# الحساب الآلي لطول الصف بغرض تقليل التأخير على الموجهات

Osama Ahmed Bashir, Yahia Abdalla Mohamed, Mohamed Awad

Computer Systems and Network Dept., Sudan University of Science and Technology (SUST), Khartoum, Sudan
jublabe@yahoo.com

*Received:28.05.2014*
*Accepted:05.07.2014*

**ABSTRACT -The delay experienced by packets on routers, has become a problem and it the first cause, which reduces the effective functioning of the Internet, this delay is due to the significant increase in the length of the queues on routers, the increase in buffer size by manufacturing in order to prevent the packets drops, forgetting that prevent the packet dropping lead to increase the delay, and the drop of the packets itself is important for the TCP protocol, in this paper we developed a new algorithm called the automatic calculation of the length of the queues (auto), the main work for this algorithm is to modify the length of the queue B automatically, according to the equation $B = \frac{RTT*C}{\sqrt{n}}$, where that C is the channel capacity and RTT is the time to send a packet and return Ack, and n is the number of connections probably estimated and not retain any information relating to connections, the difference between this algorithm and Controlling Queue Delay (Codel), is that the Codel algorithm deal with all packets entering to the router, while the new algorithm to estimate the length of the accurate queue, estimates the number of connections in the router probably, the performance of this algorithm has been compared to the performance of the (Codel) in addition to algorithms working in the field of congestion control, such as Drop-Tail and RED and found that the performance of the auto algorithm is always better with respect to congestion that causes Bufferbloat in addition to the preservation of the channel capacity.**

***Keywords:*** Bufferbloat, Codel, Bottleneck Link, Active Drop-Tail, Standing queue, AQM.

المستخلص - أصبحت مشكلة التأخير الذى تتعرض له الرسائل على الموجهات المسبب الأول الذى يقلل من فعالية عمل شبكة الانترنت، يعود هذا التأخير للزيادة الكبيرة فى طول الصفوف على الموجهات نسبة لزيادة الذواكر من قبل المصنعين وذلك لمنع سقوط الرسائل، متناسين ان منع سقوط الرسائل يؤدى الى زيادة التأخير وان سقوط الرسائل نفسه مهم لاستمرار عمل بروتوكول (TCP)، في هذه الورقه طورت خوارزمية جديدة سميت الحساب الآلى لطول الصف (auto)، فى هذه الخوارزمية يتم تعديل طول الصف B آلياً وذلك حسب المعادلة $B = \frac{RTT*C}{\sqrt{n}}$، حيث ان C هي سعة القناة وRTT هو زمن ارسال الرساله وعودة الاقرار عليها ويتم تقدير عدد الاتصالات n إحتمالياً من غير الاحتفاظ بأي معلومات تتعلق بالاتصالات الماره والفرق بين هذه الخوارزمية وخوارزمية التحكم بالتأخير (Codel)، هو ان خوارزمية (Codel) تتعامل مع كل الرسائل التى تدخل الى الموجه، في حين ان الخوارزميه الجديده تقوم لتقدير طول الصف المناسب بتقدير عدد الاتصالات التي تمر بالموجه وذلك بصوره احتماليه، تم مقارنة اداء خوارزمية auto مع اداء خوارزمية (Codel) بالاضافه الى الخوارزميات العامله في مجال التحكم بالازدحام مثل خوارزمية Drop-Tail وخوارزمية RED ووجد ان اداء الخوارزميه دائما افضل فيما يتعلق بالازدحام المسبب لتضخم الصفوف (Bufferbloat) بالاضافه الى محافظتها على سعة القناة.

## 1. المقدمه

تقدر الاستفادة من سعة القناة بمقدار مشغولية قناة عنق الزجاجة (Bottleneck Link)، حيث تحرص معظم بروتوكولات التحكم بالازدحام على جعل هذه القناة مشغولة معظم الوقت، ولجعل هذه القناة مشغولة يتم زيادة طول الصفوف على الموجهات لضمان عدم سقوط الرسائل، تسبب زيادة طول الصفوف زيادة التأخير الذي تتعرض له الرسائل في الصف عرفت هذه المشكله فيما بعد بمشكلة تضخم الصفوف (Bufferbloat) [1] ولكن ثبت فيما بعد ان الاستفادة القصوى من سعة القناة ليست هي العامل الحاسم لكفاءة شبكة الانترنت، حيث أن هناك قدر كبير من الرسائل التى تشغل القناة وهي عبارة عن رسائل مكررة ناتجة عن اعادة الارسال الناتج عن انتهاء المؤقتات عند المرسلين، لذلك كان هناك عامل آخر يسمى الاستفادة الفعلية للمستخدم (goodput) وهي كمية البيانات التي تم استلامها في المستقبل فعليا فى مقابل الاستفادة من سعة القناة (throughput) وهي كمية البيانات التي تم ارسالها من الموجه.

## 2. الاعمال السابقه

في هذه الفقره والتي تتحدث عن الخوارزميات السابقه التي تهتم بموضوع التاخير على صف الموجه تم اختيار خوارزميتين تعملان في هذا المجال هما اولا خوارزمية إدارة التأخير والتحكم به (Codel) وثانيا خوارزمية الاسقاط من آخر الصف ذات الضبط الآلي Active Drop-Tail ADT

### 1.2 خوارزمية إدارة التأخير والتحكم به (Codel)

خوارزمية إدارة التأخير والتحكم به (Codel) [2] تحتوى على ثلاثة مكونات اساسية تميزها عن خوارزميات ادارة الصفوف الفعاله (AQM) السابقة. أولاً هذه الخوارزمية لا تعتمد على

العوامل التاليه طول الصف، متوسط طول الصف، عتبه طول الصف، معدل الارسال، الاستفاده القصوى من سعة القناة، معدل الاسقاط او كمية الرسائل الموجودة بالصف، ثانيا: تستخدم هذه الخوارزمية الطول الاقصر للصف كتقدير جيد ودقيق للصف العامل "Standing queue" بعد ذلك يمكن استعمال متغير واحد لتحديد الطول الاقصر المناسب هل هو ادنى او اعلى من القيمة الهدف (Target) للصف العامل (Standing queue). أخيراً بدلاً من حساب طول الصف بالبايت او بعدد الرسائل فإنه يتم حساب زمن يسمى زمن زيارة الرسالة عبر الصف (packet-sojourn time). استعمال التأخير الفعلي الذي تختبره اي رساله لا يعتمد على معدل الارسال ويعطي اداره افضل من استعمال طول الصف وهو بالتالى مرتبط بصورة مباشرة بالكفاءة التى يراها المستخدم.

تقوم هذه الخوارزميه بتسجيل زمن وصول الرساله الى الموجه، اذا كان الصف ممتلئا عند ورود الرساله فإنه في هذه الحاله يتم اسقاطها. تفترض خوارزمية (Codel) ان الصف العامل ذو التأخير المستهدف (Target) مقبول ولايتم اسقاط اي رسالة من صف الموجه وذلك عندما يكون طول الصف يساوي رسالة واحدة فقط. تعتمد طريقة (Codel) فى تحديد التأخير الدائم بواسطة مراقبة التأخير الاقصى والذى تختبره الرسائل فى الصف، ولضمان ان الاقصرية لن تصبح قيمة غير ذات فائدة يلزم التأكد من القيمة الاقصريه داخل الفترة الزمنية الاقرب، عندما يتجاوز تأخير الصف الهدف (Target) فتره زمنية محددة فإن الرسالة يتم اسقاطها وبعد ذلك يتم حساب زمن الاسقاط التالى، زمن الاسقاط التالى يتناسب بصورة عكسية مع مربع عدد الرسائل المسقطة وذلك منذ دخول منطقة الاسقاط وعندما يصبح تأخير الصف ادنى من القيمة المستهدفه (Target) فإن المتحكم يوقف الاسقاط. هناك ايضا بعض الضمانات لعدم إعادة الدخول فى منطقة الإسقاط بعد فترة قصيره من الخروج من هذه المنطقه.

### 2.2 خوارزمية الاسقاط من آخر الصف ذات الضبط الآلي Active Drop-Tail ADT

تعتبر خوارزمية ADT [3] من اقدم الخوارزميات التي تهتم بمعالجة مشكلة زيادة التاخير، ظهرت هذه الخوارزميه قبل عملية استكشاف وظهور مشكلة تضخم الصفوف Bufferbloat، الفكره الاساسيه لهذه الخوارزميه هي السماح فقط للرسائل التي تحافظ على سعة قناة عنق الزجاجه في وضع جيد بالدخول الى صف الموجه واسقاط بقيه الرسائل، تعتبر هذه الدراسه ان الغرض الاساسي هو تقليل التاخير ثم بعد ذلك المحافظه على سعة القناة ولكن في الحقيقه الغرض الاساسي هو المحافظه على سعة القناة ثم تقليل التاخير في المرحله التاليه، ذلك واضح من طريقة عمل الخوارزميه، تعمل خوارزمية ATD في خطوتين الاولى تقدير متوسط الاستفادة من سعة القناة، والثانيه هي موائمة طول صف الموجه حسب متوسط الاستفادة من سعة القناة والذي يقاس دوريا.

تتميز هذه الخوارزميه بان لها آداء متماثل مع آداء خوارزمية الصفوف القصيره التي تنبني عليها الخوارزميه التي تم تطويرها في هذا البحث (auto)، ولكن خوارزمية ADT تتميز بان لها آداء معقد جدا ولا يمكن تطبيقها عمليا باي حال من الاحوال نسبه لتعقيد هذه الطريقه والموضحه في الشكل (1).

**ON EVERY PACKET ARRIVAL:**

*IF (now − LastUpdate) > SamplePeriod*

. $CURTHR := \frac{NmbBEnq(now)-NmbBEnq(LastUpdate)}{now-LastUpdate}$

. $THR := \rho\ CURTHR + (1-\rho)THR$

. $IF\ (\frac{THR}{ServiceRate} < u)$

. $qADT := qADT \cdot c$

. *OTHERWISE*

. $qADT := qADT\ \frac{qADT}{c}$

. *END*

. $qADT := min(qADT, SizeOfBuffer)$

. *LastUpdate := now*

*END*

**الشكل (1): الشفره الاوليه لخوارزمية ADT**

حيث ان p عباره عن معامل توسيط ياخذ القيمه بين (0-1) ويستخدم لحساب متوسط سعة القناة، c يكون بقيمه اكبر من الواحد وهو عبارة عن معامل دالة التحكم MIMD للتحكم بالمعامل qADT، SamplePeriod عباره عن فتره زمنيه لاخذ العينات، u عبارة عن سعة القناة المطلوبه، now زمن اخذ العينه الحالي، LastUpdate آخر زمن تعديل، CURTHR سعة القناة الحاليه، NmbBEnq(t) يحدد عدد البايت الداخله الى الصف في الفتره (t-0)، THR سعة القناة في زمن اخذ العينه السابق، qADT عباره عن معامل يحدد عدد الرسائل او يحدد طول الصف.

ينحصر تعقيد هذه الخوارزميه في نقطتين هما:

أ) عدد عمليات الجمع والطرح والضرب والقسمه والمقارنه التي توجد في هذه الخوارزميه والتي لابد من اجرائها عند دخول اي رساله الى الموجه، تؤدي هذه الخوارزميه الى حمل تنفيز كبير على معالج الموجه وهي بالتالي عمليه غير ممكنه لانه لا يمكن اشغال معالج الموجه المشغول اصلا بعمليه اخرى هي عملية التوجيه وهي العمليه الاساسيه للموجه بالطبع.

