شبكات الند للند من أجل مشاركة الملفات*

م. هلا أمين**

د. محمد خالد شاهين*** أ.د. جان لوكا ماتزيني****

الملخص

تعتمد شبكات الند للند على التعاون بين المستخدمين والتشارك في الموارد، وقد شاع استخدام هذا النوع من الشبكات في كثير من التطبيقات، من أشهرها مشاركة الملفات. تقسم تطبيقات مشاركة الملفات، المعتمدة على بنية الند للند إلى منظومات مركزية (تعتمد على وجود مخدم مركزي لتنظيم العمل) ومنظومات موزعة، كما يمكن أن تقسم إلى منظومات بنيوية أو غير بنيوية، تبعاً لآلية إدارة المعلومات وتنظيمها ضمن المنظومة.

يقدم هذا المقال تطبيق ند لند مركزياً يُستخدم من أجل مشاركة الملفات فضلاً عن نتائج الاختبارات على هذا التطبيق، التي تؤكد ميزات شبكات الند للند من حيث التصعيد الكبير ووثوقية التطبيق.

الكلمات المفتاحية: الند للند – مشاركة الملفات – منظومات مركزية – شبكة هجينة – منظومات بنيوية – مخدم مركزي.

* أعد البحث في سياق رسالة الماجستير للطالبة هلا أمين بإشراف الدكتور محمد خالد شاهين ومشاركة الأستاذ الدكتور جان لوكا ماتزيني.

** قسم الاتصالات - كلية الهندسة الميكانيكية والكهربائية - جامعة دمشق.

*** أستاذ مساعد - قسم هندسة الإلكترونيات والاتصالات -كلية الهندسة الميكانيكية والكهربائية -جامعة دمشق.

**** أستاذ - جامعة فيرارا الإيطالية

1. المقدمة

أدى ظهور شبكات الند للند وما تتمتع به من إمكانية تصعيد scalability كبيرة، وبتكلفة منخفضة جداً، إلى تزايد اهتمام مطوري تطبيقات الإنترنت بهذه البنية. وتعتمد بنية الند للند على وجود عدد من العقد التي تتشارك في الموارد resource sharing من أجل تبادل المعلومات، إذْ تؤدي كل عقدة دور المخدم والزبون في الوقت نفسه. وتستقبل كل عقدة ضمن بنية الند للند المعلومات من عقدة أو أكثر، ثم تقوم بإعادة إرسالها إلى عقد أخرى [1-3].

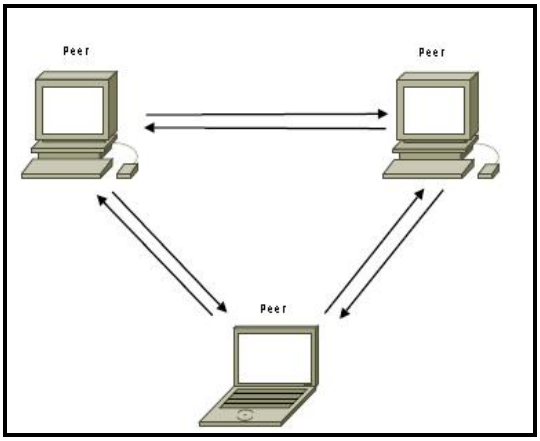
ونظراً إلى الميزات العديدة التي تتمتع بها شبكات الند للند، فقد استخدمت من أجل مشاركة الملفات عبر الإنترنت، إذْ ظهرت العديد من التطبيقات مثل تطبيق نابستر Napster المستخدم لتبادل ملفات الموسيقا [4]، وتطبيق البت تورنت BitTorrent الذي يعدُّ من أشهر تطبيقات الند للند المستخدمة من أجل تحميل download الملفات بسرعة وكفاءة [5, 6].

هَدَفَ هذا البحث إلى تصميم تطبيق مشاركة ملفات يعتمد على بنية الند للند للاستفادة من ميزات هذه البنية، وتنفيذه برمجياً، وتقييم الأداء من حيث التأخير delay وجودة quality الاستقبال.

2. المبدأ النظري

1.2. مفهوم شبكة الند للند

شبكة الند للند هي شبكة من العقد nodes لها الصلاحيات والمسؤوليات نفسها، وتتشارك في مواردها سواء المادية (مثل الطابعات، ومساحة التخزين، وعرض مجال الشبكة وغيرها)، أو المنطقية (كالخدمات) بهدف إنجاز مهام محددة مثل التشارك بالملفات، وتبادل الرسائل الفورية instant messaging والمعالجة الموزعة distributed computing [7, 8]. يبيّن الشكل (1) مثالاً على بنية هذه الشبكة. وعند انضمام عقد جديدة إلى الشبكة تضاف مواردها إلى الموارد المتاحة في الشبكة، ومن ثم يمتاز هذا النوع من الشبكات بتصعيد كبير. كما أن وجود البيانات نفسها في العديد من العقد يزيد من وثوقية المنظومة في حال إخفاق بعض العقد أو مغادرتها للشبكة. لكن هذه الميزات تزيد في الوقت نفسه من صعوبة إدارة الشبكة وتتسبب بانخفاض أمنها.



الشكل (1) مثال على بنية شبكة الند للند.

