الأساسية في شبكة إنترنت الأشياء إذ أنها تربط حقل الاستشعار والأجهزة مع التطبيقات التي تعمل في الخلفية؛ خدمات الويب مثلاً. حيث تقوم بتجميع بيانات الاستشعار ثم تشاركهم مع المخدم/تطبيق الوجهة بواسطة الإنترنت أو غيره من وسائل الاتصال؛ اللاسلكي مثلاً، مثال على ذلك بوابة للرعاية الصحية منصبة على الهاتف المحمول تُجمّع من الحساسات بياناتٍ صحيةٍ مختلفة مثل ضغط الدم ومستويات السكر، وتقومُ بمعالجتها وتحليلها ثم نشرها إلى تطبيقات الواجهة الخلفية (تطبيق مخصص للأطباء لمتابعة المرضى)عبر الإنترنت مثلاً[5]. يَعتِبر هذا البحث تلك البوابات عبارة عن أنظمة مدمجة محدودة الموارد تعمل بالبطارية ولديها قدرات تخزين ومعالجة محدودة لذلك تتطلبُ استخدامَ بروتوكول إنترنت أشياء مناسب لتستطيع العمل بالكفاءة المطلوبة. ويركز هذا البحث على دراسة نقل البيانات ما بين البوابة التي تجمع بيانات الاستشعار وما بين المخدم، ولأنّ البوابة افُترض أنها ذات موارد محدودة فلا بد لها من الاعتماد على بروتوكولات مناسبة مثل البروتوكولات قيد الدراسة. وقد تم استخدام البروتوكول Coap مع نموذج الارسال مراقبة المورد Resource Observe المشابه لنموذج نشر اشتراك Publish/Subscribe المعتمد في البروتوكول Mqtt.

بنية منظومة إنترنت الأشياء في البحث: يمثل الشكل(1) منظومة إنترنت الأشياء المطبقة ضمن البحث، إذ أن المخدم عبارة عن حاسب شخصي laptop بمواصفات تقنية؛ معالج core2due وذواكر رام 2غيغابايت، أما البوابة هي أداة الراسبيري باي raspberry pi من الإصدار B 3 [16] وهي تعد أحد أشهر أجهزة إنترنت الأشياء. وقد تم إعداد وتهيئة الراسبيري باي وتنصيب المخدمات البرمجية coapthon[17] و Mosquitto [18] الخاصة بالبروتوكولات Coap و Mqtt على التوالي وذلك على الحاسب الشخصي، وتم تطوير تطبيقات العملاء وتنصيبهم على الراسبيري باي حيث أن العميل المرسل هو برنامج يحاكي عمل الحساس، أما العميل المستقبل هو عبارة عن تطبيق مهتم بالبيانات الواردة من الحساس، وتم تشغيل بيئة العمل بعد تهيئة الاتصال ما بين الراسبيري باي والحاسب الشخصي بما في ذلك الاتصال اللاسلكي عن طريق أداة الربط الشبكية الراوتر وفتح المنافذ 5683 و1883 الخاصة بالبروتوكولات Coap و Mqtt على التوالي.

الشكل(1): منظومة إنترنت الأشياء

الدراسة التجريبية:

لاختبار أداء البروتوكولات تم اقتراح سيناريوهات عمل مشابهة لعملهم في الحقل التطبيقي وتصميم منظومات إنترنت أشياء مخصصة لذلك، إذ تعتمد المنظومات هذه على البروتوكولات Coap و Mqtt في طبقة التطبيقات وذلك بدوره يساعد في قياس أداء البروتوكولات المدروسة. ويعد السيناريو الأول تطبيقاً تدفق البيانات فيه مستمر وهو بذلك يتشابه في آلية عمله مع بعض تطبيقات إنترنت الأشياء مثل تطبيقات الرعاية الصحية، حيث تُرسل الحساسات الطبية البيانات بشكل مستمر إلى تطبيق الوجهة؛ تطبيق مراقبة الحالة الصحية مثلاً. ويعد السيناريو الثاني تطبيقاً تدفق البيانات فيه متقطعاً منتظماً وهو بذلك يتشابه في آلية عمله مع تطبيقات إنترنت الأشياء في المزارع الذكية إذ ترسل فيها الحساسات بيانات الحرارة بشكل منتظم إلى تطبيق الوجهة؛ تطبيق مراقبة الحرارة مثلاً.

السيناريو الأول

تم تنفيذ التجربة كتنفيذ لإرسال البيانات بشكل مستمر إذ تمّ ارسال عشر رسائل متتالية، وتم دراسة أداء البروتوكولات من أجل كافة مستويات الوثوقية التي تدعمها ومن أجل أحجام بيانات مختلفة حيث أنه تم تقسيم البيانات إلى مجموعتين:

- **مجموعة A وهي مجموعة للبيانات ذات الأحجام الصغيرة: 10,100,250,500,750,1000 بايت**
- **مجموعة B وهي مجموعة للبيانات ذات الأحجام الكبيرة: 2000,5000,10000,20000,50000 بايت**

السيناريو الثاني

تم تنفيذ التجربة كتنفيذ لإرسال البيانات بشكل منتظم (الارسال المتقطع) إذ تمّ ارسال عشر رسائل بينهم فواصل زمنية متساوية، وتم دراسة أداء البروتوكولات من أجل كافة مستويات الوثوقية ومن أجل فواصل زمنية مختلفة: **0.1 ثانية و0.5 ثانية و1 ثانية و1دقيقة و5دقائق**

استُخِدم في تقييم الأداء البارامترات حجم البيانات المتبادلة والحمل الزائد والكفاءة والتأخير الزمني وذلك لكونها تعطي تصوراً كافياً وواضحاً عن هذه البروتوكولات.

حجم البيانات المتبادلة transferred data size [12][6][5]: هو عدد بايتات البيانات المولدة لنقل الرسالة مضافاً إلى بايتات الرسالة الأصلية، إنّ حجم البيانات المتبادلة يعدّ مؤشراً على استهلاك البروتوكول لعرض الحزمة.

الحمل الزائد overhead: يُمثل بالنسبة بين حجم البيانات المنقولة الغير المفيدة والحجم الكلي للبيانات المنقولة؛ حيث أنّ البيانات المنقولة الغير مفيدة تمثل بايتات التحكم ويمكن حسابها بطرح حجم الرسالة user data من الحجم الكلي للبيانات المنقولة.

كفاءة البروتوكول efficiency[9]: تُعرَف كفاءة البروتوكول بأنها النسبة ما بين حجم البيانات المفيدة و حجم البيانات المنقولة في طبقة التطبيقات وطبقة النقل.

التأخير الزمني نهاية إلى نهاية end to end delay: هو الزمن المنقضي من لحظة إرسال المرسل للرسالة ولحظة استقبال المستقبل لها. وقد تم الاعتماد في تقييم هذه البارامترات على دراسة تغيراتها مع تغير حجم الرسالة المرسلة ومستويات الوثوقية التي تقدمها البروتوكولات عند الارسال والفاصل الزمني المطبق بين الرسائل المرسلة.