ب) المعاملات تحتاج قيم مختلفه تختلف باختلاف القناة التي يتحكم فيها الموجه، هذه المعاملات هي اولا معامل الفتره الزمنيه لاخذ العينه (sample period)، والمعامل c نصت الدراسه على ان يكون c بقيم اكبر من واحد، فعندما تكون سعة القناة الحاليه اصغر من سعة u المطلوبه، يكون المعامل c اكبر من الواحد ليتم ضربه في المعامل qADT، يؤدي هذا الى ان تزيد قيمة عدد الرسائل qADT ويؤدي هذه التقليل الى تقليل معدل الارسال. اما عندما تكون سعة القناة الحاليه اصغر من سعة القناة المطلوبه فيكون اصغر من الواحد، ليتم قسمة المعامل qADT على المعامل c يؤدي هذ الى ان تقل قيمة عدد الرسائل qADT و يؤدي هذه الى زيادة معدل الاسقاط.

## 3. التحليل

في الجزئيه الاولى من هذه الفقره تم تناول المعادلات الرياضيه لتحديد طول الصف في الخوارزميات الاكثر انتشارا وهي خوارزمية الصفوف القصيره وخوارزمية (Codel) وخوارزمية القاعده القديمه، حيث تم التوضيح بواسطة مثال رقمي ان اطوال الصفوف تتفاوت حسب الخوارزميه المستخدمه، وسنتناول في الجزيئه الثانيه من هذه الفقره مشاكل خوارزمية (Codel)، حيث ان هذه الخوارزميه مقترحه لحل مشكلة التأخير على الصف.

### 1.3 اتمته نظرية الصفوف القصيرة

الدراسه [4] تقول انه للاستفادة القصوى من سعة قناة عنق الزجاجة لابد ان يكون طول الصف مساوياً للزمن RTT مضروباً فى سعة القناة والتى تنتج صفوف تكون طويلة جداً وتؤدي الى تأخير كبير. حيث يكون طول الصف متناسب تناسباً طردياً مع سعة القناة فمع زيادة سعة القناة يزيد طول الصف

وبالتالى يزيد التأخير على الصف. جاءت بعد ذلك دراسة [5] اخرى تقول ان الدراسه السابقه [4] لا تكون صحيحه الا عندما يكون عدد الاتصالات قليلاً ولا تصح عندما يكون عدد الاتصالات كبيراً، وهي الحالة الاكثر شيوعاً ان يكون عدد الاتصالات كبيراً، أثبتت هذه الدراسة انه عندما يكون عدد الاتصالات كبيراً فان حجم صف صغير كافى تماماً للاستفاده من سعه قناة عنق الزجاجة ويؤدى ذلك ايضاً الى ان يكون التأخير الذي تتعرض له اي رسالة صغيراً. اذا فإن الصفوف القصيرة فى هذه الدراسة تؤدى الى الاستفادة من سعة القناة وتؤدى ايضا الى تأخير غير ملحوظ، فى الجانب الآخر كان الغرض الاساسى من خوارزمية (Codel) هو تقليل التأخير بعد إنتشار مشكلة تضخم الصفوف وايضاً الاستفادة من سعة القناة، اذا التساؤل هنا هو هل خوارزمية (Codel) هى عملية اتمته لخوارزمية الصفوف القصيره؟ اي جعل الصفوف قصيرة بصوره آليه للاجابة على مثل هذا التساؤل لابد ان نعرف كيف يتم تحديد حجم الصف فى الخوارزميات الثلاث، خوارزمية الصف القصير تحسب طول الصف من المعادله

$$B = \frac{\mathrm{RTT}*\mathrm{C}}{\sqrt{n}} \qquad (1)$$

حيث ان RTT هو زمن ارسال الرسالة وعودة الاقرار عليها و C هي سعة القناة، وn عدد الاتصالات التى تمر في القناة و B هو طول الصف المطلوب اما خوارزمية Codel تحسب طول الصف من المعادله

$$B = Nominal\ RTT * C \qquad (2)$$

حيث ان Nominal RTT هو قيمة الزمن الاسمى وهى قيمة محددة تساوى (100ms)، خوارزمية القاعدة القديمة تحسب طول الصف من المعادله

$$\mathrm{B} = \mathrm{RTT} * \mathrm{C} \qquad (3)$$

هذه القاعدة تشبه خوارزمية (Codel) ماعدا ان خوارزمية (codel) تستخدم بدلا من الزمن RTT القيمة الاسمية RTT، تستخدم الدراسه قيمة RTT والتى تقدر ب 250ms كقيمة ثابته لجعل الصف طويل بشكل كافي.

سوف نأخذ مثال رقمى لتوضيح اطوال الصفوف فى كل انواع الخوارزميات اعلاه، نأخذ القيم التالية RTT= 100ms، C=2 Mb/s وعدد الاتصالات هو 11 إتصال، وباعتبار ان طول الرساله هو 1500 بايت، يوضح الجدول رقم (1) نوع الخوارزميه ومعادله حساب طول الصف وطول الصف الناتج.

**الجدول رقم (1) يوضح اطوال الصفوف الناتجه عن انواع الخوارزميات الثلاث.**

| الرقم | الخوارزميه | المعادله المستخدمه | طول الصف |
|---|---|---|---|
| 1 | الصفوف القصيرة | $\mathrm{B} = \frac{100\ ms \times 2\ Mb/s}{\sqrt{11}}$ | 9 |
| 2 | خوازرمية Codel | B =100 ms × 2 Mb/s | 25 |
| 3 | القاعدة القديمة | B = RTT * C | 62 |

طول الصف في الخوارزميات 9، 25، 62 رسالة لكل من خوارزمية الصفوف القصيرة وخوارزمية (Codel) وخوارزميه القاعدة القديمة على التوالى اذا طول الصف فى خوارزمية (Codel) ذو قيمه كبيرة والاسواء من ذلك انها تتناسبا تناسب طردياً مع سعة القناة C وبالتالى لابد أن يكون الصف طويلاً لتعمل خوارزمية (Codel) بصورة جيدة ولا تتحول الى (Drop-Tail). تعتمد خوارزمية (Codel) على الاسقاط بكثافة من الرسائل الواردة وذلك لضمان مقدار تأخير قليل للرسائل التى تنجو من الاسقاط.

### 2.3 مشاكل خوارزمية التحكم بالتأخير (Codel)

اكبر مشكلة قد تواجه خوارزمية التحكم بالتأخير (Codel) هو سرعة خط الدخل (Access link) حيث ان لخوارزمية (Codel) ميزة أساسية وعن طريقها تحافظ على قيمة التأخير القليلة والتى صممت أساساً من أجلها. هذه الميزة هي القيمة الابتدائية التى تنتظرها خوارزمية Codel قبل ان تبدأ بالاحساس ان هناك تأخير على صف الموجه وهي عبارة عن قيمة زمنية لاتتجاوز 5ms وهذه القيمه تترجم الى عدد الرسائل الموجوده بالصف كالآتى، تقوم الخوارزمية بتسجيل زمن دخول الرسالة وفيما بعد تسجل زمن خروجها من الموجه ويسمى هذا الفرق بزمن الزيارة وهو الزمن الذى تمكثه الرساله فى الصف بالموجه.

اذا كان زمن الزيارة أكبر من الزمن 5ms (الزمن الهدف) فإن الخوارزمية تحس بأن هناك تأخير ثم تبدأ بالتصرف حيال ذلك، يتحول الزمن 5ms فى الصف الى عدد رسائل متناسب مع سرعة خط الدخل (Access link) الى الموجه فاذا كانت سرعة الخط 2Mb/s والتأخير على خط الدخل 20ms فإن سعة خط الدخل 3 رسائل (2Mb/s*20ms/1500*8) اي ان هناك ثلاثة رسائل تنتشر على امتداد خط الدخل، او بعباره اخرى هناك رسالة واحدة تدخل الى صف الموجه كل 7ms تقريباً (20ms/3)، واذا كان الاتصال وحيداً فإن ذلك لن يؤثر على عمل خوارزمية (Codel) باعتبار ان سعة الصف تساوي 20 رساله، فاذا تم زيادة عدد الاتصالات الى 10 اتصالات وهذه الاتصالات ترسل رسائلها بصورة متزامنة فإن صف الموجه سوف يستقبل 10 رسائل لكل 7ms وهذا ايضاً لن يؤثر على عمل خوارزمية (Codel) بمعنى ان ورود 10 رسائل (10 رسائل كل 7ms) فى آن واحد لن يؤدي الى ان يمتلئ صف خوارزمية (Codel) ويقوم بالاسقاط من آخر الصف ومن جهه اخرى فإن الخوارزمية سوف تكتشف ان هناك بوادر إزدحام وذلك بعد ان تكتشف ان زمن الزيارة أكبر من 5ms ولكن هذا لا يعني ان تقوم الخوارزمية بالاسقاط مباشرة فهي لن تقوم بالاسقاط الا بعد 100ms (ثابت في Codel) وان يكون زمن الزيارة مازال اكبر من 5ms.

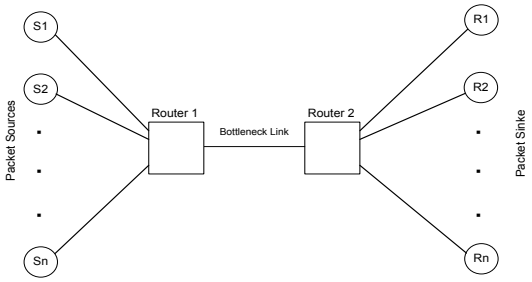
مثال آخر لسرعة خط الدخل فى هذه الحاله نفترض ان سرعة خط الدخل هي 300Mb/s، وذلك يعني ان هناك 500 رسالة تنتشر على خط الدخل كله (300Mb/s*20ms/1500*8)، ويعنى ايضاً ان هناك أكثر من رسالة واحدة تدخل الى صف الموجه (بالضبط 20ms/500=0.4) لكل 0.4ms كمثال اذا كان هناك 10 اتصالات متزامنة يؤدي ذلك الى دخول 4 رسائل كل 1ms وذلك لن يؤثر على عمل خوارزمية (Codel) ولكن اذا كان عدد الاتصالات اكبر من 10 اتصالات كمثال اذا كان عدد الاتصالات 30 اتصال فإن ذلك يعنى دخول 30*0.4ms= 12 رسالة كل 1ms و 60 رسالة كل 5ms واذا كان طول الصف الكلى لخوارزمية (Codel) 20 رسالة فإن الصف فى هذه الحالة سيمتلئ تماماً ويقوم باسقاط الرسائل من آخر الصف قبل ان تكتشف خوارزمية (Codel) ان هناك بادرة إزدحام، استخدمنا كلمة بادرة ازدحام لان خوارزمية (Codel) عندما تحس بالازدحام وذلك عند زيادة زمن الزيارة للرسالة عن 5ms لاتقوم بالاسقاط مباشرة وتعتبر ذلك إشارة او بادرة للازدحام وتقوم بالاسقاط بعد 100ms وذلك اذا استمر زمن الزيارة اكبر من 5ms.

### 4. الحساب الآلي لطول الصف auto

تعتمد خوازرمية الصفوف القصيرة على المعادلة (1) لتقدير طول الصف المناسب للعمل عليه، العامل الذى يصعب تقديره هو عدد الاتصالات n التى تتشارك فى القناة وبالتالى لابد من إيجاد طريقة ما يمكن بها تقدير عدد الاتصالات، سوف يتم تقدير

عدد الاتصالات بصورة احتمالية من الرسائل الوارده، هذه الفكرة مأخوذة من خوارزمية سابقة هي (SRED) [6] والتي تستخدمها لتحديد الاتصالات الاكثر استخداماً للقناة ومعاقبتها.