2.2. تصنيف شبكات الند للند

يمكن أن تصنف شبكات الند للند تبعاً لاستخداماتها إلى [9]:

- مشاركة الملفات
- اتصالات هاتفية
- بث الوسائط المتعددة (صوت، فيديو)

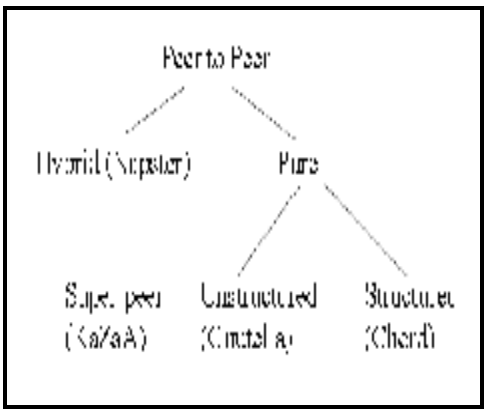
كما يمكن أن تصنف تبعاً لدرجة مركزيتها إلى [1, 10]:

- شبكة صافية pure أو مركزية centralized
- شبكة هجينة hybrid أو لامركزية decentralized

أو أن تصنف تبعاً لآلية إدارة المعلومات وتنظيمها ضمن البنية إلى [11]:

- شبكة بنيوية structured
- شبكة غير بنيوية unstructured

يبيّن الشكل (2) مخططاً لطرائق تصنيف شبكات الند للند مع بعض الأمثلة [12].



الشكل (2) تصنيف شبكات الند للند [12].

1.2.2 الشبكة الصافية

نقول عن شبكة: إنَّها صافية إذا تكونت من عناصر يعمل كل منها كمخدم وزبون في الوقت نفسه من دون وجود مخدم مركزي مسؤول عن إدارة الشبكة وتنظيمها. وتمتاز هذه الشبكة بالوثوقية وانخفاض التكلفة بسبب عدم الحاجة إلى تجهيزات بمواصفات مميزة [4].

2.2.2 الشبكة الهجينة

تشتمل على مخدم مركزي مسؤول عن بناء الشبكة والتحكم بها .

3.2. التشارك في الموارد

تُعدُّ مشاركة الموارد من أهم ميزات شبكات الند للند، التي أدت إلى انتشارها الواسع. عند انضمام عقدة جديد إلى الشبكة، تضاف مواردها سواءً الفيزيائية، مثل مساحة التخزين وعرض المجال، أو المنطقية مثل الخدمات services والمعلومات، إلى موارد الشبكة.

1.3.2 عرض المجال

يعدُّ توفير عرض المجال من أهم الميزات التي تقدمها بنية شبكات الند للند الذي يتحقق من خلال تشارك العقد بعرض المجال الخاص بها من أجل ضمان توزيع البيانات ضمن الشبكة .

2.3.2 موارد المعالجة

تعدُّ إجرائيات المعالجة processing مورداً تشاركياً آخرَ في شبكات الند للند، إذْ تقوم كل عقدة بتنفيذ مجموعة من التعليمات instructions أو بعض عمليات معالجة البيانات المطلوبة .

3.3.2 التخزين

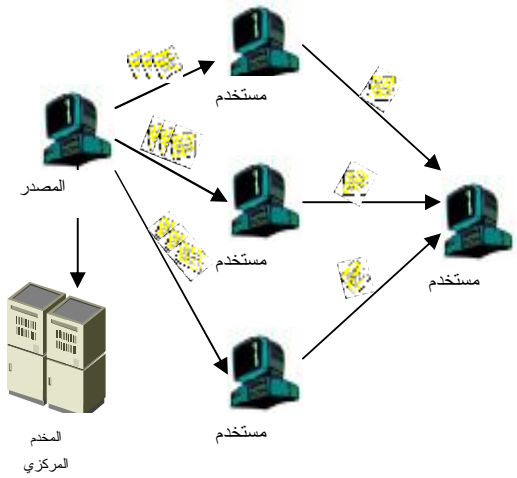
تستخدم شبكات الند للند أيضاً لبناء منظومة تخزين كبيرة، نتيجة تشارك جميع العقد بمساحة التخزين المتاحة ضمنها. فمثلاً من أجل شبكة تضم 100 عقدة، وبفرض تقديم كل عقدة مساحة تخزين 1 غيغابايت، تصبح مساحة التخزين الإجمالية للمنظومة 100 غيغابايت .

3. التطبيق العملي

1.3. البنية المقترحة للنظام

تمثل البنية المقترحة تطبيق ند للند هجيناً يعمل من أجل مشاركة الملفات بين عدد من المستخدمين.

تتألف البنية المقترحة، كما هو مبيّن بالشكل(3)، من مخدم مركزي مسؤول عن تخزين معلومات الملفات المراد مشاركتها عبر الشبكة وعن إدارة وتنظيم اتصال العقد، ومن مصدر source يملك ملفاً (أو أكثرَ) ويرغب بمشاركته عبر الشبكة، وأخيراً من مستخدمين يستقبلون الملف من المصدر ويعيدون إرساله إلى مستخدمين آخرين، أي إنَّها تعمل كعناصر تأخير relay.



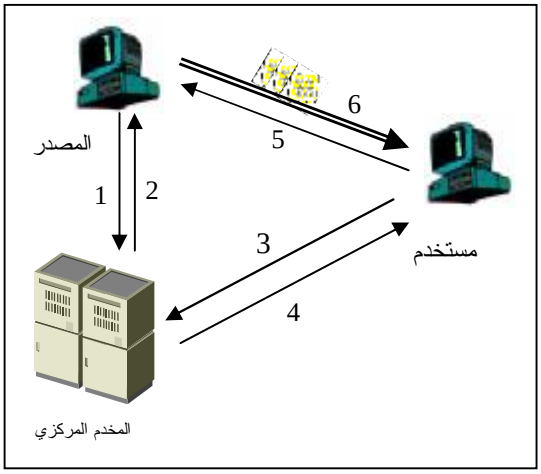
الشكل (3) البنية المقترحة للتطبيق.