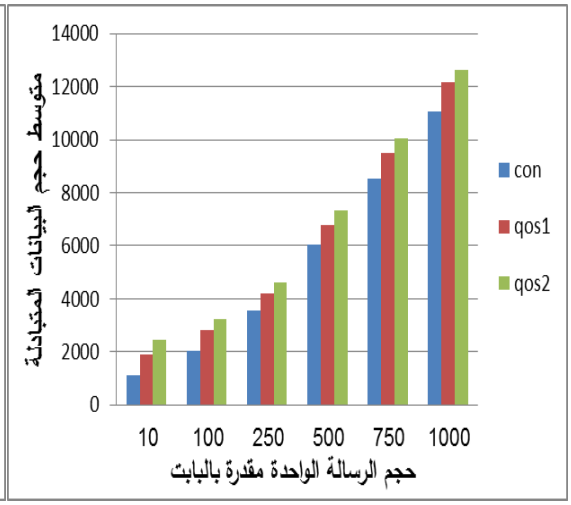
النتائج والمناقشة:

طُبقت التجربة وفقاً للسيناريوهات المبينة أعلاه وذلك بتكرار التجربة لعشر مرات لكل حالة وأُخذت القيم المتوسطة لمجموع القيم الملتقطة بواسطة أداة التقاط الرزم وايرشارك والناجمة عن تنفيذ الحالات المدروسة، حيث استخلصت جداول بقيم بارامترات التقييم التي عرضت سابقاً من أجل كل سيناريو ومن أجل كل تكرار. وتم بعدها أخذ المتوسط الحسابي لقيم البارامترات المذكورة. ومن ثم رُسمِت المخططات البيانية التي تبين تغير أداء البروتوكولات في السيناريوهات الموضوعة، وأُجريت بعدها مقارنة لأداء البروتوكولات ومفاضلة فيما بينها.

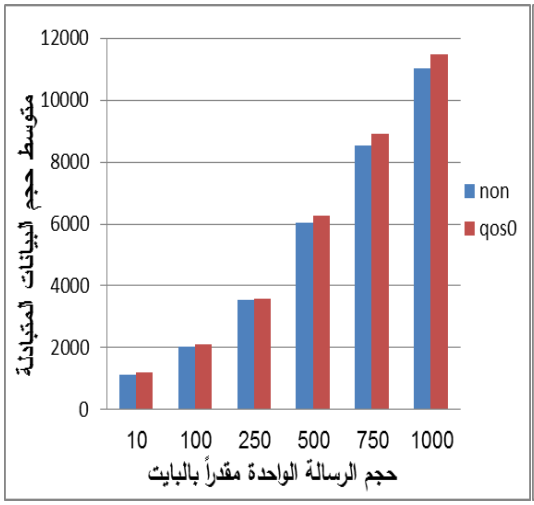
مقارنة الأداء في السيناريو الأول: تم في السيناريو الأول دراسة تأثير تبدل حجم البيانات المرسلة ومستويات الوثوقية على أداء البروتوكولاتCOAP وMQTT. واختبار الأداء وفق البارامترات حجم البيانات المتبادلة والحمل الزائد والكفاءة والتأخير الزمني.

1-حجم البيانات المتبادلة: i.حجم البيانات المتبادلة للبروتوكولات عند ارسال عشر رسائل كل منها ذات حجم صغير متساوٍ: من أجل عشر تكرارات للتجربة يبين الشكل(2) متوسط حجم البيانات المتبادلة كتابع لحجم الرسالة الواحدة المرسلة عشرَ مراتٍ بشكل مستمر، وتأخذ قيماً مختلفة مقدرةً بالبايت وتنتمي لمجموعة البياناتA ذات الأحجام الصغيرة ذلك من أجل مستويات الوثوقية المتقابلة Non وQos0 للبروتوكولات Coap وMqtt على التوالي، ويبين الشكل (3) متوسط حجم البيانات المتبادلة بدلالة حجم الرسالة المرسلة عشر مرات وتنتمي لمجموعة البياناتA، من أجل

مستويات الوثوقية المتقابلة con و qos1 للبروتوكولات Coap و Mqtt على التوالي ومن أجل qos2 مستوى الوثوقية الأعلى في Mqtt. علماً أن الارسال تم من العميل المرسل إلى المخدم.



الشكل(2): متوسط حجم البيانات المتبادلة بدلالة حجم الرسالة الواحدة صغيرة الحجم من أجل مستويات الوثوقية non وqos0

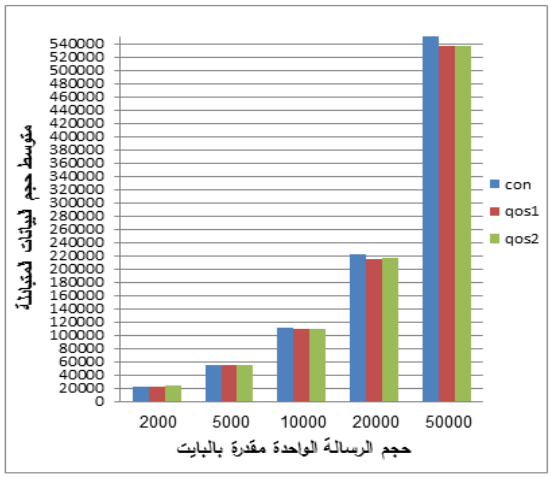


الشكل(3):متوسط حجم البيانات المتبادلة بدلالة حجم الرسالة الواحدة صغيرة الحجم من أجل مستويات الوثوقيةCon و qos1 و qos2

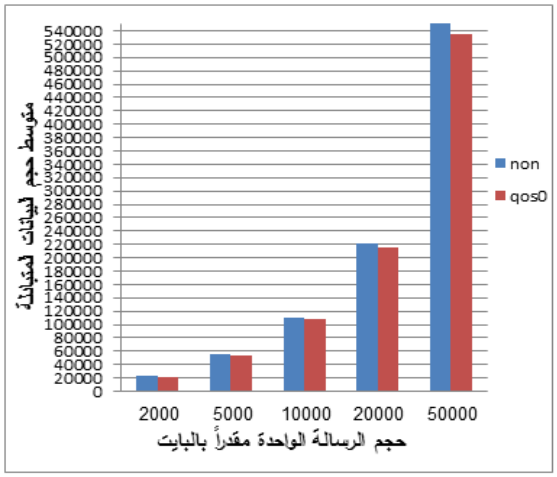
ويتضح مباشرة أنّه عند عمل البروتوكولات بطريقة إرسال البيانات بشكل غير موثوق أي عند استخدام المستويات Non و Qos0 كما هو مبين في الشكل (2). يستهلكُ البروتوكول Mqtt عرض حزمة عند الارسال أكثر من نظيره Coap، ويعود ذلك إلى اعتماده على البروتوكول TCP في نقل البيانات بينما يعتمد Coap على البروتوكول UDP الذي لديه ترويسة Header ذات حجم 8 بايت أصغر من ترويسة TCP ذات الحجم 32 بايت غير أنّ البروتوكول Tcp يتضمن رسائل مصافحة البداية Tcp-Syn ,Tcp-Ack Syn ورسائل الإنهاءTcp Finو Fin ack عند انتهاء الارسال وهذه الرسائل غير موجودة في Udp. ومن الشكل(3) يتضح مباشرة بمقارنة أداء البروتوكولCoapعند عمله بمستوى الوثوقية Conمع البروتوكولMqttعند عمله بمستوى الوثوقية المقابلqos1 إنّ البروتوكول Mqtt يستهلك عرض حزمة أكثر من نظيره Coap ويُعزى ذلك إلى أنّ آلية الوثوقية في Mqtt تتطلب بيانات أكثر من آلية الوثوقية المعتمدة في البروتوكول Coap غير أن Tcp لديه كلفة في نقل البيانات أكثر من Udp. وبمقارنة الأشكال (2) و(3) يمكن ملاحظة أنّ البروتوكول Mqtt عند عمله بمستوى الوثوقية الأعلى qos2 يستهلك عرض حزمة أكثر من عرض الحزمة المستهلك في مستويات الوثوقية الأدنىqos1 و qos0 والسبب يعود إلى رسائل تأكيد الوصول acknowledgmentالتي تفرضها آلية العمل في مستوى الوثوقية الأعلى qos2 وهي عبارة عن رسائل مصادقة Mqttفي طبقة التطبيقات تسمى PUBREC(Publish Received) وPUBREL(Publish Release) وPUBCOMP(Publish Complete)، ويكون لـ Mqtt استهلاك أكبر لعرض الحزمة عند عمله في مستوى الوثوقية qos1 مقارنةً مع عمله في مستوى الوثوقية الدنيا qos0 وذلك لأنه عند العمل بالمستوى qos0 يعتمد فقط على البروتوكول tcp لنقل البيانات بشكل موثوق ولايتكلف بأية رسائل مصادقة في طبقة التطبيقات، بينما عند العمل