اختريت هذه الطريقه وهي تقدير عدد الاتصالات بصورة احتماليةً من الرسائل الواردة وذلك لان السبب الذي جعل خوارزمية (Codel) لا يمكن تطبيقها عمليا هو انها تطلب من الموجه ان يقوم بحساب زمن دخول وخروج اي رساله ليحسب منه فيما بعد زمن الزياره، وهي عمليه غير منطقيه وخاصه في الموجهات الكبيره والتي تتعامل مع عدد كبير من الرسائل، الشكل (2) يوضح تصميم الشبكه التي تم إجراء التجربه عليها، حيث تحتوي على موجهين متصلين عبر خط سعته 10Mbit/s وذو تأخير إنتشار يبلغ 10ms، الجهاز S1 يرسل الجهاز الى R1 الجهاز S2 يرسل الى الجهاز R2 وهكذا، يرسل المستقبلين رسائل إقرار ACK كما هو مطلوب من بروتوكول TCP [7].




**الشكل (2) يوضح شبكه تحتوي على موجهين وبينهما قناة عنق الزجاجه.**

## 1.4 طريقة عمل خوارزمية الحساب الآلي لطول لصف

تقوم خوارزمية الحساب الآلي لطول الصف بتقدير طول الصف المناسب للعمل عليه عن طريق المعادلة رقم (1) والمستعملة فى خوارزمية الصفوف القصيرة حيث ان خوارزمية الصفوف القصيرة هي الخوارزمية المناسبة للعمل فى عالم اليوم المتسم بالتأخير المتسبب فى مشكلة تضخم الصفوف (Bufferbloat) ولكن خوارزمية الصفوف القصيرة ينقصها عامل حاسم وهو تقدير حجم الصف المناسب بصورة آلية حسب المعادله $(B = \frac{RTT*C}{\sqrt{n}})$ وحيث ان طول الصف فى خوارزمية الصفوف القصيرة يتم تقديره من عدة معاملات هي سعة القناة (Bandwidth)، والمعامل RTT وعدد الاتصالات n حيث ان سعة القناة هو معامل ثابت و RTT هو معامل متغير ولكن يتم أخذ قيمة ثابته له فى هذه الحالة وهي100ms وذلك لمقارنة هذه الخوارزمية الجديدة مع خوارزمية Codel إما المعامل الذي يجب تقديره هو عدد الاتصالات n التى تمر حالياً فى القناة.

## 2.4 تقدير عدد الاتصالات التى تمر بالموجة

يمكن للموجة ان يعرف عدد الاتصالات التى تمر به ولكن ذلك يتطلب منه جهد وذاكرة كبيرين يتمثل الجهد الكبير الذي يبذله الموجه فى عدد دورات المعالج التى يحتاجها لتصنيف الرسائل، حيث يحتاج الموجه لتصنيف الرسائل بغرض معرفة عنوان المرسل وعنوان المستقبل وهذا عمل كبير سيكون من الصعب على الموجه القيام به بالاضافة الى عمله الاساسي وهو استقبال عدد كبير من الرسائل وتوجيهها ويحتاج الموجه ايضاً الى ذاكرة كبيرة لتصنيف الرسائل بهذه الطريقة.

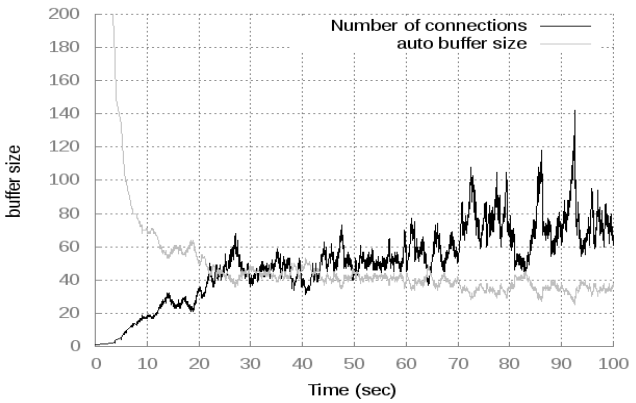
### 1.2.4 طريقة ميسره لمعرفة عدد الاتصالات

يمكن تقدير عدد الاتصالات احتمالياً من الرسائل التى تمر بالموجه حيث انه وفى دراسة سابقة طورت لمعرفة الاتصالات التى تستهلك القناة اكثر من غيرها تقوم فكرة هذه الخوارزمية على إنشاء قائمة تحتوى على الرسائل الموجودة فى الصف ويمكن ان تكون اكبر من ذلك لتحتوى على الرسائل الموجودة فى الصف (الاحياء) وبعض الرسائل التى تم ارسالها (الاموات) تسمى هذه القائمة باسم (Zombie list) قائمة الاموات الاحياء وذلك بسبب ان الرسائل التى تم ارسالها عبارة عن رسائل ميتة (اموات) ولكن يحتفظ بها للقيام ببعض الاعمال التى تقوم بها هذه الخوارزمية وهي اولا ملء قائمة الاموات الاحياء ثم بعد ذلك مقارنة كل رسالة واردة مع رسالة من القائمة بصورة عشوائية فاذا تطابقت هاتان الرسالتان يسمى هذا (Hit) واذا لم يحدث تطابق يسمى هذا (Miss) ويمكن فى هذه الحالة ان تحل الرسالة الجديدة محل الرسالة القديمة بصورة احتمالية ايضاً وبما ان أي إتصال يمر بالموجه يمكن ان يكون له رسالة واحدة او عدة رسائل فى الصف، فإنه يمكن تقدير عدد الاتصالات من حالات التطابق التى تحدث، لايجاد عدد الاتصالات من حالات التطابق نفترض ان لدينا P(t) عباره عن تقدير لحالات التطابق حول زمن وصول رساله محدده للصف من المعادلة (4) حيث ان Hit(t)=0 if no hit و Hit(t)=1 if hit و $\alpha$ قيمه ثابته $0 < \alpha < 1$، من المعادلة (5) يمكن معرفة عدد الاتصالات التى تمر بالموجه.

$$P(T) = (1-\alpha)p(t-1) + \alpha Hit(t) \quad (4)$$

$$Number\ of\ connections\ (n) = \frac{1}{P(t)} \quad (5)$$

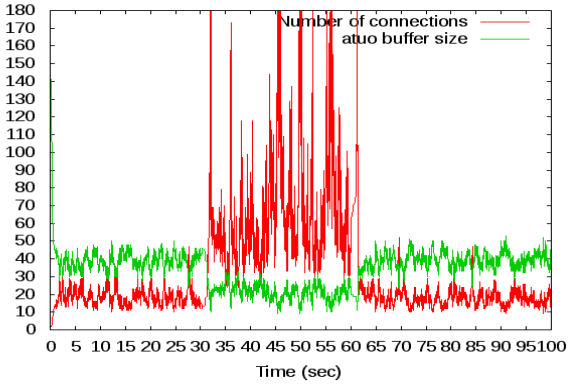
يمكن استعمال عدد الاتصالات من المعادلة (5) مباشرة فى المعادلة (1) والتى تستخدم فى تقدير طول الصف المناسب.




**الشكل (3) يوضح تحكم خوارزمية الحساب الالي في طول الصف حيث يقل طول الصف مع زيادة عدد الاتصالات حسب معادلة خوارزمية الصفوف القصيره.**

في الشكل (3) المحور الافقي يمثل الزمن والمحور الرأسي يمثل طول الصف المخصص من بداية التجربه للموجه وهو 200 رساله، تم اجراء التجربه عن طريق زيادة عدد الاتصالات تدريجيا ابتداءا من اتصال وحيد وحتى 100 اتصال،كما هو ظاهر من الشكل عندما كان هناك اتصال وحيد كان طول الصف 180 رساله وعند الزمن 10 ثانيه زاد عدد الاتصالات مما ادى الى ان يقل طول الصف الى 40 رساله، وعند الزمن 30 ثانيه وبعد زيادة عدد الاتصالات مره اخرى قل طول الصف الى 20 رساله فقط ثم استقر طول الصف عند قيمه تتراوح حول القيمه 10-20 رساله، الشكل (3) يوضح امكانية تقدير طول الصف المناسب في خوارزمية (auto) من عدد الاتصالات، حيث انه وفى البداية عندما يكون عدد الاتصالات قليلاً كان طول الصف كبيرا وذلك حسب القيمه الاساسيه لطول الصف والتي تم تخصيصها للموجه وعند زيادة عدد الاتصالات يقل طول الصف. في الشكل (4) تم تغيير عدد الاتصالات من 20 اتصال الى 70 اتصال ومره اخرى الى 20 لملاحظة اثر تغيير عدد الاتصالات على طول الصف، وجاءت النتائج مطابقه للتوقعات

حيث انه عندما كان عدد الاتصالات 20 اتصال كان الصف طويلا وذلك في اول 30 ثانيه من زمن التجربه وعندما تم تغيير عدد الاتصالات الى 70 اتصال قامت خوارزمية الحساب الالي بتعديل طول الصف اليا ليكون متناسبا مع عدد الاتصالات حيث يلاحظ من الشكل (4) وعند الزمن 31 زاد عدد الاتصالات وبالتالي قل طول الصف وذلك بصوره آليه وعند الزمن 61 ثانيه تم ارجاع عدد الاتصالات مره اخرى الى 20 اتصال حيث قامت خوارزمية الحساب الالي بسرعه بتعديل طول الصف حيث قامت في هذه المره بزيادة طول الصف ليكون متناسبا مع عدد الاتصالات القليل (20 اتصال) وذلك حسب خوارزمية الصفوف القصيره والتي تحسب من المعادله رقم (1).




**الشكل (4) يوضح تحكم خوارزمية الحساب الالي في طول الصف حيث يتناسب طول الصف مع عدد الاتصالات حسب معادلة خوارزمية الصفوف القصيره.**

بما أن شبكات الحاسب هي شبكات تبادل رسائل اذا لابد ان تحتوى هذه الشبكات على صفوف وذلك لتخزين الرسائل لحين ارسالها واذا لم يكن هناك صف كافي لتخزينها فيمكن ان يسقط عدد كبير منها، ولكن وجود صف كبير يؤدى ايضاً الى مشاكل ليست اقل سوءاً حيث يؤدى الى زيادة التأخير، اذاً تقدير طول الصف امر حيوي جداً ولابد منه وذلك لضمان عمل الشبكة بصورة جيدة، فى الخوارزمية الجديدة للحساب الآلي لطول الصف تم الاعتماد على المعادلة (1) من الدراسة [5] والتى تقول ان هذا الطول للصف هو الطول المناسب والذي يضمن تأخير قليل جداً وفى نفس الوقت لايؤدى الى حدوث ضياع من سعة قناة عنق الزجاجة والتى نحرص ان تكون مشغولة دائماً. حيث ان المعادلة (1) هي نفس معادلة الدراسة [4] ولكن يتم قسمة حاصل ضرب سعة القناة و RTT على الجزر التربيعي لعدد الاتصالات حيث يكون الناتج هو طول صف وذلك اقل بكثير من المعادلة (2) من الدراسة [4].