آلية عمل التطبيق

عند انضمام مصدر جديد إلى الشبكة يُعرّف عن نفسه للمخدم بإرسال العنوان الخاص به، ومعلومات عن الملفات التي يمتلكها. ومن هذه المعلومات اسم الملف وطوله length. بعدها يقوم المخدم بتخزين هذه المعلومات ضمن قاعدة بياناته ثم يخصص رقم منفذ port وحيد لكل ملف ويعيد إرساله إلى المصدر الذي يقوم باستخدامه من أجل استقبال طلبات المستخدمين.

يُقسَّمُ الملف في المصدر إلى كتل بحجم ثابت توضع كل منها ضمن حاوية container تضم معلومات عن الرقم التسلسلي sequence number للكتلة وزمن إرسالها sending time. ويستفاد من المعلومات المتضمنة بالحاوية عند إعادة تجميع كتل الملف لاحقاً.

عند انضمام مستخدم جديد إلى التطبيق، فإنه يقوم بإرسال استعلام باسم الملف المطلوب إلى المخدم الذي يقوم بدوره بإعادة إرسال عناوين المصادر والعقد المناسبة التي يستطيع المستخدم الاتصال بها للحصول على الملف المطلوب، حيث يتم اختيار العقد المناسبة تبعاً للخوارزمية المقترحة والمفصلة في الفقرة (3-3). يقوم المستخدم فيما بعد بالاتصال بهذه العقد واستقبال كتل الملف منها ومن ثم إعادة تجميعها للحصول على كامل الملف. يبيّن الشكل (4) آلية العمل المقترحة.



1 : العنوان ومعلومات الملف	2 : رقم المنفذ
3 : اسم الملف المطلوب	4 : عناوين وأرقام الكتل
5 : أرقام الكتل المطلوبة	6 : تبادل الملف

الشكل (4) آلية عمل التطبيق المقترح

2.3. مكونات النظام

تتألف البنية المقترحة للتطبيق من ثلاث مكونات أساسية هي: المخدم المركزي، والمصدر، والزبون.

1.2.3 المخدم المركزي

يتركز الهدف الأساسي من وجود المخدم المركزي في تنظيم العقد ضمن المنظومة بسهولة وكفاءة، وكذلك في تخزين معلومات الملفات وعناوين العقد التي تمتلكها مما يسهل عملية البحث.

عند اتصال مصدر الملف بالمخدم فإنه يرسل عنوانه ومعلومات الملف الذي يرغب بمشاركته، عندها يقوم المخدم بتخزين هذه المعلومات ضمن قاعدة البيانات الخاصة به التي تتألف من الحقول الآتية: الرقم المعرف للملف (ID)، واسم الملف، وحجم الملف، وعنوان المصدر (IP)، ورقم المنفذ الذي يقوم المخدم بتخصيصه للملف، كما هو مبيّن بالجدول (1).

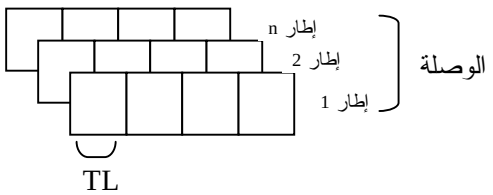
الجدول (1) بنية قاعدة البيانات في المخدم.

ID	HOST NAME	FILE NAME	FILE SIZE	Port

كما يتضمن المخدم جدولاً ثانياً يدعى جدول التخصيص allocation table حيث يخصص لكل ند سطر يضم المعلومات الآتية:

- عنوان الند الذي يمثل مفتاح الجدول (IP)
- العدد الكلي للحيزات slots ضمن إطار الوصلة (TL)، كما هو مبيّن بالشكل (5). ويحسب وفقاً للمعادلة الآتية:

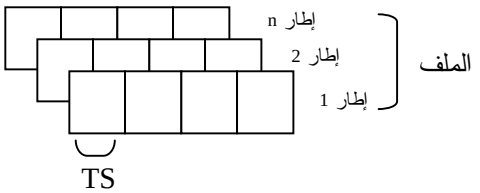
$$TL = BW_L/BW_U$$ [1]



الشكل (5) آلية تقسيم الوصلة.

- العدد الكلي للحيزات ضمن إطار المحتوى (TS)، كما هو مبيّن بالشكل (6).ويحسب وفقاً للمعادلة الآتية:

$$TS = BW_S/BW_U$$ [2]



الشكل (6) آلية تقسيم الملف.

[1] إذْ BW_L :عرض مجال الوصلة ضمن الشبكة، ويقاس بكمية المعطيات المرسلة بين نقطتين في الشبكة خلال واحدة الزمن (Bps). أما BW_U : وحدة عرض المجال، وهو قيمة اختيارية.

[2] إذْ BW_S :عرض المجال أو معدل البتات للملف ويقاس (Bps). أمَّا BW_U : فهو وحدة عرض المجال، وهو قيمة اختيارية.

- عناوين العقد التي يقوم الند بتخديمها (PC)
- شعاع تخصيص الحيزات slot allocation وهو عبارة عن أزواج مكونة من رقم الحيز للمحتوى مع رقم الحيز المخصص لها من الوصلة (VC) (link slot, content slot)، كما هو مبيّن بالشكل (7).

Vc	1	2	3	4	5
(l,c)	(1,1)	(2,2)	(3,3)	(4,1)	(5,2)

الشكل (7) شعاع تخصيص الحيزات من أجل عدد حيزات وصلة (TL=5) وعدد حيزات ملف (TS=3).

ويبيّن الجدول (2) بنية جدول التخصيص.

الجدول (2) بنية جدول التخصيص في المخدم.

IP	TL	TS	PC				VC			

يقوم المخدم بملء هذا الجدول اعتماداً على الخوارزمية المقترحة (الشكل 9).