بمستوى qos1 يعتمد أيضاً على طبقة التطبيقات إذ أنه يتبادل رسالة مصادقة هي رسالة Mqtt تسمى PUBACK (publish acknowledgment).

ii. حجم البيانات المتبادلة للبروتوكولات عند ارسال عشر رسائل كل منها ذات حجم كبير متساوٍ: من أجل عشر تكرارات للتجربة يبين الشكل (4) متوسط حجم البيانات المتبادلة من أجل أحجام رسائل مختلفة كل منها مرسل عشر مرات بشكل مستمر وينتمي لمجموعة البيانات B ذات الأحجام الكبيرة وذلك من أجل مستويات الوثوقية المتقابلة Non و Qos0 للبروتوكولاتCoap و Mqtt على التوالي. علماً أن الارسال من العميل المرسل إلى المخدم.



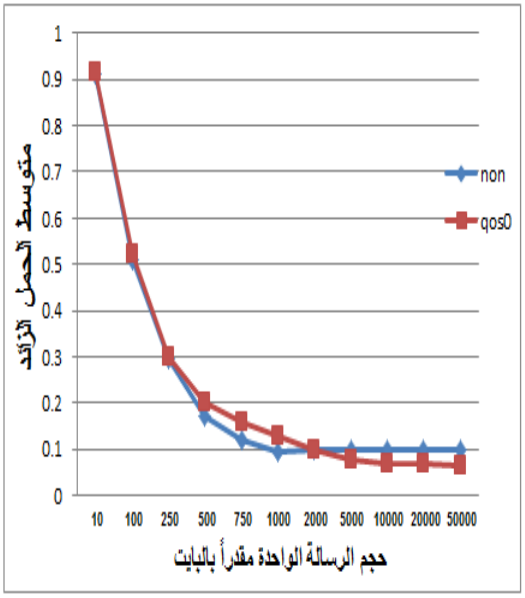
الشكل (4): متوسط حجم البيانات المتبادلة بدلالة حجم الرسالة الواحدة كبيرة الحجم من أجل مستويات الوثوقية Non و Qos0



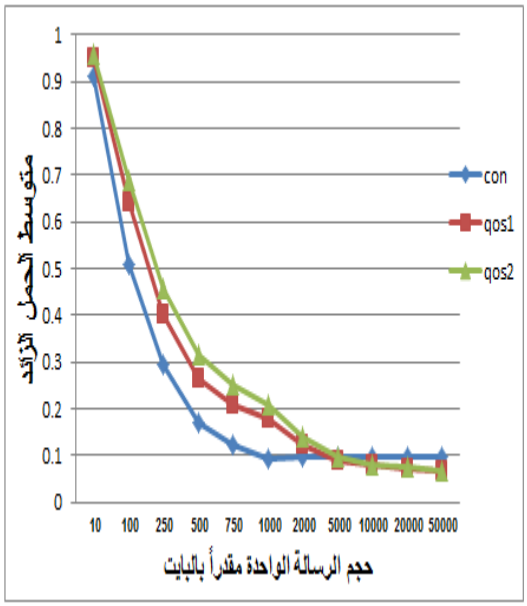
الشكل (5): متوسط حجم البيانات المتبادلة بدلالة حجم الرسالة كبيرة الحجم من أجل مستويات الوثوقية Con و Qos1 و Qos2

ويبين الشكل (5) متوسط حجم البيانات المتبادلة من أجل أحجام رسائل مختلفة كل منها مرسل عشر مرات بشكل مستمر وينتمي لمجموعة البياناتB ذات الحجم الكبير وذلك من أجل مستويات الوثوقية المتقابلة Con و Qos1 للبروتوكولات Coap و Mqtt على التوالي، ومن أجل Qos2 مستوى الوثوقية الأعلى في Mqtt. و بمقارنة الأشكال (2) و(4)، (3) و(5) يتضح مباشرة أنّه عند النقل الموثوق والغير موثوق للرسائل ذات الحجم الكبير يستهلك البروتوكول Coap عرض حزمة أكثر من Mqtt وذلك على نقيض سلوكه عند التعامل مع الرسائل ذات الحجم الصغير. وقد يعود ذلك إلى كلفة عملية تقطيع البيانات data fragmentation الأكبر عند Coap، ولأن Coap يمتلك ترويسة Header ذات حجم 4 بايت أكبر من ترويسة Mqtt ذات الحجم 2 بايت. الجدير ذكره أن عملية Block-wise هي عملية لتقطيع البيانات في Coap إذ أنّها آلية ينفذها Coap من أجل تقسيم البيانات الكبيرة إلى كتل blocks لإرسال وإعادة تجميع الكتل في طبقة التطبيقات عند الاستلام. ويمكن من الأشكال (4)و(5) ملاحظة أنّه عند نقل الرسائل ذات الحجم الكبير فإن Mqtt لديه عرض الحزمة الأكبر لدى عمله بمستوى الوثوقية الأعلى qos2، ويكون لديه استهلاك أكبر عند عمله في مستوى الوثوقية qos1 مقارنةً مع عمله بمستوى الوثوقية qos0. هذا يشير إلى أن البروتوكولات على اختلاف مستويات الوثوقية عند ارسالها بيانات ذات أحجام كبيرة تسلك السلوك نفسه الذي تسلكه عند إرسال بيانات ذات أحجام صغيرة، أي أنّ مستويات الوثوقية لها نفس التأثير على أداء البروتوكولات عند ارسال بيانات متنوعة الأحجام صغيرة وكبيرة.

2- الحمل الزائد: من أجل عشر تكرارات للتجربة ومن أجل عشر رسائل مرسلة بأحجام متساوية يُبيّن الشكل(6) متوسط نسبة الحمل الزائد للبروتوكولات كتابع لحجم الرسالة الواحدة مقدرةً بالبايت ومن أجل مستويات الوثوقية non و qos0. ويبين الشكل (7) أيضاً متوسط نسبة الحمل الزائد للبروتوكولات كتابع لحجم الرسالة الواحدة مقدرةً بالبايت ومن أجل مستويات الوثوقية Con و Qos1 و Qos2.