### 3.4 تقدير طول الصف المناسب من غير فقد في الاستفاده من سعة القناة (throughput)

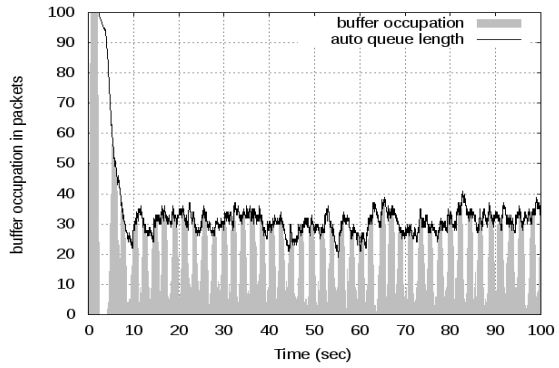
يعتبر العمل الاساسي لخوارزميه auto هو تقدير طول الصف المناسب للموجه بحيث يحافظ على سعة قناة عنق الزجاجه بصوره تامه، لمعرفة ان الخوارزميه auto تقوم بتقدير طول الصف المناسب للعمل عليه تم قياس سعة قناة عنق الزجاجه للتأكد من ان الخوارزميه لا تتسبب في اهدار سعة هذه القناة ويقاس ايضا التأخير وهو العامل الهام والذي صممت الخوارزميه من اجله، حيث ان هذه الخوارزميه صممت بالاساس للتخلص من التاخير الزائد في صفوف الموجهات او بكلمات اخرى التخلص من تضخم الصفوف (Bufferbloat)، في هذه الفقره تم اجراء تجربتين، في التجربه الاولى جعلت سعة قنوات الدخل مساوي لسعة قناة الخرج وفي التجربه الثانيه تم مضاعفة قنوات الدخل مره واحده، الغرض من هذه التجارب هو التأكد من ان الخوارزميه تحافظ على مقدار تأخير قليل بالاضافه الى المحافظه على سعة قناة عنق الزجاجه.

### 1.3.4 تقدير طول الصف بواسطة خوارزمية auto

طول الصف المقدر بواسطة خوارزمية auto هو الطول المحسوب بواسطة المعادله (1) حيث ان B هو الطول المناسب والذي يحافظ على سعة قناة عنق الزجاجه حسب الدراسه [5]، في الفقره السابقه تم تحليل قدرة خوارزمية على تقدير عدد الاتصالات الحيه n الماره عبر الموجه حيث ان للخوارزميه القدره العاليه على تقدير عدد الاتصالات بصوره دقيقه وجيده، في هذه الفقره تم تحليل قدرة الخوارزميه على تقدير طول الصف المناسب B والذي يحافظ على سعة القناة عنق الزجاجه، اجريت عدة تجارب عمليه على المحاكي ns-2 [8] لمعرفة قدرة الخوارزميه auto على تقدير طول الصف المناسب والذي يحافظ على سعة قناة عنق الزجاجه.

**التجربه الاولى**

في التجربه الاولى والتي تهدف لمعرفة طول الصف المناسب والذي يمكن خوارزمية auto من الاستفاده من سعة قناة عنق الزجاجه، عدد الاتصالات 100 اتصال TCP ذات خطوط اتصال access link حتى الموجه atuo بسعة 200kb\s، وبالتالي فان الدخل الكلي الى موجه auto يساوي مجموع عدد الاتصالات مضروبا في سعة الوصول access link والتي تنتج 20Mb\s.

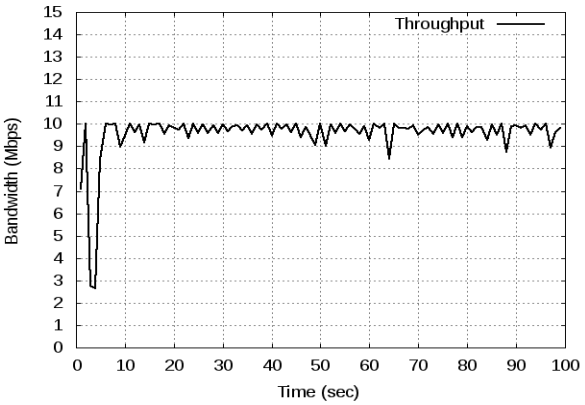



**الشكل (5) الخط المتعرج باللون الاسود يوضح طول الصف العامل المقدر بواسطة خوارزميه auto والمساحه في الاسفل باللون الرمادي توضح الحيز المملؤ من طول الصف.**

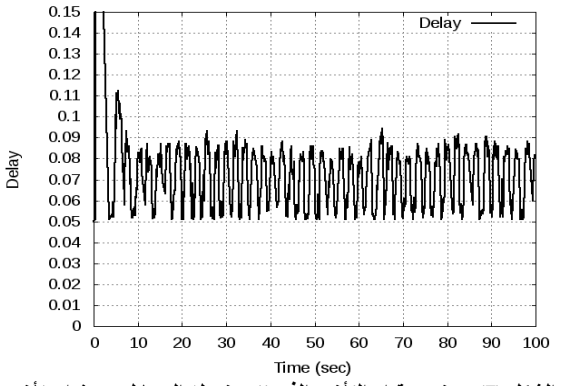
يلاحظ من الشكل (5) ان الخط الاسود الثخين والذي يمثل الطول الكلي لصف الموجه auto المقدر بواسطة الخوارزميه، والحيز المملؤ بواسطة اللون الرمادي يمثل الحيز المشغول من صف الموجه بواسطة مجموع عدد الاتصالات التي تقوم بالارسال الان، يلاحظ ان الحيز المملؤ بالرمادي لا يستطيع ان يتجاوز الخط الاسود الثخين وذلك لان هذا الخط هو الحد الاعلى المسموح باشغاله حاليا من صف الموجه (عمليا يحتوي الموجه على طول اكبر من ذلك وهو بالضبط 200 رساله) عندما يصل الخط الرمادي الى الخط الاسود فأن ذلك يعني ان هناك رسائل تصل الى صف الموجه ولا يستطيع الموجه تخزينها حسب خوارزمية auto ولان ذلك يعني زيادة التأخير وبالتالي فأن هذه الرسائل يتم اسقاطها فورا، هناك بعض الفراغات بين الخطوط الرماديه المرتفعه والتي تمثل حيز من صف الموجه فارغ من الرسائل، هذه الارتفاعات تعبر بصوره واضحه عن طريقة عمل برتوكول TCP في الارسال وهي طريقة الارسال الانفجاري[9-11].

من الشكل (6) يتضح مقدار الاستفاده من سعة قناة عنق الزجاجه والتي تتراوح مابين 10Mb\s و 9.5Mb\s تقل الاستفادة

بصوره ملحوظه في هذه التجربه وذلك بسبب ان هناك عدد كبير من الرسائل تم اسقاطها والتي سوف تؤدي الى تقليل معدل الارسال حيث ان اي اتصال TCP يتعرض لعملية اسقاط رساله يقوم بانقاص معدل ارساله في المره القادمه وبالتالي فأن معدل الاستفاده من سعة قناة عنق الزجاجه يقل بمقدار ضئيل.




**الشكل (6) يوضح الحيز المستفاد منه من سعة قناة عنق الزجاجه موضع الدراسه والبالغ 10mb/s يلاحظ ان الاستفاده تقل بصوره ملحوظه وذلك بسبب تزايد معدل الاسقاط.**

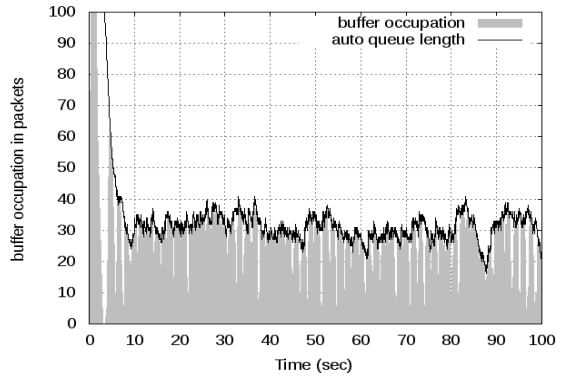



**الشكل (7) يوضح مقدار التأخير الذي تتعرض له الرسائل، حيث ان تأخير قناة عنق الزجاجه 50ms ويزيد التأخير الذي تتعرض له الرسائل داخل الصف بمقدار يتراوح مابين 1ms و 40ms حيث يصل التأخير الكلي الى 90ms.**

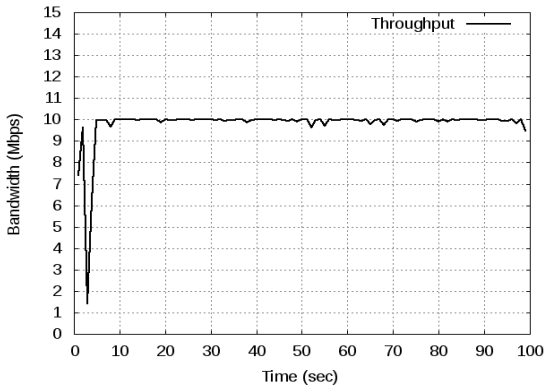
الشكل (7) يوضح مقدار التأخير الذي تتعرض له الرسائل والذي يتراوح مابين 50ms و 90ms اي ان التأخير الكلي مابين 1ms وحتى 40ms حيث ان تأخير عنق الزجاجه 50ms، اقصى يسار الشكل (7) يوضح ان هناك قيمه كبيره للتأخير والتي تصل الى 200ms تنتج هذه القيمه الكبيره للتأخير وذلك لان خوارزمية auto في البدايه تحتاج الى زمن قليل لتقدير طول الصف المناسب وبالتالي فهي تعمل بالصف المعطى من قبل الموجه وهو 200 رساله ولكن بعد زمن قليل جدا تتمكن من تقدير عدد الاتصالات الحيه وبالتالي تقوم بانقاص طول الصف العامل الفعلي وبالتالي يقل التأخير الذي تتعرض له الرسائل نتيجه لوقوفها في الصف بانتظار الارسال، اي ان صف الموجه يسمح بتاخير اعظمي للرسائل على الصف يصل هذه التاخير حتى القيمه 40ms ولكن لن يسمح بتجاوز هذه القيمه من التأخير مطلقا.

**التجربه الثانيه**

في التجربه الثانيه والتي ايضا تهدف لمعرفة طول الصف المناسب والذي يحافظ على قدرة خوارزمية auto على الاستفاده من سعة قناة عنق الزجاجه، عدد الاتصالات 100 اتصال TCP ذات خطوط اتصال access link حتى الموجه auto بسعة 300kb\s وبالتالي فان الدخل الكلي الى الموجه auto يساوي مجموع عدد الاتصالات والذي يساوي 100 اتصال مضروبا في سعة قناة الوصول access link والتي تنتج القيمه 30Mb\s من الشكل (8) الخط العلوي باللون الاسود يمثل طول الصف B المقدر بواسطة خوارزمية auto من المعادله (1) والجزء السفلي باللون الرمادي يمثل الحيز الممتلئ من صف الموجه auto، هناك طبعا بعض الفراغات في الجزء الممتلئ باللون الرمادي والتي تدل على ان الصف لم يمتلئ بصوره كامله عندما يرتفع الخط الرمادي حتى يلامس الخط الاسود فان ذلك يعني ان عدد الرسائل التي تدخل الى الصف اكبر من طول الصف الفعلي وبالتالي فإن الرسائل في هذه الحاله يتم اسقاطها، يتعرج الخط باللون الاسود بالرغم من ان عدد الاتصالات 100 اتصال بصوره ثابته ولكن الخوارزميه تقوم بتقدير عدد الاتصالات من الرسائل الواصله فعندما يتم اسقاط بعض الرسائل من الصف فان ذلك يؤدي الى ان تقوم الاتصالات بتقليل ارسالها في المره القادمه حسب توجيهات بروتوكول TCP وبالتالي فان عدد الرسائل التي تصل في المره التاليه تؤدي الى تقليل الصف نسبة لتقليل عدد الرسائل الواصله والتي منها يتم تقدير عدد الاتصالات وبالتالي تقليل طول الصف، تظهر هذه النقطه جليه في المنطقه التي تسبق الزمن 90s من الشكل (8) يتضح ان هناك انهيار في طول الصف المقدر وذلك بسبب الاسقاط المكثف في المنطقه التي تسبق منطقه انهيار طول الصف.