عند استعلام أي زبون عن اسم ملف فإن المخدم يقوم بالبحث ضمن قاعدة بياناته، ثم يرسل إلى الزبون عناوين العقد المناسبة للاتصال بها، وكذلك كتل الملف المتاحة في كل عقدة من العقد وعلى أي فتحة من الوصلة يجب على المستخدم استقبالها.

2.2.3 المصدر

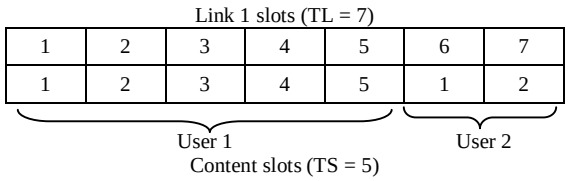
يمتلك المصدر ملفاً (أو أكثرَ) يرغب بمشاركته مع باقي مستخدمي التطبيق. في البداية يُعرف المصدر عن نفسه للمخدم من خلال إرسال عنوانه ومعلومات الملفات التي يملكها، ثم يستقبل من المخدم رقم المنفذ الواجب استخدامه من أجل كل ملف من الملفات.

بعدها يقوم المصدر بفتح المنفذ وانتظار طلبات المستخدمين، وفي الوقت نفسه يهيئ الملف إذْ يقوم

بتقسيمه إلى كتل بحجم ثابت ويضع كل كتلة ضمن حاوية مصممة لهذا الغرض.

تمتاز بنية الحاوية المقترحة بالبساطة إذ تتضمن حقلين أساسيين هما الرقم التسلسلي للكتلة الذي يستخدم من أجل إعادة ترتيب الكتل وتجميعها بشكل صحيح عند الاستقبال، وزمن إرسال الكتلة بحيث يستخدم من أجل تحديد التأخير، ومن ثَم مزامنة الكتل وتجميعها بشكل صحيح في المستقبل.

عند ورود طلب من المستخدم يقوم المصدر بتوضيع كتل الملف ضمن حيزات الوصلة المناسبة لطلبات المستخدم ومن ثم إعادة إرسالها إليه. وقد تتضمن الوصلة الواحدة كتل ملف من أجل عدة مستخدمين إِذْ يقع على المصدر مهمة تخصيص حيزات الوصلة المناسبة لكل منهم، كما هو موضح بالشكل (8).



الشكل (8) تخصيص حيزات الوصلة من أجل حيزات المحتوى.

3.2.3 المستخدم

عندما يرغب المستخدم بالحصول على ملف ما، فإنه يرسل استعلاماً باسم الملف إلى المخدم، الذي يقوم بدوره بإرسال عناوين المصادر والعقد الموجودة ضمن الشبكة التي يمكن للمستخدم الاتصال بها، وكذلك أرقام كتل الملف وحيزات الوصلة المخصصة من أجل كل مصدر أو عقدة مرسلة، عندها يقوم المستخدم بالاتصال بهذه المصادر والعقد واستقبال كتل الملف المطلوبة من كل منها؛ وذلك ضمن فتحات الوصلة link slot المخصصة لها. بعدها يقوم المستخدم بإعادة ترتيب هذه الكتل بالاستفادة من معلومات الحاوية، ثم عرضها بعد إجراء عمليات المعالجة المطلوبة، يمكن للمستخدم بعد استقبال عدد من كتل الملف أداء دور المرسل، بحيث يقوم بإعادة إرسال هذه الكتل إلى مستخدمين آخرين موجودين ضمن الشبكة.

3.3. آلية تنظيم العقد ضمن الشبكة

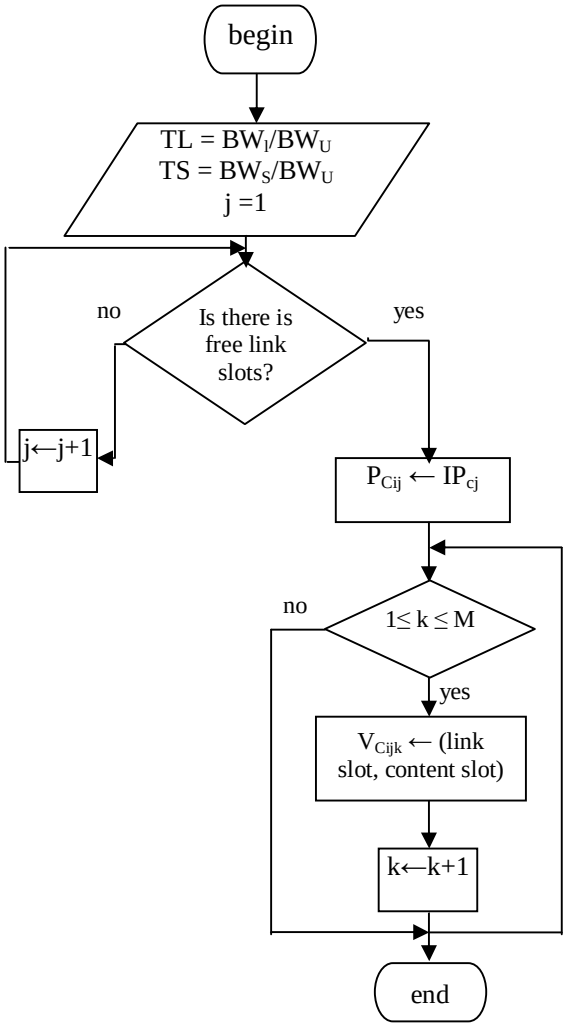
المخدم المركزي مسؤول عن تنظيم العقد ضمن الشبكة تبعاً لخوارزمية قمنا بتطويرها وسنبيّن أولاً الموسطات والرموز المستخدمة.