الشكل (6): متوسط الحمل الزائد للبروتوكولات بدلالة حجم رسالة واحدة من أجل مستويات الوثوقية Non و Qos0

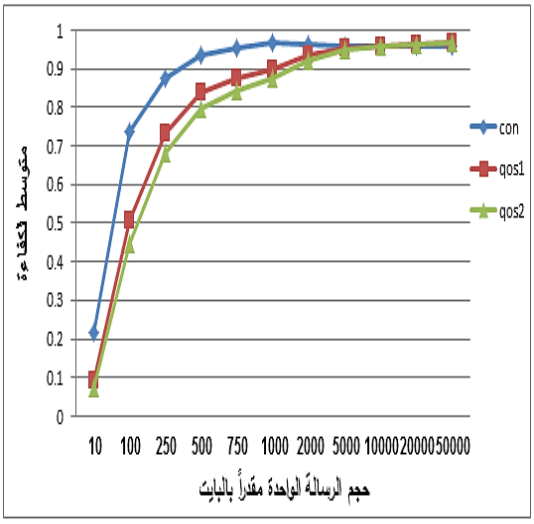


الشكل (7): متوسط الحمل الزائد للبروتوكولات بدلالة حجم رسالة واحدة من أجل مستويات الوثوقية Con و Qos1 و Qos2

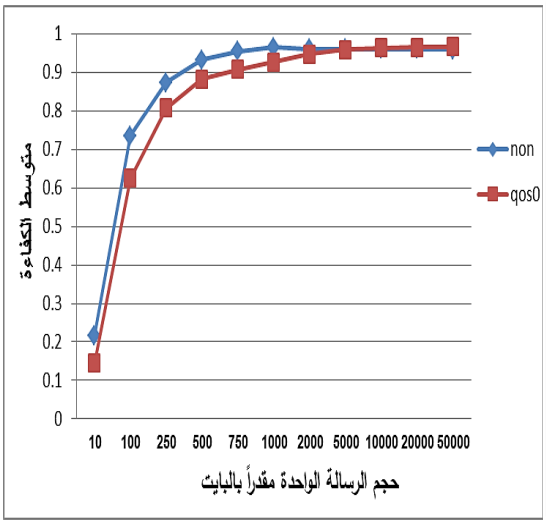
يوضح المثال التالي الحمل الزائد المحسوب؛ من أجل عشر رسائل مرسلة بشكل مستمر ومتساوية بالحجم ليكن 10 بايت فإن نسبة الحمل الزائد مساوية لـ $\frac{1160-100}{1160}$ أي ما يقارب 0.914، هذا يعني أنه من أجل نقل عشر رسائل بأحجام 10 بايت أي من أجل نقل 100 بايت فإنّ الحمل الزائد الكلي يكون بنسبة 0.914 من أجمالي البيانات المنقولة التي هي 1060 بايت. ويتضح مباشرة من الأشكال (6) و(7) أنّه من أجل النقل الموثوق والغير موثوق إنَّ البروتوكول mqtt لديه كمية حمل زائد أكثر من نظيره Coap ذلك من أجل أحجام الرسائل الصغيرة، وهذا يعود إلى اعتماده على البروتوكول TCP ذو حجم الترويسة الأكبر من ترويسة الـ Udp الذي يعتمد Coap عليه في النقل، غير أن Tcp لديه رسائل مصافحة عند البداية والنهاية بينما udp لا يحتويها. أما من أجل أحجام الرسائل الكبيرة فإن البروتوكولات تسلك سلوكاً مختلف إذ أنّ Coap لديه الحمل الزائد الأعلى والسبب قد يعود إلى كلفة عملية التقطيع وإلى أن لـ Coap ترويسة ذات حجم أكبر من ترويسة Mqtt. ويتبين أيضاً من الأشكال (6) و(7) أن البروتوكول Mqttعند عمله مع مستويات الوثوقية الأعلى qos1 وqos2 لديه overhead أكبر مما هو عليه عند استخدام مستوى الوثوقية qos0، ويعود ذلك إلى أن البروتوكول يستخدم من أجل مستويات الوثوقية qos1 وqos2 رسائل مصادقة في طبقة التطبيقات أيضاً، ويمكن ملاحظة أنّ هذه النتائج تتقاطع بشكل مباشر مع النتائج التي ظهرت عند قياس حجم البيانات المتبادلة؛ إذ أصبح واضحاً أنّه من أجل مجموعة البيانات A ذات الأحجام الصغيرة يمتلك البروتوكول Mqtt حجم بيانات متبادلة أكبر من نظيره Coap لذا يمكن بالعموم القول أن Mqtt يستهلك عرض حزمة

أكبر لأنّ الحمل الزائد لديه أكثر. وفي ذات السياق من أجل مجموعة البيانات B ذات الأحجام الكبيرة يمتلك البروتوكول Coap حجم بيانات متبادلة أكبر أي أنه يستهلك عرض حزمة أكبر من Mqtt ويعود ذلك إلى الحمل الزائد الأكبر لديه والذي تفرضه آلية عمله.

3-كفاءة البروتوكول من ناحية نقل البيانات: من أجل عشر تكرارات للتجربة ومن أجل عشر رسائل مرسلة بأحجام متساوية يبين الشكل(8) متوسط كفاءة البروتوكولات كتابع لحجم الرسالة الواحدة مقدراً بالبايت، وذلك من أجل مستويات الوثوقية Non و Qos0. ويبين الشكل(9) أيضاً كفاءة البروتوكولات كتابع لحجم الرسالة الواحدة مقدراً بالبايت، ذلك من أجل مستويات الوثوقية Con و Qos1 و Qos2.



الشكل (8): متوسط الكفاءة بدلالة حجم رسالة واحدة من أجل مستويات الوثوقية Non و Qos 0

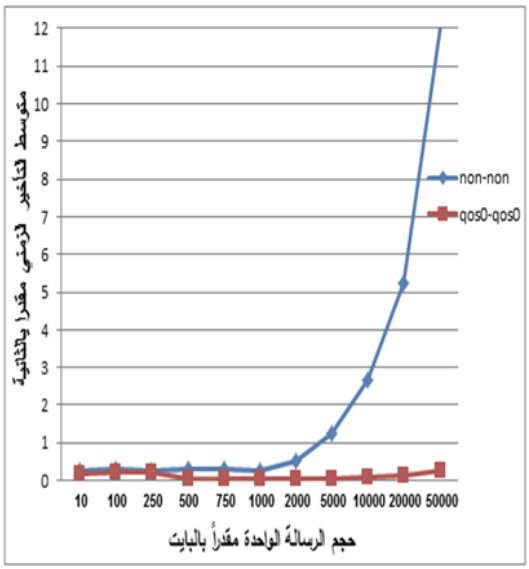


الشكل (9): متوسط الكفاءة بدلالة حجم رسالة واحدة من أجل مستويات الوثوقية Con و Qos1 و Qos2