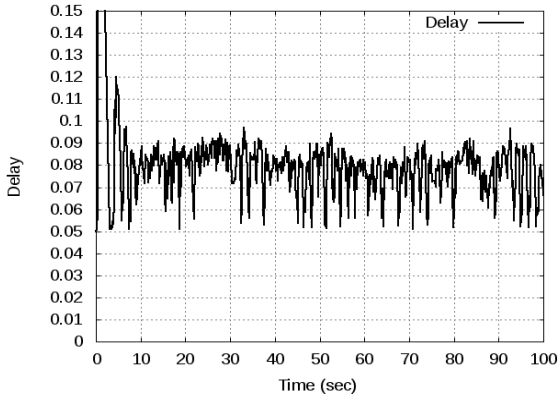

**الشكل (8) الخط المتعرج باللون الاسود يوضح طول الصف العامل المقدر بواسطة خوارزميه auto والمساحه في الاسفل باللون الرمادي توضح الحيز المملؤ من طول الصف.**




**الشكل (9) يوضح الحيز المستفاد منه من سعة قناة عنق الزجاجه موضع الدراسه والبالغ 10mb/s يلاحظ ان الاستفاده تقل بصوره ملحوظه وذلك بسبب تزايد معدل الاسقاط.**

الشكل (9) يوضح انه يتم المحافظه على سعة قناة عنق الزجاجه بصوره عامه فيما عدا ان المنطقه الزمنيه الابتدائيه والتي لم تقدر الخوارزميه بعد طول الصف المناسب يحدث انهيار تام لسعة القناة وذلك ايضا بسبب ان الجزء الاول من خوارزمية

التحكم بالازدحام في بروتوكول TCP وهي خوارزمية البدايه البطيئه Slow Start [12] والتي يحدث فيها الكثير من الاسقاط قبل ان تستقر اتصالات TCP بعد ذلك في حيز ارسال مناسب مع سعة القناة المتاحه، اما في الشكل (10) والذي يوضح التأخير الذي تتعرض له الرسائل فيلاحظ ان التأخير تناقصت قيمته قليلا ما بين 50ms و 80ms والذي يبلغ 30ms، اي ان التأخير الذي تتعرض له الرسائل داخل صف الخوارزميه auto يتراوح مابين 1ms وحتى 30ms ولكن بعد هذه التاخير لن يسمح لاي رساله بالمكوث في صف الموجه وسيتم اسقاطها فورا.




**الشكل (10) يوضح مقدار التأخير الذي تتعرض له الرسائل، حيث ان تأخير قناة عنق الزجاجه 50ms ويزيد التأخير الذي تتعرض له الرسائل داخل الصف بمقدار يتراوح مابين 1ms و 40ms حيث يصل التأخير الكلي الى 90ms.**

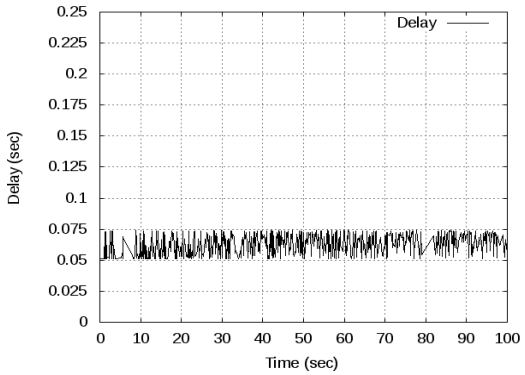
## 5. المحاكاة

في هذه الفقره تم مقارنة الخوارزميات التي تعمل في مجال التحكم بالازدحام وبالتالي التحكم بالتأخير على الصف وهما خوارزمية Drop-tail [4] وخوارزمية RED [13]، بالاضافه الى الخوارزميات التي تعمل في مجال التحكم بالتأخير فقط وهما خوارزمية Codel وخوارزمية الحساب الآلي لطول الصف auto، قمنا باجراء تجربتين في الاولى ثبتنا طول الصف عند 20 رساله لمعرفة التأخير الذي تتعرض له الرسائل عند هذا الطول وفي التجربه الثانيه جعلنا طول الصف 100 رساله وذلك لمعرفة من اين ينتج التأخير الكبير في الشبكه الموضحه في الشكل (2).

### 1.1.5 خوارزمية Drop-Tail

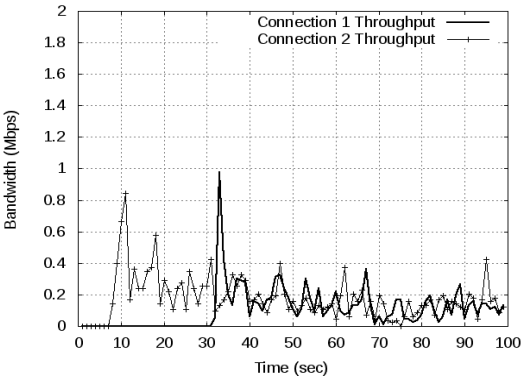
**التأخير:** تقوم خوارزمية Drop-Tail بإسقاط الرسائل من آخر الصف وهي الرسائل التى لم تجد لها مكاناً فى الصف، وعندما يكون الصف قصيراً تقوم الخوارزمية إلزامياً باسقاط كل الرسائل الواردة والتى لم تجد لها مكاناً وبالتالى عدد الرسائل التى تستطيع هذه الخوارزمية تخزينها فى الصف سيكون قليلاً وذلك لأن طول الصف قصير، وبما أن الاسقاط فى هذه الخوارزمية مربوط بطول الصف وبما ان طول الصف قصير بالتالى فان التأخير يكون قليلاً ومتناسب مع طول الصف، كما هو موضح في الشكل (11) فإن التأخير في منطقة عنق الزجاجه والذي لا يتعدى 10ms قد يصل الى 30ms وذلك عند استخدام خوارزمية Drop-Tail بالرغم من ان حجم الصف لا يتعدى 20 رساله فقط.

**العداله:** لدينا في هذه التجربه اكثر من مائة اتصال تتشارك في قناة عنق الزجاجه وبالتالي فان ضمان اقل حد من العداله مطلوب بين هذا العدد الكبير من الاتصالات، الشكل (12) يوضح الحيز الذي تحصل عليه كل اتصال من اتصالين تم اختيارهما بصوره عشوائيه من مجموعة الاتصالات المائه.




**الشكل (11): التأخير في منطقة عنق الزجاجه الذي يتراوح بين 1ms-25ms حيث تم تثبيت حجم الصف عند 20 رساله.**

من الشكل يتضح ان الاتصال ذو الخط بعلامة الجمع استطاع ان يتحصل على حيز اكبر لانه استطاع ان يبداء الارسال بصوره مبكره اما الاتصال الاخر فانه لم يستطيع ان ينشئ اتصاله بسرعه لانه تعرض لحالات اسقاط متكرر عند محولته انشاء الاتصال لذلك يظهر في الشكل انه لم يبداء الارسال الا عند الثانيه 30 حيث يظهر جليا عند هذه اللحظه انشاء الاتصال وهو في مرحلة البدايه البطيئه والدليل على ذلك الارتفاع الشديد لهذا الاتصال ثم الانخفاض والاستقرار عند منطقه محدده، من العيوب الوضحه مما سبق ان خوارزمية Drop-Tail لا تستطيع ان تضمن اقل حد للعداله لكنها تتفوق لى الخوارزميات الاخرى في مجال المحافظه على سعة القناة.

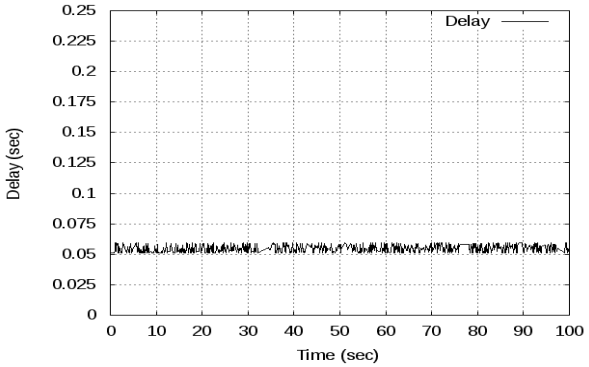



**الشكل (12): العداله بين الاتصالات التي تمر عبر موجه Drop-Tail، هناك اتصالين من مجموعة اتصالات ماره بالموجه يتحصل احدهما على حيز من اكثر من الاخر.**

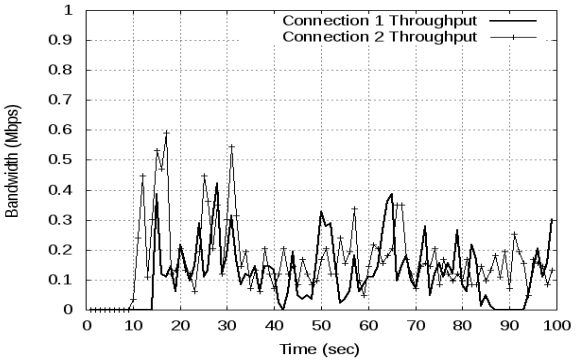
### 2.1.5 خوارزمية RED

**التأخير:** أما خوارزمية RED فهي ايضاً خوارزمية تعتمد على ضبط طول الصف يدوياً واحيانا في بعض الخوارزميات المطوره عن RED يتم ضبطه آليا [14-16]، حيث ان هذه الخوارزمية لديها معاملان مهمان وهما minth وهي العتبه الدنيا وmaxth وهي العتبه العليا، تقوم خوارزمية RED بتقسيم طول الصف الى ثلاثة مقاطع، المقطع الاول يبتدئ من واحد وينتهى عند minth فى هذا المقطع لا تقوم الخوارزمية RED باسقاط اي رسالة بل تسمح لكل الرسائل بالدخول والمقطع الثاني يبدأ من minth وينتهي عند maxth فى هذا المقطع تقوم الخوارزمية باسقاط الرسالة الواردة باحتمال يتناسب مع عدد الرسائل التى تأتي من هذا الاتصال المحدد، والمقطع الثالث يبدأ من maxth وينتهي عند نهاية حدود الصف وتقوم الخوارزمية باسقاط اي رسالة ترد في هذا المقطع. مما سبق يتبين ان خوارزمية RED ايضاً تعتمد على حدود ثابته يتم وضعها للصف فاذا وضعت حدود كبيرة للمعاملات اعلاه ولم يكن هناك

الا عدد قليل من الاتصالات باعتبار ان قناة عنق الزجاجة ذات سعة اقل فان الصف يكون كبيراً نسبة لكبر حجم معاملاته (minth،maxth ). وتراكم الرسائل فى النهاية يؤدى الى زيادة التأخير على الصف وبالتالى فإن التأخير على صف خوارزمية RED مشابه للتأخير على صف خوارزمية Drop-Tail وذلك عندما يكون طول الصف فى خوارزمية Drop-tail مساوي لطول المعامل maxth فى الصف المعتمد على خوارزميةً RED، فى النهاية فان خوارزمية RED تنقصها ميزة تقدير طول الصف بصورة آلية وذلك لتقليل التأخير، كما يتضح من الشكل (13) يتراوح التأخير ما بين 10ms - 22ms، حيث تتعرض الكثير من الرسائل لتأخير يصل حتى 22ms داخل الصف، اي تنتظر الرسائل دورها في الارسال لمدة 12ms داخل الصف حتي يتم ارسالها.