تجري عملية تخصيص العقد من أجل كل ملف الذي نرمز له بالرمز (c)، علماً أن كل محتوى مرتب ضمن أطر كل منها مقسم إلى عدد من الحيزات التي يرمز لعددها بـــ TS.

يُعرَّفُ كل مكون entity (مصدر أو عقدة) بالعنوان الخاص به IP_{ci}. ويخصص من أجل كل مكون السطر i-th من الجدول الذي يتضمن المعلومات الآتية:

- TL_i العدد الكلي للحيزات ضمن كل إطار من أطر الوصلة
- P_{Cij} شعاع يتضمن عناوين العقد التي يجري تخديمها بواسطة المكون i، إِذْ يشير j إلى العقد المتصلة مع المكون i.
- V_{Cijk} شعاع يضم أزواج من حيز وصلة مع حيز محتوى، إِذْ يشير إلى تخصيص حيز المحتوى ضمن حيز وصلة محددة. علماً أن k يراوح بين الواحد وعدد الحيزات المخصصة.
- M عدد حيزات الوصلة المخصصة.
- TF عدد حيزات الوصلة المتاحة.

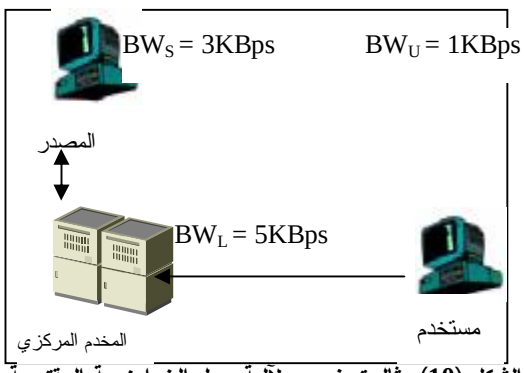
يبيّن الشكل (9) المخطط التدفقي للخوارزمية المقترحة إِذْ TL العدد الإجمالي للحيزات ضمن الإطار و TS العدد الإجمالي لحيزات المحتوى ويحسبان بافتراض عرض مجال الوصلة BW_L والمحتوى BW_S. ويمثل BW_U وحدة عرض المجال التي سنفترضها مساوية إلى 1KBps.



الشكل (9) المخطط التدفقي للخوارزمية المقترحة.

مثال توضيحي لعمل الخوارزمية

عند اتصال مستخدم جديد بالمخدم، كما هو مبيّن بالشكل (10) ووفقاً للخوارزمية السابقة، تُنفَّذُ الخطوات الآتية.



الشكل (10) مثال توضيحي لآلية عمل الخوارزمية المقترحة

1- تحديد عدد حيزات الوصلة وحيزات الملف من أجل وحدة عرض مجال اختيارية BWU = 1KBps.

$TL = BW_L/BW_U = 5KBps/1KBps = 5$

$TS = BW_S/BW_U = 3KBps/1KBps = 3$

2- من أجل العقدة الأولى j=1 التي تمثل المصدر في هذه الحالة، يكون عدد الوصلات المتاحة ويرمز لها بـــ TF يساوي 5. ومن ثَمَّ يكون شرط وجـــود حيـــزات وصـــلة متاحة محققاً.

3- يُضَافُ عنوان المستخدم إلى شعاع العناوين التي يخدمها المصدر كما يأتي:

$P_{C11} \leftarrow client_IP$

4- تُخَصَّصُ حيزات الملف الغير مخصصة سابقاً ويرمز لها بـــ M وعددها 3 إلى الحيزات المتاحة من الوصلة وتساوي TL=5 كما يلي:

$k=1 \rightarrow V_{C111} = (l1,c1)$
$k=2 \rightarrow V_{C112} = (l2,c2)$
$k=3 \rightarrow V_{C113} = (l3,c3)$

5- عند انضمام مستخدم آخر، تُعَادُ الخطوات السابقة لكن مـــع عـــدد حيـــزات وصـــلة متـــاح مـــساوي إلـــى TF= 5-3=2.

4.3. مقارنة بين البنية المقترحة وبعض تطبيقات الند للند الشهيرة

سنبين فيما يلي مقارنة بين البنية المقترحة وبعض أشهر تطبيقات الند للند المستخدمة في مشاركة الملفات، وهي: نابستر، وغنوتلا، وبت تورنت.

نابستر: يعتمد التطبيق على مخدم مركزي مسؤول عن تخزين الملفات وآلية اتصال العقد مع بعضها بعضاً، مما يجعل عملية البحث فعالة وسريعة. يقوم المستخدم بالحصول على عناوين العقد التي تملك الملف من المخدم، ثَمَّ اختيار إحدى هذه العقد واستقبال كامل الملف منها.

غنوتلا: تُخَزَّنُ الملفات في كل عقدة بشكل محلي. ويقوم المستخدم بإرسال رسائل لاستكشاف العقد الموجودة في

الشبكة، حتى يصل إلى العقد التي تملك الملف واستقبال الملف منها.

يمتاز تطبيق غنوتلا بتكلفة بحث موزعة على عدة عقد. في حين تعاني المنظومة من وجود تحميل زائد على الشبكة نتيجة تعويم الرسائل بين العقد جميعها، وكذلك زيادة حجم المعالجة المتمثل في بحث كل عقدة عن الملفات ضمن قاعدة معطياتها المحلية.