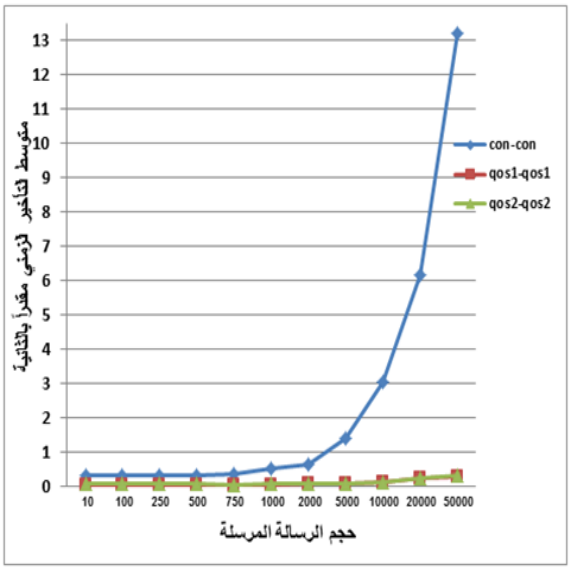
ويتضح مباشرة من الأشكال(8) و(9) أن كفاءة البروتوكول Coap أكبر من كفاءة البروتوكول Mqtt من أجل مجموعة البيانات A ذات الأحجام الصغيرة؛ ويعود ذلك إلى اعتماد Coap على البروتوكول Udp في النقل وهو بدوره لديه كلفة أقل من البروتوكول Tcp، إذ أنّه يمتلك ترويسة ذات حجم أصغر غير أنه لا يتطلب رسائل مصادقة ACK مثلما هو الحال في Tcp. يوضح المثال التالي الكفاءة المحسوبة؛ إذا كانت كفاءة البروتوكول مساوية لـ $\frac{1}{3}$ فهذا يعني أنه يوجد بايت واحد هو بايت معلومات مفيدة من أجل ثلاث بايتات متبادلة. ويتضح أيضاً أنه من أجل مجموعة البيانات ذات الأحجام الكبيرة B؛ تسلك البروتوكولات سلوكاً مختلفاً عن سلوكها عند العمل مع مجموعة البيانات A ذات الأحجام الصغيرة، إذ أنّ كفاءة البروتوكول Mqtt تكون الأعلى مقارنةً مع Coap. من ناحيةٍ أخرى يمكن ملاحظة أن البروتوكول Mqtt عند عمله مع مستويات الوثوقية الأعلى qos1 و qos2 لديه كفاءة أدنى مما هي عليه عند استخدامه لمستوى الوثوقية qos0، ويعود ذلك إلى أن Mqtt يستخدم من أجل مستويات الوثوقية qos1 و qos2 رسائل مصادقة Ack في طبقة التطبيقات أيضاً. ويمكن ملاحظة أنّ هذه النتائج تتقاطع مع النتائج التي ظهرت عند قياس حجم البيانات المتبادلة إذ أنّه من أجل أحجام بيانات صغيرة يستهلك البروتوكول Coap عرض حزمة أقل من Mqtt وبذلك يتميز بكفاءة أفضل، أما من أجل أحجام البيانات الكبيرة فيستهلك عرض حزمة أكبر ويتصف عندها بأنه ذو كفاءة أدنى. وهذا يقودنا إلى التنويه إلى أن كفاءة البروتوكول Coap تتناقص عند العمل مع أحجام البيانات الكبيرة

عن كفاءته عند العمل مع أحجام البيانات الصغيرة، وقد يعود السبب إلى كلفة عملية التقطيع التي يطبقها Coap عند التعامل مع الرسائل ذات الأحجام الكبيرة. أما Mqtt فهو على النقيض إذ أن كفاءته تزداد بازدياد حجم البيانات المرسلة.

4- التأخير الزمني نهاية إلى نهاية: من أجل عشر تكرارات للتجربة يبين الشكل (10) متوسط التأخير الزمني للبروتوكولات كتابع لحجم الرسالة الواحدة من أجل مستويات الوثوقية Non و Qos0. وذلك من أجل خمس رسائل بذات الحجم مرسلة من المرسل وخمس رسائل مستقبلة من المستقبل أي من أجل عشر رسائل متبادلة؛ ويبين الشكل (11) متوسط التأخير الزمني للبروتوكولات كتابع لحجم الرسالة الواحدة ذلك من أجل مستويات الوثوقية Con و Qos1 و Qos2.



الشكل (10): متوسط التأخير الزمني للبروتوكولات بدلالة حجم الرسالة الواحدة من أجل مستويات الوثوقية Non، Qos0

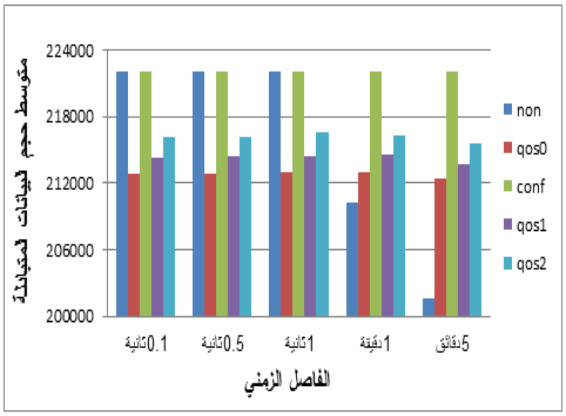


الشكل (11): متوسط التأخير الزمني للبروتوكولات بدلالة حجم الرسالة الواحدة من أجل مستويات الوثوقية Con,Qos1,Qos2

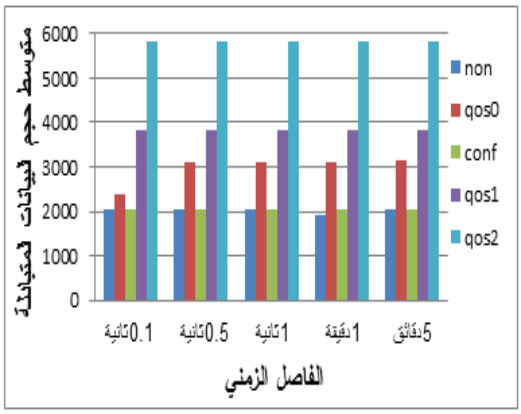
ويتضح مباشرة أنّه من أجل مستويات الوثوقية المختلفة وأحجام الرسائل المختلفة إنّ للبروتوكول Coap تأخيراً زمنياً أكبر من التأخير الزمني للبروتوكول Mqtt. والسبب قد يعود إلى اعتماد البروتوكول Coap عند استخدامه لنموذج التراسل مراقبة المورد resource observe على آلية لترتيب وإعادة ترتيب الرسائل Reordering وذلك لضمان وصول الرسائل والإشعارات بترتيبها الصحيح، لكونه يعتمد على البروتوكول Udp في طبقة النقل وهو بدوره لا يضمن وصول الرسائل مرتبةً إلى وجهتها. في المقابل يستخدم Mqtt البروتوكول Tcp الذي يدعم بدوره آلية لترتيب الرسائل وضمان وصولها مرتبةً إلى وجهتها لذلك Mqtt ليس بحاجة لآلية في طبقة التطبيقات تضمن ترتيب وإعادة ترتيب الرسائل. وبمقارنة الشكل (10) مع الشكل (11) يتبين أنّ البروتوكولات تبدي تأخيراً زمنياً أكبر من أجل مستويات الوثوقية الأعلى إذ أن Coap لديه التأخير الزمني الأكبر من أجل مستوى الوثوقية Con بالمقارنة مع مستوى الوثوقية Non، أما البروتوكول Mqtt فلديه التأخير الزمني الأعلى من أجل مستويات الوثوقية الأعلى Qos2 ثم Qos1 ثم Qos0 وذلك على الترتيب. ويعود السبب إلى أنّ البروتوكولات تستهلك عرض حزمة أكبر عند استخدامها مستويات الوثوقية الأعلى.

مقارنة الأداء في السيناريو الثاني: تم في السيناريو الثاني دراسة تأثير تبدل الفاصل الزمني المطبق بين الرسائل المرسلة على أداء البروتوكولات Coap و Mqtt. واختبار الأداء وفق البارامترات حجم البيانات المتبادلة والحمل الزائد والكفاءة والتأخير الزمني. وتمت الدراسة من أجل عينة من المجموعة A أي من أجل رسالة واحدة ذات حجم 100 بايت مرسلة عشر مرات بشكل متقطع منتظم، ومن أجل عينة من المجموعة B أي من أجل رسالة واحدة ذات حجم 20000 بايت مرسلة أيضاً عشر مرات بشكل متقطع منتظم من العميل المرسل إلى المخدم. الجدير ذكره أن الارسال لكونه متقطع منتظم يكون الفاصل الزمني المطبق بين الرسائل المرسلة نفسه طيلة فترة الارسال.