**الشكل (13): التأخير الذي تتعرض له الرسائل داخل صف خوارزمية RED حيث يتراوح ما بين 10ms وحتى 22ms وذلك عندما يكون طول الصف 20 رساله.**

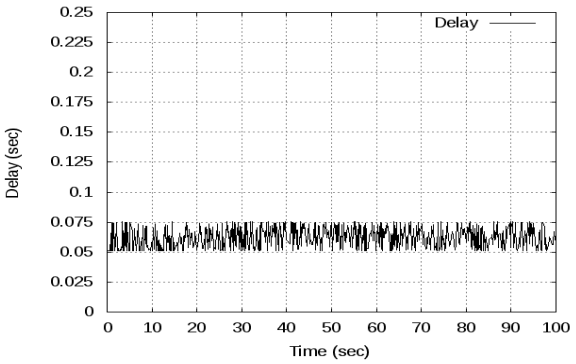



**الشكل (14): الحيز الذي يتحصل عليه اتصالين من مجموعة الاتصالات التي تتشارك في القناة حيث انه يساوى مقدار ما يتحصل عليه كل اتصال من سعة القناة بصوره تقريبيه.**

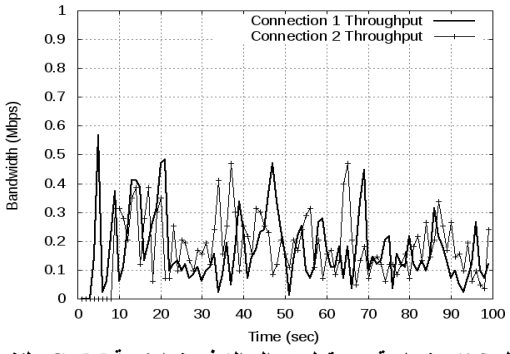
**العداله:** في هذه الفقره تم استكشاف قدرة خوارزمية RED على ضمان الحد الادنى من العداله للاتصالات التي تمر عبر الموجه، الشكل (14) يوضح الحيز الذي يتحصل عليه اتصالين من مجموعة الاتصالات التي تتشارك في القناة حيث انه يتساوى مقدار ما يتحصل عليه كل اتصال من سعة القناة بصوره تقريبيه من الشكل، استطاعت خولرزمية RED ان تضمن العداله بصوره جيده ولكنها لم تستطع ان تضمن العداله من بداية التجربه حيث بداء الاتصال الاول بعد 10 ثواني والاتصال الثاني بداء بعد 15 ثانيه بالرغم من ان كل الاتصالات بدأت في زمن واحد، تفسير ذلك يرجع الى ان هذين الاتصالين لم يستطيعا ان يقيما الاتصال وذلك بسبب الاسقاط المتكرر لرسائل اقامة الاتصال بسبب التزاحم الشديد الناتج عن قصر الصف وذلك بالرغم من ان من اهداف خوارزمية RED الاساسيه العداله.

### 3.1.5 خوارزمية (Codel)

**التأخير:** اما خوارزمية (Codel) فعند تزويدها بصف قصير تقوم بضمان تأخير قليل لكل الرسائل، تقوم خوارزمية (Codel) بالاسقاط العشوائي للرسائل وذلك بغض النظر عن حجم الصف المتوفر حيث تحسب زمن دخول اي رسالة وزمن خروجها وتحسب منه زمن الزيارة وتستنتج من هذا الزمن اذا كان هناك ازدحام ام لا فإن كان هناك ازدحام او بوادر ازدحام تقوم الخوازرمية بالاسقاط العشوائي من الصف وبالتالى تضمن أن حجم الصف يكون بصورة دائمه قصيراً، يوضح الشكل (15) التأخير الذي تتعرض له الرسائل في صف خوارزمية (Codel).




**الشكل (15): التأخير الذي تتعرض له الرسائل داخل صف خوارزمية Codel حيث يتراوح ما بين 1ms وحتى 25ms وذلك عندما يكون طول الصف 20 رساله .**



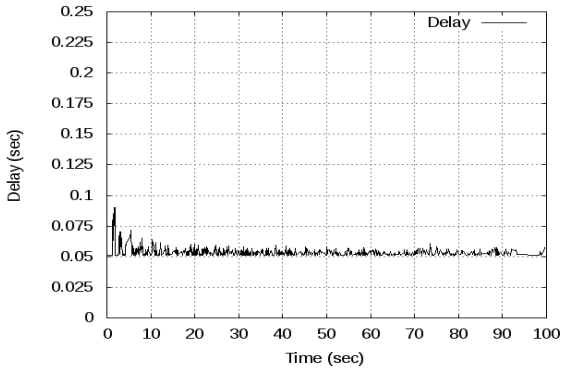

**الشكل (16): ضمان قدر معقول من العداله في خوارزمية Codel ولكنها لم تستطع ان توفر الحيز الكافي لكل اتصال ليبدأ الاتصال من البدايه حيث انه في الشكل لم يتمكن الاتصالان من العمل الثانيه الاولى في التجربه.**

**العداله :** من الشكل (16) يلاحظ ان اداء خوارزمية Codel فيما يتعلق بالعداله بين الاتصالات ذات اداء جيد ولكن لدينا نفس المشلكه السابقه وهي عدم قدرة خوارزميه Codel على توفير الحيز الكافي لاي اتصال ليبدأ عمله بطريقه مستقره، حيث لم يبدأ الاتصال الأول الارسال الا عند الثانيه الثالثه اما الاتصال الثاني فلم يبدأ الا عند الثانيه السابعه وذلك كما اوردنا بسبب التزاحم الشديد من قبل الاتصالات والذي يسبب اسقاط رسائل انشاء الاتصال، ينتج هذا الامر لان الاتصالات الاولى التي تنجح في انشاء الاتصال تدخل في مرحلة الارسال الكثيف بواسطة خوارزمية البدايه البطيئه (Slow Start) يؤدي هذا الارسال الكثيف الى امتلاء الصف بواسطة الرسائل المرسله من هذه الاتصالات المحظوظه وعندما ترد رسائل من الاتصالات الاخرى تجد ان الصف ممتلئ مما يؤدي الى سقوطها.

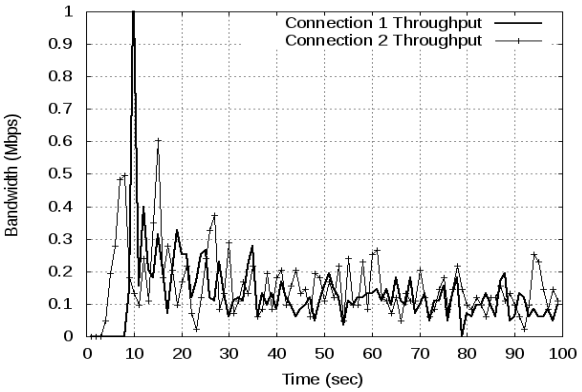
### 4.1.5 خوارزمية الحساب الآلي للصف (auto)

**التأخير:** اما الخوارزميه الاخيره فهي خوارزمية الحساب الآلي للصف (auto) فعند تزويدها بصف قصير تقوم بضمان تأخير قليل لكل الرسائل، تقوم خوارزمية (auto) اولا بتقدير طول

الصف المناسب لذلك يزيد التأخير بصوره كبيره (تمتد فترة الزياده الكبيره في الزمن مابين الثانيه 0 والثانيه 20) ولكن لفتره قصيره ثم يبدأ في التناقص مع زيادة عدد الاتصالات الماره، حيث ان التأخير المحسوب في الثواني الاولى من زمن التجربه هو تأخير لاتصال واحد ثم يبدأ هذا التأخير في التناقص مع دخول الاتصال الثاني ويستمر في التناقص حتى يستقر التأخير عند حد معين ثابت يتناسب مع عدد الاتصالات الماره وطول الصف، يتضح من الشكل (17) ان خوارزمية الحساب الآلي تحافظ على تأخير اقل بكثير من التأخير المفروض على الرسائل في خوارزمية (Codel).




**الشكل (17): التأخير الذي تتعرض له الرسائل داخل صف خوارزمية الحساب الآلي (auto) حيث يتراوح ما بين 1ms وحتى 30ms ولكن القيمه السائده للتأخير تتراوح مابين 1ms- 10ms وذلك عندما كان طول الصف 20 رساله فقط.**




**الشكل (18): ان خوارزمية auto تضمن قدر معقول من العداله بين الاتصالات الماره بالموجه.**

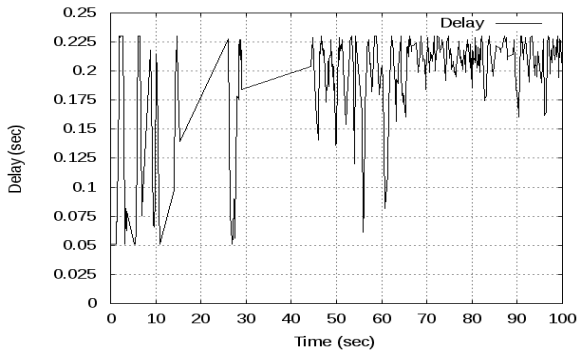
**العداله:** استطاعت خوارزمية auto ان تضمن قدر جيد من العداله بين الاتصالات الماره وذلك لان الخوارزميه تقدر طول الصف المناسب للعمل عليه، بالرغم من ان طول الصف اقصر من اللازم الا ان الخوارزميه استطاعت ان تعمل بصوره جيده لضمان العداله، ولكن الخوارزميه تواجه نفس المشاكل السابقه وهي عدم قدرة الخوارزميه على ضمان الحيز الكافي من الصف لكل اتصال حيث لم تستطع ان تبدأ الاتصالات الموضحه في الشكل (18) الارسال الا بعد فتره زمنيه ليست بالقصيره، خورازمية RED وخوارزمية Codel بالاضافه الى خوارزمية auto تقوم بالاسقاط العشوائي الذي ساعد في ضمان العداله ولكن خوارزمية auto تساهم زيادة على ذلك بالاسقاط المتناسب مع طول الصف المضاف للموجه.

## 2.5 الصف الطويل

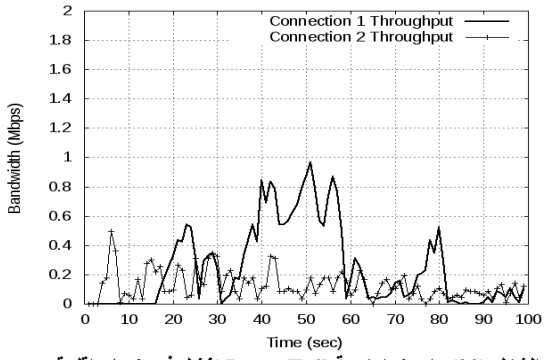
عند إعاد إجراء نفس التجربة السابقة للخوارزميات الاربعة : خوارزمية الاسقاط من آخر الصف وخوارزمية الاسقاط المبكر العشوائي وخوارزمية التحكم بالتأخير وخوارزميه الحساب الالي لطول الصف ولكن بعد زيادة حجم الصف من 20 الى 100 رسالة مع تثبيت كل المعاملات الاخرى كان أداء الخوارزميات كالآتى:

### 1.2.5 خوارزمية Drop-Tail

**التأخير:** تتميز هذه الخوارزمية بأنها لن تسقط اي رسالة من الصف الا اذا إمتلأ الصف عن آخره وبالتالى وبما ان قناة عنق الزجاجة التى تتحكم بها الخوارزمية مزدحمة دائماً فإن صفاً سوف يتكون ويزداد طوله تدريجياً مع عدم قدرة القناة على تمرير كل الرسائل وفى النهاية سوف يمتلئ الصف ويبدأ فى اسقاط الرسائل التى ترد اليه، الشكل (19) يوضح التأخير المحسوب هنا وهو التأخير الذى تمكثه الرسالة داخل الصف (Drop-Tail).