بت تورنت: يُعدُّ التطبيق بت تورنت من أشهر تطبيقات الند للند المستخدمة في تحميل الملفات بسرعة وكفاءة، إذْ يعتمد على تقسيم الملف إلى أجزاء متساوية الحجم تدعى كتلاً، وذلك لتسهيل عمليات الإرسال وتسريعها. يقوم كل ند في التطبيق بتحميل أجزاء الملف التي يحتاج إليها من العقد التي تمتلكها، وكذلك إرسال الأجزاء التي يمتلكها إلى العقد التي تحتاج إليها.

البنية المقترحة: تضم البنية المقترحة مخدماً مركزياً مسؤولاً عن تخزين الملفات وتنظيم آلية بناء الشبكة، لتسهيل آلية البحث والتحكم بالشبكة وتسريعها، كما هو الحال في نابستر. بالمقابل يُقسَّمُ الملف إلى أجزاء متساوية بالحجم تدعى كتلاً، بشكل مشابه للبت تورنت. يقوم المستخدم باستقبال أجزاء الملف من عدد من العقد، ومن ثم إعادة ترتيب أجزاء الملف وإعادة تجميعها.

4. البيئة البرمجية المستخدمة في التطبيق

استخدمنا لتنفيذ التطبيق برمجياً لغة السي (C)، وذلك لكتابة ثلاثة برامج أساسية لتمثيل كل من المخدم، المسؤول عن تنظيم الشبكة، والمصدر الذي يمتلك الملف، والمستخدم.

التنفيذ البرمجي للمخدم

يعدُّ المخدم مسؤولاً عن تنظيم اتصال العقد مع بعضها بعضاً وعن عملية تخزين الملفات والتحم بعملية البحث. من الممكن تقسيم برنامج المخدم إلى ثلاثة أقسام أساسية وهي: تهيئة الاتصال، والتعامل مع المصدر، والتعامل مع المستخدم.

التنفيذ البرمجي للمصدر

يمتلك المصدر ملفاً واحداً أو أكثر ويرغب بمشاركتها عبر الشبكة. في البداية، يعالج برنامج المصدر ثلاث نقاط أساسية وهي: الاتصال مع المخدم، والاتصال مع المستخدم، وتهيئة الملف وتقسيمه إلى أجزاء.

التنفيذ البرمجي للمستخدم

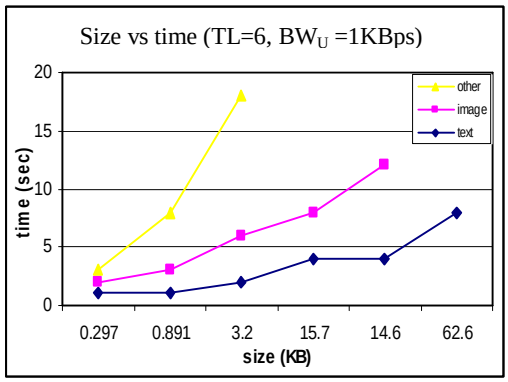
يقوم برنامج المستخدم بتأمين الاتصال مع كل من المخدم والمصدر، وكذلك تقع عليه مسؤولية استقبال أجزاء الملف وإعادة ترتيبها بشكل صحيح.

5. النتائج

اِستُخدِمَ نظام التشغيل لينوكس LINUX لبناء شبكة افتراضية من عدة مستخدمين بهدف دراسة أداء المنظومة. من ثم إجراء عدة قياسات على قيم تغيّر الأداء بدلالة الموسطات المختلفة ومنها: عدد حيزات الوصلة (TL)، ووحدة عرض المجال (BW_U)، وعدد المستخدمين للتطبيق.

بعد ذلك، تُرسَلُ عدة ملفات بأنواع وأحجام مختلفة وأجريت القياسات على الزمن اللازم لاستقبال كل منها، مع افتراض أن عدد الحيزات للوصلة TL يساوي 6 حيزات ووحدة عرض المجال BW_U مساوية إلى 1KBps. قمنا اختيار هذه القيم لعدد الحيزات ووحدة عرض المجال بشكل افتراضي، وبما يتناسب مع محدودية الموارد resource المتاحة في الحاسب الذي استخدم لإجراء المحاكاة. ومن ثم أجرينا اختبارات لدراسة تأثير تغيير هذه القيم في أداء التطبيق.

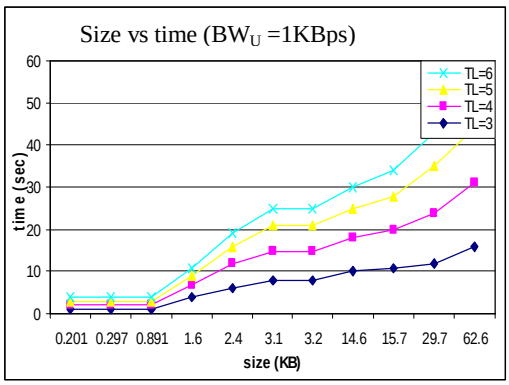
ويبيّن الشكل (11) الزمن اللازم للاستقبال بدلالة حجم الملفات من أجل ملفات بأنواع مختلفة، وعدد حيزات وصلة مساوي 6، ووحدة عرض مجال مساوية إلى 1KBps.



الشكل (11) الزمن المقيس من أجل ملفات بأنواع وأحجام مختلفة.

من النتائج السابقة نلاحظ ازدياد الزمن اللازم لاستقبال الملفات مع ازدياد حجم الملف، وكانت النتائج متقاربة من أجل أنواع الملفات المختلفة. فمن أجل زيادة حجم الملف بمقدار 200% ازداد زمن الاستقبال بمقدار 150%. نلاحظ أن زمن الاستقبال للملفات هو الأقل نظراً إلى انخفاض عمليات المعالجة المطلوبة لتقسيم الملف ومن ثم إعادة تجميعه، مقارنة بالصور وغيرها.