1- حجم البيانات المتبادلة: من أجل عشر تكرارات للتجربة ومن أجل فواصل زمنية مختلفة 0.1 و0.5 و1 ثانية و 1دقيقة و5 دقائق. يبين الشكل (12) متوسط حجم البيانات المتبادلة للبروتوكولات من أجل رسالة ذات حجم 100 بايت مرسلة عشر مرات بشكل متقطع منتظم وذلك من أجل كافة مستويات الوثوقية Non,Qos0 و Con,Qos1 و Qos2. ويبين الشكل (13) متوسط حجم البيانات المتبادلة من أجل رسالة ذات حجم 20000 بايت مرسلة عشر مرات بشكل متقطع منتظم، وذلك من أجل كافة مستويات الوثوقية Non,Qos0 و Con,Qos1 و Qos2.



الشكل (12): متوسط حجم البيانات المتبادلة عند ارسال عشر رسائل ذات حجم 100 بايت، من أجل فواصل زمنية مختلفة

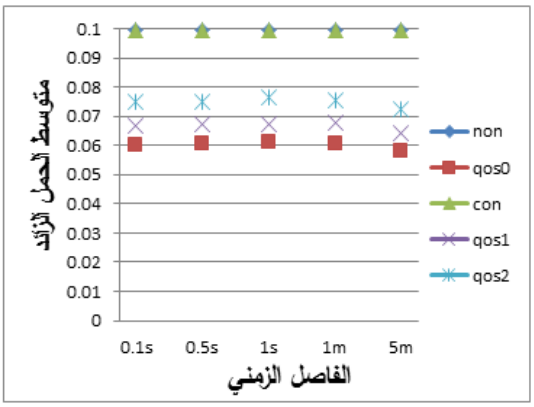


الشكل (13): متوسط حجم البيانات المتبادلة عند ارسال عشر رسائل ذات حجم 20000 بايت، من أجل فواصل زمنية مختلفة

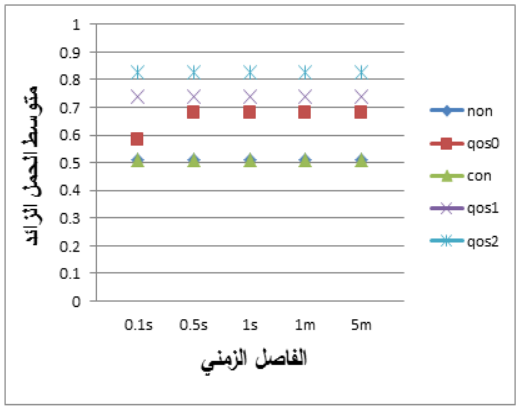
يتبين من الأشكال (12) و(13) بمقارنة أداء البروتوكولات عند عملها بمستوى وثوقية معين، من أجل فاصل زمني محدد، مع عملها بمستوى الوثوقية ذاته من أجل الفواصل الزمنية الأخرى، الحالة العامة هي أنَّ للبروتوكول حجم البيانات المتبادلة نفسها عند نقل الرسائل العشر، وذلك من أجل الفواصل الزمنية المختلفة المدروسة في البحث. ويعزى ذلك إلى عدم قيام البروتوكولات بإعادة تأسيس للاتصال مع بداية كل حيز زمني؛ إذ أن البروتوكول Coap يعتمد على البروتوكول Udp في نقل البيانات ونظراً لكونه بروتوكول غير موجه(بدون اتصال) connectionless؛ ليس لديه اغلاق للوصلة connection أي أنَه لا يملك أية رسائل إضافية لإعادة تأسيس الاتصال[19]. أما Mqtt فهو يعتمد على البروتوكول Tcp الذي يتميز بأنه بروتوكول موجه نحو الاتصال connection oriented [19] ، الجدير ذكره أنه في Mqtt يبقي على الوصلة مفتوحة بشكل مستمر أي Persistent connection إلى أن يقوم العميل المرسل بالانتهاء من ارسال أخر رسالة ويغلق الاتصال، لذلك فإن Mqtt لا يتكبد أية بيانات إضافية عند تغير الفاصل الزمني المطبق بين الرسائل المرسلة ولكن ذلك يكون عند تعطيل خاصية في Mqtt تدعى الإبقاء على قيد الحياة keep

alive time وهي آلية تستخدم في الاتصالات الفضائية والمحمولة لتفادي حصول انقطاع الاتصال وضياع البيانات، إذ أنه من أجل فترة إبقاء على قيد الحياة keep alive time افتراضية أي دقيقة واحدة يتم تبادل رسائل Mqtt تسمى ping request و ping response ما بين المرسل والمستقبل وذلك للتحقق الدوري من فعالية الاتصال. وذلك بدوره يجعل Mqtt يمتلك كلفة لنقل البيانات أكبر عند زيادة مدة الفاصل الزمني المطبق بين رسائله المرسلة. الجدير ذكره أنه في هذا النوع من التطبيقات يتم تعيين قيمة بارامتر الإبقاء على قيد الحياة بشكل متوافق مع قوة الإشارة والاتصال. ويتضح أيضاً من الأشكال (12) و(13) أنه يوجد حالات من التكرارات العشر لا يكون حجم البيانات المتبادلة فيها ثابتاً بتبدل الفاصل الزمني وهذا يعود إلى وجود بعض الحالات في Mqtt يحصل فيها تبادل عدد أقل من رسائل مصادقة Tcp أو ضياع لبعض رسائل مصادقة Tcp مما يتوجب إعادة ارسالها، ووجود بعض من الحالات في Coap يحصل فيها خسارة في البيانات بسبب طبيعة الاتصال في الشبكة.

2- الحمل الزائد: من أجل عشر تكراراتٍ للتجربة ومن أجل فواصلٍ زمنية مختلفة يبين الشكل (14) متوسط الحمل الزائد من أجل رسالة ذات حجم 100 بايت مرسلة عشر مرات بشكل متقطع وذلك من أجل كافة مستويات الوثوقية Non,Qos0 و Con,Qos1 و Qos2، ويبين الشكل (15) متوسط الحمل الزائد من أجل رسالة ذات حجم 20000 بايت مرسلة عشر مراتٍ وذلك من أجل كافة مستويات الوثوقية Non,Qos0 و Con,Qos1 و Qos2.



الشكل (14): متوسط الحمل الزائد عند ارسال عشر رسائل ذات حجم 100 بايت، من أجل فواصل زمنية مختلفة

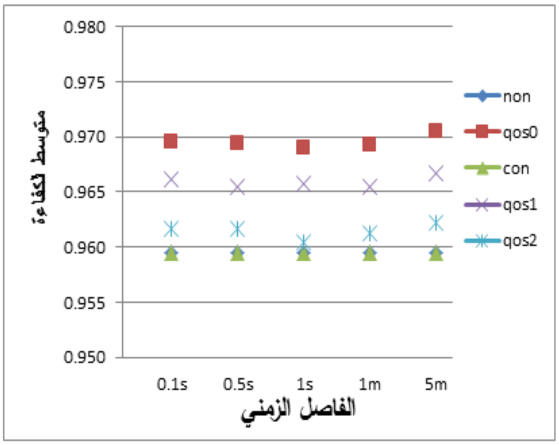


الشكل (15): متوسط الحمل الزائد عند ارسال عشر رسائل ذات حجم 20000 بايت، من أجل فواصل زمنية مختلفة