**الشكل (19): التأخير الذي تتعرض له الرسائل داخل صف خوارزمية Drop-Tail حيث يتراوح ما بين 10ms وحتى 130ms وذلك عندما يكون طول الصف 100 رساله.**




**الشكل (20): ان خوارزمية Drop-Tail تفشل في ضمان اقل قدر من العداله بين الاتصالات بسبب طول الصف.**

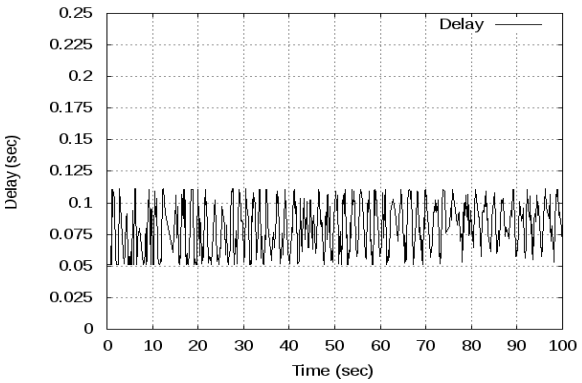
يلاحظ ان التأخير يزيد عن التأخير المحسوب فى التجربة السابقة عندما كان طول الصف 20 في الشكل (11) رسالة وذلك لأن الرسائل فى هذه التجربة تمكث فترة أطول فى الصف وذلك لأن الصف نفسه طويل، في خوارزمية الاسقاط من آخر الصف يزداد التأخير الذى تفرضه على الرسائل مع زيادة طول الصف ولا تملك هذه الخوارزمية اي وسيلة للتحكم بطول الصف المناسب للعمل والذى يوفر معدل تأخير قليل، وفي نفس الوقت تقليل طول الصف بصورة يدويه غير مدروسة قد يؤدي الى ان لا يستفاد من سعة قناة عنق الزجاجة (Underutilization) حيث يستفاد من الرسائل الموجودة فى الصف فى قناة عنق الزجاجة (Bottleneck) عندما تتراجع الاتصالات عن الارسال

**العداله:** تفشل خوارزمية Drop-Tail تماما عندما يتعلق الامر بضمان العداله بين الاتصالات الماره بالموجه، في الشكل (20) يتضح ان كل اتصال يتحصل على قدر مختلف عن القدر الذي يتحصل عليه الاتصال الاخر، في هذه التجربه والتي تم فيها زيادة طول الصف الى 100 رساله استطاع الاتصال بالخط

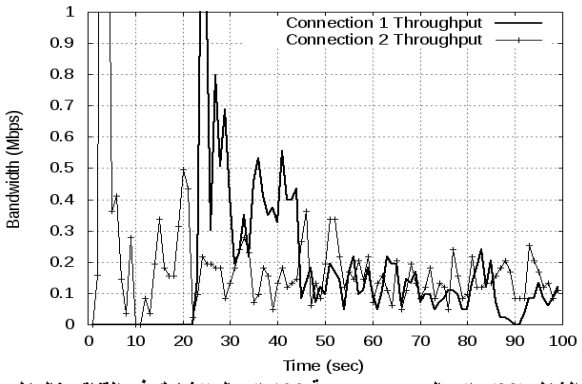
الثخين ان يقوم بتخزين عدد كبير من الرسائل في صف الموجه اكثر من الاتصال الاخر ولذلك يتحصل هذا الاتصال على حيز اكبر من سعة قناة عنق الزجاجه.

### 2.2.5 خوارزمية RED

**التأخير:** اما خوارزمية RED فهي لا تختلف كثيراً عن خوارزمية الاسقاط العشوائي سوى أن لها معاملات تتحكم فى عدم الاسقاط، معامل للتحكم فى الاسقاط العشوائي او الاسقاط الالزامي هو العتبه العليا (maxth) فعند زيادة طول الصف فى هذه التجربة الى 100 رسالة وزيادة حدود المعاملات ( minth ، maxth) لوحظ ان قيمة التأخير الذى تفرضه هذه الخوارزمية على الرسائل المارة يزداد مع زيادة طول الصف والمعاملات.




**الشكل (21): التأخير الذي تتعرض له الرسائل داخل صف خوارزميةRED حيث يتراوح ما بين 10ms وحتى 55ms وذلك عندما يكون طول الصف 100 رساله.**



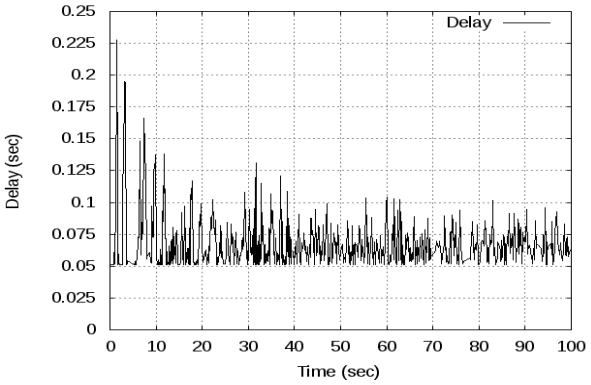

**الشكل (22): اتصالين من مجموعة 100 اتصال تتشارك في القناة، ينال كل اتصال حيز مختلف عن الاخر بسبب طول الصف.**

بصورة أوضح يتناسب التأخير مع المعاملات (minth) و (maxth) والتى تتحكم فى الاسقاط والتى يتم ضبطها يدوياً للاستفادة من سعة قناة عنق الزجاجة، يتم ضبط هذه المعاملات الى قيم كبيره حسب توصيات المصنعين لتقليل اسقاط الرسائل، هذه الخورازمية تم تطويرها أصلاً للتخلص من مشكلة التزامن وتوفير بعض العدالة ولم تصمم لتقليل التأخير، يتضح من الشكل (21) التأخير الذي تتعرض له الرسائل في صف خوارزمية (RED) ، حيث ان التأخير اقل بكثير من التأخير على خوارزميتة الاسقاط من آخر الصف وذلك لأن هذه الخوارزميه تعتمد الاسقاط المكثف من الاتصالات الماره.

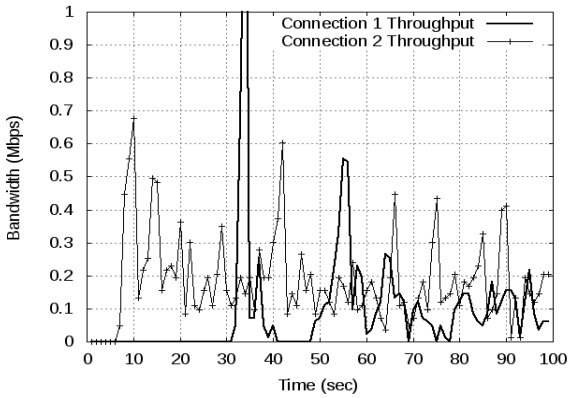
**العداله:** عند زيادة طول الصف الى 100 رساله يتعرض موضوع العداله الى الكثير من الاذى وذلك بسبب ان كل اتصال يستطيع ان يخزن عدد مختلف من الرسائل في ذلك الصف الطويل، من الشكل (22) يتضح ان تخزين عدد متفاوت من الرسائل في الصف ينعكس سلبا على مقدار الحيز الذي يناله كل اتصال من الصف وبالتالي على مقدار التأخير الذي تتعرض له الرسائل.

### 3.2.5 خوارزمية (Codel)

**التأخير:** أما فى خوارزمية (Codel) وعند زيادة طول الصف الى 100 رسالة لوحظ ان التأخير يظل كما هو وذلك بالمقارنة مع التجربة السابقة والتى كان فيها طول الصف 20 رساله يتضح ذلك من الشكل (23) حيث يكون التأخير ثابت عند كلا الطولين فى التجربتين وبالتالى فإن خوارزمية (Codel) تتحكم فى طول الصف وذلك بعزل طول الصف عن التأثير فى التأخير.




**الشكل (23): التأخير الذي تتعرض له الرسائل داخل صف خوارزميةCodel حيث يتراوح ما بين 10ms وحتى 50ms وذلك عندما يكون طول الصف 100 رساله.**



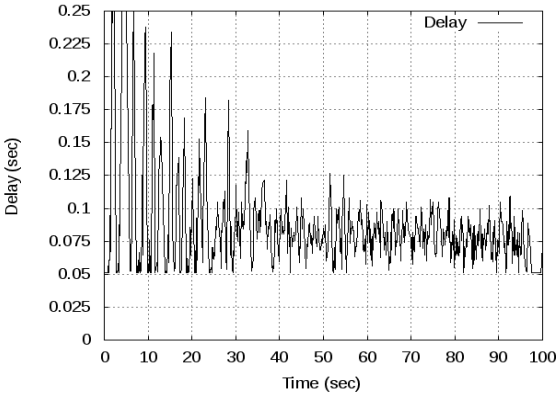

**الشكل (24) التحسن الطفيف في العداله بين الاتصالين تحت الدراسه، ويوضح الشكل ايضا عدم قدرة الاتصالين على الابتداء بسهوله.**

**العداله:** تتفوق خوارزمية Codel على خوارزمية RED في موضوع العداله بين الاتصالات الماره في القناة من الشكل (24) يتضح ان هناك تحسن طفيف يؤدي الى تناسب الى درجه مقبوله من العداله بين الاتصالين الموضحين في الشكل، ولكن هناك تأخر ملحوظ للاتصالين في انشاء الاتصال حيث يبدأ الاتصال الاول عند الثانيه السابعه اما الاتصال الثاني فيبدأ عند الثانيه الواحده والثلاثون وذلك طبعا بسبب الاسقاط الكثيف الذي يسقط رسائل انشاء الاتصال بالنسبه للاتصالين تحت الدراسه.