كما يبيّن الشكل (12) الزمن اللازم للاستقبال بدلالة حجم الملفات من أجل عدد حيزات وصلة مختلفة، ووحدة عرض مجال مساوية إلى 1KBps.

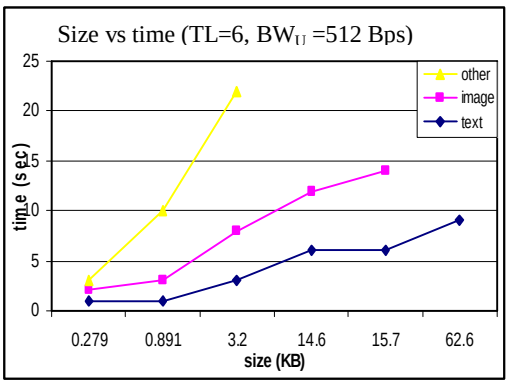


الشكل (12) الزمن المقيس من أجل ملفات بأحجام مختلفة وعدد حيزات مختلفة للوصلة.

نلاحظ في الشكل (12) تناقص الزمن اللازم لاستقبال الملفات مع ازدياد عدد حيزات الوصلة المتاحة، فمع ازدياد عرض مجال الوصلة بنحو 65% تناقص الزمن بمقدار 50%.

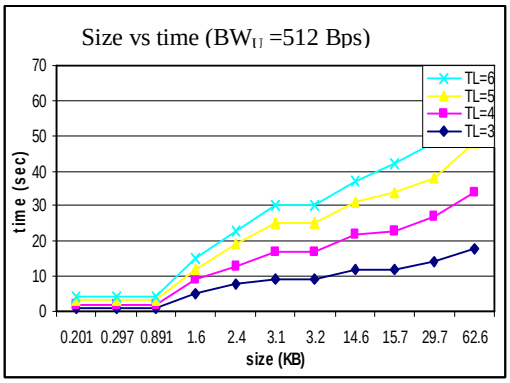
من الملاحظ أن ازدياد عدد حيزات الوصلة المستخدمة يقلل من زمن الاستقبال للملف، نظراً إلى زيادة حجم كتل الملف الممكن إرسالها في كلِّ ثانية.

ويبيّن الشكل (13) الزمن اللازم للاستقبال بدلالة حجم الملفات من أجل أنواع ملفات مختلفة، ووحدة عرض مجال مساوية إلى 512Bps.



الشكل (13) الزمن المقيس من أجل ملفات بأنواع وأحجام مختلفة.

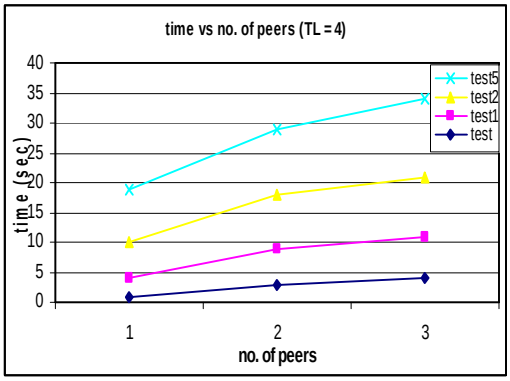
كما يبيّن الشكل (14) الزمن اللازم للاستقبال بدلالة حجم الملفات من أجل عدد حيزات وصلة مختلفة، ووحدة عرض مجال مساوية إلى 512Bps.



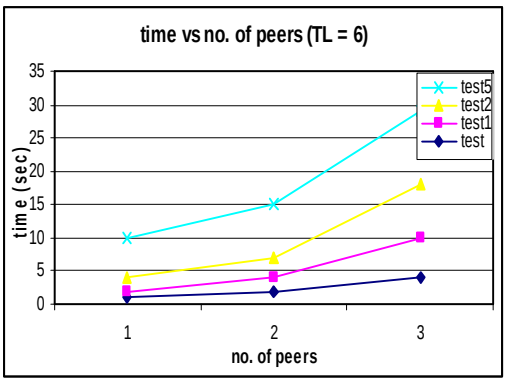
الشكل (14) الزمن المقيس من أجل ملفات بأحجام مختلفة وقيم مختلفة لعدد حيزات الوصلة TL.

بمقارنة الشكلين (11-12) من أجل وحدة عرض مجال مساوية إلى 1KBps بالشكلين (13-14) من أجل وحدة عرض مجال 512Bps، نلاحظ ازدياد زمن استقبال الملف عند اختيار وحدة عرض مجال أقل. من الملاحظ أن ازدياد عدد حيزات الوصلة المستخدمة يقلل من زمن الاستقبال للملف، نظراً إلى زيادة عدد كتل الملف الممكن إرسالها في كلِّ ثانية، وكذلك عدد المستخدمين الممكن تخديمهم بشكل متزامن.

أخيراً، جرت زيادة عدد العقد المستقبلة (المستخدمين) وقياس الزمن المطلوب لاستقبال الملف.

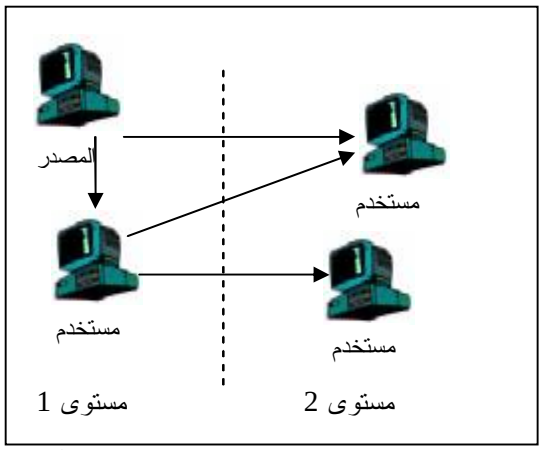


الشكل (15) الزمن المقيس من أجل ملفات نصية بأحجام مختلفة؛ وذلك مع ازدياد عدد المستخدمين، باعتبار TL = 4.



الشكل (16) الزمن المقيس من أجل ملفات نصية بأحجام مختلفة؛ وذلك مع ازدياد عدد المستخدمين، باعتبار TL = 6.

نلاحظ من الشكلين (15-16) أن ازدياد عدد المستخدمين المنضمين للشبكة يؤدي إلى ازدياد الزمن المطلوب للاستقبال ولكن بنسبة مقبولة لا تتجاوز 13%. تقود زيادة عدد المستخدمين إلى ازدياد زمن استقبال الملف، إذْ يزداد عدد مستويات التأخير، وبالتالي يحتاج المستخدم إلى الانتظار حتى يستقبل المستخدم السابق كتل الملف ومن ثم يعيد إرسالها له. كما يوضح الشكل (17).



الشكل (17) آلية اتصال العقد ضمن الشبكة ومستويات التأخير.

6. الخاتمة والأعمال المستقبلية

هذا المقال قدم بناء تطبيق ند لند من أجل مشاركة الملفات يعتمد على البنية المركزية من أجل تسهيل آلية بناء الشبكة وتصميمها، وكذلك تسهيل آلية البحث وتسريعها.

كما اقتُرِحَ تطوير خوارزمية جديدة لضمان أفضل اتصال بين المستخدمين بأقل تأخير وأفضل استخدام للموارد. وطُوِّرَتْ آلية بناء الشبكة بحيث تُستَخْدَمُ كامل حيزات الوصلة المتاحة لكل عقدة، وذلك لاستغلال موارد الشبكة بشكل فعال. كما أن اختيار العقد يقلل من عدد مستويات التأخير بين المستخدم والمصدر، ومن ثَمَّ يكون زمن استقبال الملف منخفضاً مقارنة بغيرها من التطبيقات.

اسْتُخلِصَتْ عدة نتائج منها ازدياد زمن الاستقبال للملف مع ازدياد حجم الملف أو عدد المستخدمين، لكن نقصانه عند زيادة عدد حيزات الوصلة المتاحة.

لاحقاً سنقوم بتطوير التطبيق من أجل بث الفيديو، وكذلك إلغاء المخدم المركزي لزيادة تصعيد المنظومة وتجنب وجود نقطة إخفاق واحدة في المنظومة.

المراجع:

1. A. Vasudeva, Sandeepan, and N. Kumar, "PASE: P2P network based academic search and file sharing application," presented at 1st International Conference on Computational Intelligence, Communication Systems and Networks, (CICSYN'09), 2009.
2. D. Ciullo and e. al., "Network awareness of P2P live streaming applications: A measurement study," presented at IEEE transaction on multimedia, Jan. 2010.
3. T. Small, B. Li, and B. Liang, "Topology affect the efficiency of network codeing in peer-to-peer network ",China, May, 2008.
4. H. H. Tanaka, "Post-Napster: peer-to-peer file sharing systems current and future issues on secondary liability " presented at Loyola of Los Angeles futertainment law review, 2001.
5. B. Cohen, "Incentives build robustness in BitTorrent," presented at 1st workshop on economics of peer-to-peer systems, June 2003.
6. G. Sakaryan and H. Unger, "Influence of the decentralized algorithms on topology evolution in P2P distributed networks," presented at In design, analysis and simulation of distributed systems [DASD 2003], 2003.
7. R. Schollmeier, "A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications " presented at 1st International conference on peer-to-peer computing, (P2P'01)2001 ,
8. K. Aberer and M. Hauswirth, "Improving data access in P2P systems," presented at IEEE Internet computing Switzerland, 2002.
9. T. R. Somro, M. S. Laghari, and H. Wahba, "A2A share: Towards Multilingual Academic P2P," presented at International conference on sociality and economic development, (IPEDR), 2011.
10. D. Verma, "Content distribution networks ", Wiley, 2002.
11. T. Takabatake and Y. Komano, "A P2P File Sharing Technique by Indexed-Priority Metric," presented at World Academy of Science, Engineering and Technology, 2009.
12. Y. Lin, "Peer-to-Peer Systems".

مسرد المصطلحات

A	
Allocation table	جدول تخصيص
B	
Bandwidth	عرض مجال
BitTorrent	بت تورنت
C	
Centralized	مركزي
Client	مستخدم
Container	حاوية
Content	محتوى
D	
Decentralized	لا مركزي
Delay	تأخير
Distributed computing	معالجة موزعة
Download	إتيان
E	
Entity	مكون
F	
File sharing	مشاركة الملفات
H	
Hybrid	هجين
I	
Instant messaging	تبادل الرسائل الفورية
Instructions	تعليمات
L	
Linux	نظام تشغيل لينوكس
Link	وصلة
N	
Napster	نابستر
Node	عقدة
P	
Peer-to-Peer network	شبكة الند للند
Port	منفذ
Processing	معالجة
Pure	صافي
Q	
Quality	جودة
R	
Relay	عنصر تأخير
Resource	موارد
Resource sharing	مشاركة الموارد
S	
Scalability	تصعيد
Sending time	زمن الإرسال
Sequence number	رقم تسلسلي
Server	مخدم
Service	خدمة
Slot	حيز
Slot allocation	تخصيص الحيزات
Source	مصدر
Structured	بنيوي
U	
Unstructured	غير بنيوي

تاريخ ورود البحث إلى مجلة جامعة دمشق 2011/11/7