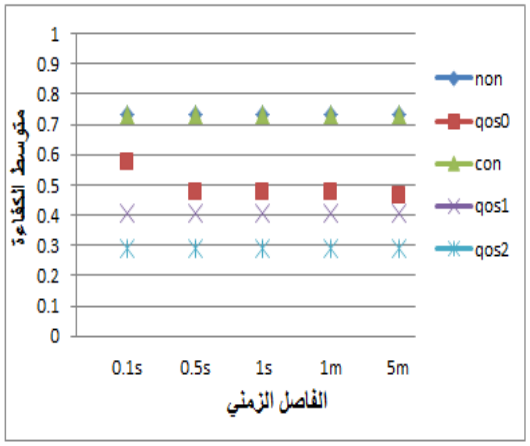
ويتضح مباشرة من الأشكال (14) و(15) أنّه في الحالة العامة لعمل البروتوكولات يكون الحمل الزائد للبروتوكول نفسه من أجل الفواصل الزمنية المختلفة، وإنّ ورود عدد من الحالات يختلف حجم البيانات المتبادلة فيها عن الحالة العامة الأساسية يوضح سبب الاختلاف في الحمل الزائد للبروتوكول بين فواصل زمنية مختلفة. على سبيل المثال كما مبين في الشكل(15) إن الحمل الزائد للبروتوكول Mqtt من أجل فاصل زمني 5 دقائق أقل من الحمل الزائد من أجل بقية الفواصل الزمنية وهذا يعود إلى ورود عدد من الحالات في التكرارات العشر للتجربة يتبادل البروتوكول Mqtt فيها رسائل مصادقة Tcp أقل مقارنةً مع عمله في الفواصل الزمنية الأخرى.

3-كفاءة البروتوكول من ناحية نقل البيانات: من أجل عشر تكرارات للتجربة ومن أجل فواصل زمنية مختلفة يبين الشكل (16) متوسط كفاءة البروتوكولات من أجل رسالة ذات حجم 100 بايت مرسلة عشر مرات بشكل متقطع منتظم

وذلك من أجل كافة مستويات الوثوقية Non, Qos0 و Con, Qos1 و Qos2. ويبين الشكل (17) متوسط كفاءة البروتوكولات من أجل رسالة ذات حجم 20000 بايت مرسلة عشر مرات بشكل متقطع منتظم وذلك من أجل كافة مستويات الوثوقية Non, Qos0 و Con, Qos1 و Qos2.



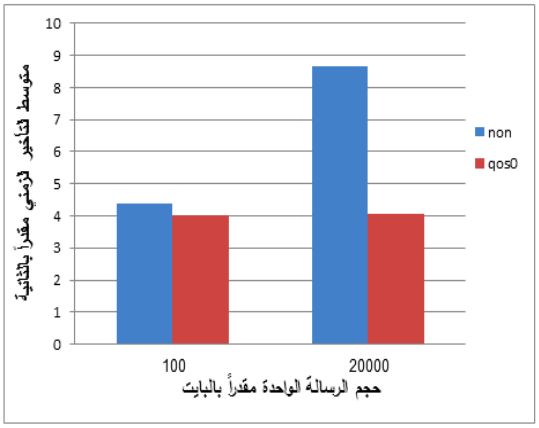
الشكل(16): متوسط الكفاءة عند ارسال عشر رسائل ذات حجم 100 بايت، من أجل فواصل زمنية مختلفة



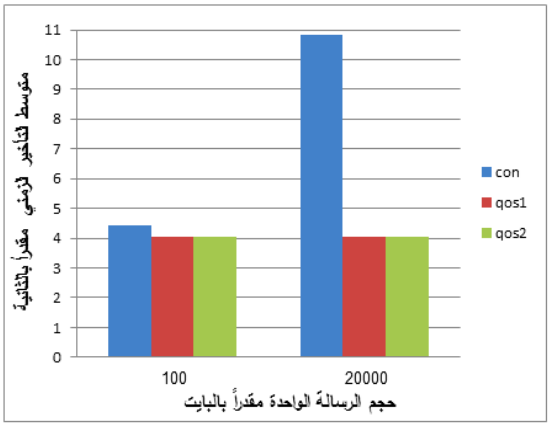
الشكل (17): متوسط الكفاءة عند ارسال عشر رسائل ذات حجم 20000 بايت، من أجل فواصل زمنية مختلفة

ويتضح مباشرة أن الحالة العامة وبتبدل الفاصل الزمني تكون كفاءة البروتوكول نفسها، أما من أجل عدد من الحالات وردت أثناء التجريب يختلف حجم البيانات المتبادلة فيها عن الحالة الأساسية فإن الكفاءة فيها تختلف أيضاً قليلاً عن الحالة العامة. على سبيل المثال كما مبين في الشكل (17) إن كفاءة البروتوكول Mqtt عند عمله بالمستوى qos0 من أجل فاصل زمني 5 دقائق أكبر قليلاً من الكفاءة من أجل الفواصل الزمنية الأخرى وذلك يعود إلى ورود عدد من الحالات في العشر تكرارات للتجربة يتبادل فيها البروتوكول Mqtt رسائل مصادقة Tcp أقل من الفواصل.

4- التأخير الزمني: تم قياس التأخير الزمني عند تبادل عشر رسائل بين العملاء المرسل والمستقبل كلٍ منها ذات حجم صغير 100 بايت، وعند تبادل عشر رسائل كل منها ذات حجم كبير 20000 بايت بينهم فاصل زمني قدره ثانية واحدة. من أجل عشر تكرارات للتجربة يبين الشكل (18) متوسط التأخير الزمني عند تبادل عشر رسائل ذات حجم صغير قدره 100 بايت ومن أجل كافة مستويات الوثوقية. ويبين الشكل (19) متوسط التأخير الزمني من أجل تبادل عشر رسائل ذات حجم كبير قدره 20000 بايت وذلك من أجل كافة مستويات الوثوقية Non, Qos0 و Con, Qos1 و Qos2.



الشكل(18): متوسط التأخير الزمني بدلالة حجم الرسالة الواحدة من أجل مستويات الوثوقية non و0 qos



الشكل(19): متوسط التأخير الزمني بدلالة حجم الرسالة الواحدة من أجل مستويات الوثوقية con, qos1, qos2

ويتضح مباشرة أنّ للبروتوكول Coap تأخيراً زمنياً أكبر من التأخير الزمني للبروتوكول Mqtt وذلك من أجل كافة مستويات الوثوقية. ويتبيّن عند مقارنة الأداء بين البروتوكولات Coap و Mqtt أنّ النتائج في السيناريو الثاني تتوافق وتؤكد نتائج السيناريو الأول.

الاستنتاجات والتوصيات:

بينت النتائج التي تم التوصل إليها في هذه الدراسة:

- يعتمد أداء البروتوكولات Coap و Mqtt على حجم البيانات المرسلة إذ أنه من أجل حجم رسائل صغيرة يستهلك Mqtt عرض حزمة أكثر من Coap أما من أجل حجم رسائل كبيرة يحصل العكس.
- يختلف أداء البروتوكولات Coap و Mqtt باختلاف مستويات الوثوقية التي تزودها إذ أنه كلما زاد مستوى الوثوقية يتطلب البروتوكول بيانات أكثر لنقل الرسائل بين تطبيقات العملاء، وهو بذلك يستهلك عرض حزمة أكثر.
- يتطلب البروتوكول Mqtt عند عمله في مستوى الوثوقية qos0 تبادلَ بياناتٍ أكثر من البروتوكول Coap عند عمله في مستوى الوثوقية Non وذلك عند ارسال رسالةٍ ذات حجم صغير وعند الارسال المستمر والمتقطع المنتظم للرسائل ذات الحجم الصغير، بينما Coap هو الذي يتطلب تبادلَ بياناتٍ أكثر عند إرسال رسالة ذات حجم كبير وعند الارسال المستمر والمتقطع المنتظم للرسائل ذات الحجم الكبير. ويتطلب Mqtt عند عمله في مستوى الوثوقية qos1 تبادلِ بياناتٍ أكثر من Coap عند عمله في مستوى الوثوقية Con وذلك عند ارسال رسالة ذات حجم صغير، وعند الارسال المستمر والمتقطع للرسائل ذات الحجم الكبير، بينما Coap هو الذي يتطلب تبادل بياناتٍ أكثر عند إرسال رسالة ذات حجم كبير وعند الارسال المستمر والمتقطع المنتظم للرسائل ذات الحجم كبير
- تعطي دراسة أداء البروتوكولات باستخدام بمقاييس أداء متعددة تصوراً أوضح وأدق من دراستها باستخدام مقاييس قليلة؛ إذا أنه تبين أنّ المقاييس حجم البيانات المتبادلة والحمل الزائد والكفاءة مرتبطة ارتباطاً وثيقاً فعندما يستهلك البروتوكول عرض حزمة أكبر يكون ذلك عائداً لتكلُّفِه بحمل زائد أكبر وتكون كفاءته عندها أقل مما لو كان يستهلك عرض حزمة أقل.

– يمتلك البروتوكول Mqtt آلية تسمى الابقاء على قيد الحياة keep alive time وهي آلية يتم تشغيلها والاستفادة منها في الاتصالات الفضائية والمحمولة، ويتبادل البروتوكول عند تشغيلها بياناتٍ أكثر مما لو تم تعطيلها. أي أنّ تفعيل هذه الآلية يقتضي بالضرورة استهلاك عرض حزمة أكبر.

– لا يؤثر تغير الفاصل الزمني المطبق بين الرسائل المرسلة على أداء للبروتوكولات Coap وMqtt، وذلك عند تعطيل آلية الإبقاء على قيد الحياة في البروتوكول Mqtt، وعند عدم وجود حالات خسارة في البيانات أثناء عمل البروتوكولات.

– يمتلك البروتوكول Coap تأخيراً زمنياً نهاية إلى نهاية أكبر من البروتوكول Mqtt وذلك عند عمله بنمط الارسال مراقبة المورد Resource observe المشابه لنمط الارسال نشر/اشتراك publish/subscribe الذي يعتمد عليه البروتوكول Mqtt.

وفقاً لنتائج البحث يمكن أن نوصي بضرورة مراعاة ماهية وخصوصية كل تطبيق، من ثم المفاضلة بين البروتوكولات واسناد البروتوكول الأنسب إذ أنه لا يمكن التعميم بأنّ أحد البروتوكولات هو الأفضل من أجل كافة التطبيقات. مثلاً من أجل تطبيقات الرعاية الصحية الحرجة، لكونها حساسة لوثوقية البيانات فمن الأفضل استخدام Mqtt لكونه يدعم مستويات وثوقية أكثر ولكن ذلك يكون على حساب الكلفة الأعلى في نقل البيانات، أما من أجل تطبيقات مراقبة دورية مثل مراقبة درجة الحرارة فإن Coap يمكنه أن يؤدي الغرض على أكمل وجه، بالمقابل في تطبيقات الاتصالات الفضائية من المؤكد أنّ Mqtt هو الأفضل، ونوصي أيضاً بضرورة الانتباه إلى حجم البيانات المرسلة وتأثيرها على أداء البروتوكول المستخدم واختيار البروتوكول تبعاً لمتطلبات التطبيق كأن يتطلب مثلاً كلفة منخفضة في الزمن بغض النظر عن استهلاك موارد الشبكة فإن البروتوكول Mqtt بالطريقة المنفذ بها في البحث يعدُّ الأنسب مقارنةً مع Coap المنفذ في البحث، أما إن كانت بيانات التطبيق صغيرة الحجم مثلاً والذي يهم هو الاستهلاك القليل لموارد الشبكة فكما تبين في البحث إنّ البروتوكول Coap يستهلك عرض حزمة أقل وبالتالي يعد الأنسب.

References:

[1] Mattern, F.; Floerkemeier, C. From the Internet of Computers to the Internet of Things. In: Sachs K., Petrov I., Guerrero P. (eds) From Active Data Management to Event-Based Systems and More, Lecture Notes in Computer Science, vol 6462. Springer Berlin Heidelberg, 2010

[2] Mensah, S. ; Laitinen, K. INTERNET OF THINGS:A Review on Connectivity Gateway Protocols and Semantic Interoperability, master thesis, Oulu University, Finland, 2017.

[3] Lopez Research LLC, An Introduction to the Internet of Things(IoT), 2013

[4] wireshark: packet analyzer, <https://www.wireshark.org/> (Online; accessed July 4, 2020).

[5] Bandyopadhyay, S. ;Bhattacharyya, A. Lightweight Internet protocols for web enablement of sensors using constrained gateway devices, Computing Networking and Communications (ICNC), 2013 International Conference on. IEEE, 2013, pp. 334–340

[6] Thangavel, D. and others; Performance evaluation of MQTT and CoAP via a common middleware, in 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing(ISSNIP), 2014, pp. 1–6-

[7] Collina, M.; Vanelli-Coralli, A. ; Caini, C.; Corazza; G., and Secchi; R. , Latency analysis of real-time web protocols over a satellite link, 31st AIAA International Communications Satellite Systems Conference, ICSSC, Italia, Oct 2013

[8] Farokhmanesh F. ,Ordinas, J, M. , analyzing and evaluating network protocols in iot, master thesis, Technical University of Catalonia (UPC), Barcelona, 2014

[9] Mijovic, S. ; Shehu, E. ; Buratti C. ; Comparing Application Layer Protocols for the Internet of Things via Experimentation, IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI),2016
[10] Oryema, B. ; Kim, H. ; Li, W.; et.al. Design and Implementation of an Interoperable Messaging System for IoT Healthcare Services, IEEE, 2017
[11] Viswanathan, A. ; Krishnamurthy, P. ; Analysis of Power Consumption of the MQTT Protocol, master thesis , University Of Pittsburgh , 2017
]12[Manohar, H. L. ; Reuban, G.A.T.; Data Consumption Pattern of MQTT Protocol for IoT Applications, springer، 2017
[13] Lagerqvist, A., Lakshminarayana, T., Day, R., IoT Latency and Power consumption:Measuringthe performance impact of MQTT and CoAP, master thesis, JÖNKÖPING University, Sweden , 2018
]14[Shelby, Z. ; Hartke, K. and Bormann, C. The Constrained Application Protocol)CoAP(,) Internet Engineering Task Force (Ietf)). Date of retrieval 1/9/2018 . <https://tools.ietf.org/html/rfc7252>
[15] International Business Machines Corporation (IBM), MQTT V3.1 Protocol Specification, Date of retrieval.2018/9/1
<http://public.dhe.ibm.com/software/dw/webservices/ws-Mqtt/Mqttv3r1 .html>
[16] <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/> (Online; accessed July 4, 2020)
[17] Tanganelli, G.; Vallati, C. ; Mingozzi E., CoAPthon: Easy development of CoAP-based IoT applications with Python , IEEE, 2015
[18] mosquitto: mqtt broker, <https://mosquitto.org/ >(Online; accessed July 4, 2020)
[19] Kurose, J.; Ross, k. Computer networking : a top-down approach.6th ed., Pearson Education, United States of America , 2013, 889.