### 4.2.5 خوارزمية الحساب الآلي للصف (auto)

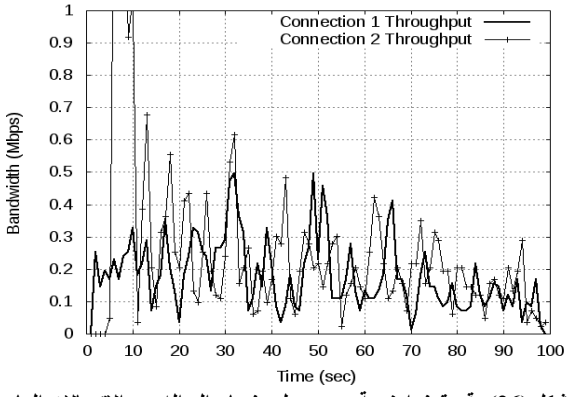
**التأخير:** في خوارزمية الحساب الآلي لطول الصف (auto) وعند زيادة طول الصف الى 100 رساله فإن التأخير يظل ثابتا كما لو ان طول الصف لم يتغير وذلك عند مقارنة الشكل (17) مع الشكل (25) حيث ان التأخير في الشكل (17) نتج عن طول صف يساوي 20 رساله والتأخير في الشكل (25) نتج عن طول صف يساوي 100 رساله، حيث يمكن تجاهل الثواني الخمسه

الاولى من نتائج المحاكاة والتي يكون فيها التأخير كبيرا، ينتج هذا التأخير الكبير لأن الخوارزميه تقوم بتقدير طول الصف استنادا على عدد الاتصالات الماره، وبالتالي فأنه وعندما يكون عدد الاتصالات قليلا وطول الصف كبير فان التأخير يكون كبيرا وبما انه لم يتم تقدير طول الصف بصوره جيده بعد، يكون التأخير كبير كما هو الحال في بداية هذه التجربه حيث تبدأ المحاكاة باتصال واحد ويزداد عدد الاتصالات بعد ذلك بصوره متتاليه، يتضح من الشكل (25) ان خوارزمية الحساب الآلي تحافظ على تأخير مشابه للتأخير المفروض على الرسائل في خوارزمية (Codel).




**الشكل (25): التأخير الذي تتعرض له الرسائل داخل صف خوارزمية الحساب الالي auto حيث يتراوح ما بين 10ms وحتى 40ms وذلك عندما يكون طول الصف 100 رساله.**

**العداله:** تتفوق خوارزمية auto على خوارزمية Codel في عدة امور اولها ان الاتصالين يبدآن الارسال في ازمان متقاربه وذلك من بداية التجربه، قدرة الاتصالين على الارسال ترجع الى قدرتهما على انشاء الاتصال وهو الامر المفقود في الخوارزميات السابقه والتي لم تستطع ان تضمن حيز كافي لكل اتصال لينشئ الاتصال الخاص به ويستمر في الارسال فيما بعد، الامر الثاني تضمن الخوارزميه قدر كبير من العداله بين الاتصالات وذلك واضح من الشكل (26) حيث يتساوى الاتصالين في القدر الذي يناله كل اتصال من سعة القناة وذلك طبعا بصوره تقريبيه من الشكل.




**الشكل (26): قدرة خوارزمية auto على ضمان العداله بين الاتصالات الماره بالموجه.**

### 3.5 الاستفادة من سعة القناة

تضمن خوارزمية الحساب الآلى للصف auto معدل استفادة كبير جداً لسعة قناة عنق الزجاجه، تم مقارنة خوارزمية الحساب الآلى مع عدة خوارزميات اخرى هى خوارزمية الاسقاط من آخر الصف (Drop-Tail) و خوارزمية الاسقاط العشوائي المبكر (RED) وخوارزمية التحكم بالتأخير (Codel) فى الشبكة المحددة فى الشكل (2) والتى تحتوى على قناة عنق الزجاجة ذات السعة (10mb/s)، الجدول (2) يوضح حجم الاستفادة من قناة عنق الزجاجة فى كل خوارزمية من الخوارزميات الاربعه اعلاه وذلك عندما يكون طول الصف 20 رسالة وهو طول قصير وعدد الاتصالات فى السيناريو حيث يوضح الجدول ان الاستفاده الاكبر تكون لخوارزمية الاسقاط من آخر الصف ثم لخوارزمية الاسقاط العشوائي ومن بعدها لخوارزمية الحساب الآلى وأخيراً خوارزمية Codel. ومع معدل تأخير قليل لكل الخوارزميات وذلك بسبب قصر طول الصف وعند زيادة طول الصف الى 100 رسالة كان ترتيب الاستفادة من سعة القناة نفس الترتيب السابق الجدول (2) مع زيادة كبيرة جدا فى التأخير لخوارزميتي الاسقاط من آخر الصف وخوارزمية الاسقاط العشوائي مما يتسبب في خروجهما من اللعبه، خوارزمية الحساب الآلى وخوارزمية (Codel) هما الخوارزميتين الفائزتين بمعدل تأخير اقل يضمن عمل تطبيقات الزمن الحقيقي بالاضافة الى ان خوارزمية الحساب الآلى تفوق خوارزمية (Codel) فى معدل الاستفادة من سعة قناة عنق الزجاجة.

**الجدول رقم (2): الاستفاده من سعة قناة عنق الزجاجه لكل واحدة من الخوارزميات تحت الدراسه وذلك عند طول صف 20 رساله ومره اخرى عند طول صف 100 رساله.**

| نوع الخوارزميه | الاستفاده من سعة القناة عند طول الصف 20 رساله | الاستفاده من سعة القناة عند طول 100 رساله |
|---|---|---|
| Drop-Tail | 93.1% | 95.7% |
| RED | 84.7% | 95.2% |
| Codel | 92.5% | 92.3% |
| Auto | 91.5% | 95.8% |

## 6. التقييم

الهدف الاساسي لخوارزمية (auto) هو الحساب الالي لطول الصف بغرض ضمان قيمة تأخير مناسبه لعمل البروتوكولات التطبيقيه على الشبكه، تشترك مع خوارزمية (auto) خوارزمية (Codel) في هذا الهدف، ولكن تسلك كل خوارزميه طريق مختلف حيث تقوم خوارزمية (Codel) بحساب زمن الزياره لكل رساله تدخل الموجه، اما خوارزمية (auto) فتقوم بحساب عدد الاتصالات التي تمر في القناة بصوره تقديريه إحتماليه.

### 1.6 حساب زمن الزياره

تقوم خوارزمية (Codel) بحساب زمن الزياره لضمان قيمة تأخير معقوله في صف الموجه، زمن الزياره هو الزمن الذي تمكثه الرساله في الموجه منذ دخولها وحتى خروجها، ولكن لحساب هذه القيمه يقوم الموجه بحساب زمن الدخول وزمن الخروج لاي رساله ليتمكن من حساب زمن الزياره الذي يساوي زمن الخروج مطروحا منه زمن الدخول، بعد حساب زمن الزياره يستطيع الموجه (Codel) ان يتخذ قرار باسقاط الرساله او عدم اسقاطها، تكمن المشكله في ان العمليه الحسابيه لزمن الزياره تستهلك معالج الموجه (Codel) حيث انه يحسب لكل الرسائل الداخله للموجه، بالرغم من ان خوارزمية (Codel) تمكنت من ضمان قيمة تأخير جيده لعمل البروتوكولات الا انها لا يمكن تطبيقها بصوره عمليه في الموجهات الحقيقيه وذلك بسبب العمليات الحسابيه الكثيفه التي تقوم بها الخوارزميه وذلك لحساب زمن الزياره.

### 2.6 حساب عدد الاتصالات الحيه

بما ان خوارزميه (Codel) لا يمكن تطبيقها بصوره عمليه بالتالي لا بد من إيجاد خوارزميه اخرى تحل محل خوارزمية

(Codel) وفي نفس الوقت تؤدي نفس الوظيفه وهي حل مشكله تضخم الصف (Bufferbloat)، قامت خوارزمية (auto) للتحكم بطول الصف بحساب عدد الاتصالات الحيه الماره عبر الموجه، لحساب عدد الاتصالات، تقوم اولا بمطابقة اي رساله داخله الى الموجه مع رساله مأخوذه بصوره عشوائيه من قائمه الاحياء الاموات (Zombie List) من عملية المطابقه تحسب الخوارزميه احتمال حدوث مطابقه (رسالتان من نفس الاتصال) او عدم مطابقه (رسالتان من اتصالين مختلفين) من احتمال حدوث المطابقه نحسب عدد الاتصالات الحيه التي تمر عبر الموجه مباشرة من الواضح ان عبء العمليات الحسابيه علي المعالج في خوارزمية (auto) اقل بكثير من عبء العمليات الحسابيه علي المعالج في خوارزمية (Codel) وبالتالي فان خوارزمية (auto) يمكن تطبيقها عمليا في الموجهات.

## 7. الخاتمه

في هذه الورقه تم ايجاد طريقه جديده وبسيطه لحساب طول الصف المناسب في الموجه، حيث يتناسب طول الصف مع عدد الاتصالات التي تمر بالموجه حسب المعادله (1)، يتم حساب عدد الاتصالات التي تمر بالموجه بصوره احتماليه وذلك عند ورود اي رساله الى الموجه، تستمد خوارزمية auto قوتها من عملية الحساب الاحتماليه وغير المكلفه لزمن معالج الموجه وذلك لعدد الاتصالات حيث انه وبعد حساب عدد الاتصالات يمكن وبكل بساطه حساب طول الصف المناسب مع عدد الاتصالات التي تدخل الموجه، هذه الطريقه البسيطه والسهله تجعل من الممكن تطبيق هذه الخوارزميه في الموجهات العمليه وذلك للتخلص من مشكلة تضخم الصفوف وزيادة التأخير المرافق له.

## 8. المراجع


[1] Gettys، J. and K. Nichols، Bufferbloat, (2011), Dark buffers in the internet. Queue, Vol. 9, No. 11, PP. 40.

[2] Nichols، K. and V. Jacobson, (2012), A Modern AQM is just one piece of the solution to bufferbloat. ACM Queue Networks, Vol. 10, No. 5.

[3] Stanojevic، R.، R.N. Shorten، and C.M. Kellett, (2006), Adaptive tuning of drop-tail buffers for reducing queueing delays. Communications Letters، IEEE. Vol. 10, No. 7, PP. 570-572.

[4] Villamizar، C. and C. Song, (1994), High performance TCP in ANSNET. ACM SIGCOMM Computer Communication Review، Vol. 24, No. 5, PP. 45-60.

[5] Appenzeller، G.، I. Keslassy، and N. McKeown, (2004), Sizing router buffers. Vol. 34.

[6] Ott، T.J.، T. Lakshman، and L.H. Wong. Sred, (1999), Stabilized red. in INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies.

[7] Jacobson، V. (1988), Congestion avoidance and control. in ACM SIGCOMM Computer Communication Review.

[8] McCanne، S. and S. Floyd, (2005), ns-LBNL network simulator.

[9] Jiang، H. and C. Dovrolis, (2005), Why is the internet traffic bursty in short time scales?, in ACM SIGMETRICS Performance Evaluation Review.

[10] Allman، M. and E. Blanton, (2005), Notes on burst mitigation for transport protocols. ACM SIGCOMM Computer Communication Review‹. Vol. 35, No. 2, PP. 53-60.

[11] Blanton، E. and M. Allman, (2005), On the impact of bursting on TCP performance، in Passive and Active Network Measurement, Springer. PP. 1-12.

[12] Shenker، S.، L. Zhang، and D.D. Clark, (1990), Some observations on the dynamics of a congestion control algorithm. ACM SIGCOMM Computer Communication Review, Vol. 20, No. 5, PP. 30-39.

[13] Floyd، S. and V. Jacobson, (1993), Random early detection gateways for congestion avoidance. Networking، IEEE/ACM Transactions on Networking, Vol. 1, No. 4, PP. 397-413.

[14] Alemu، T. and A. Jean-Marie, (2004), Dynamic configuration of RED parameters [random early detection]. in Global Telecommunications Conference. IEEE GLOBECOM'04.

[15] Christiansen، M.، et al., (2001), Tuning RED for web traffic. IEEE/ACM Transactions on Networking (TON)، Vol. 9, No. 3: PP: 249-264.

[16] Feng، W.-c.، et al. (1999), A self-configuring RED gateway. in INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies.