ABES66

Republic Of YEMEN

Thamar University

Faculty of Computer Science and Information System

Computer Science Dept.

Level 3



**اعداد الطالب /**

**عبدالناصر سعد محمد الزوبه**

# الدكتور :منير القباطي .

**الفهرس:-**

**المؤلف .........................................................................................................4**

[تجريد 5](#_Toc354772408)

[مقدمة 6](#_Toc354772409)

[Chord p2p protocol 7](#_Toc354772411)

[خصائص Chord 8](#_Toc354772411)

[كيفية الربط بين العقد والقيم 8](#_Toc354772412)

[قيود بروتوكول chord 8](#_Toc354772413)

[جداول Hash الموزعة( [distributed hash table](http://en.wikipedia.org/wiki/Distributed_hash_table)) 8](#_Toc354772414)

[مقارنات 9](#_Toc354772414)

[Napster, Gnutella etc. vs. Chord 9](#_Toc354772415)

[DNS vs. Chord 9](#_Toc354772416)

[مشاكل عالجها البروتوكول 9](#_Toc354772417)

[مثال لتطبيق يستخدم الـ Chord (النسخ التعاوني) 10](#_Toc354772418)

[أساسيات بروتوكول Chord 10](#_Toc354772419)

[صلابة موقع المفتاح Scalable Key Location 11](#_Toc354772412)

[خصائص عملية Hashing الثابتة ((Consistent Hashing 11](#_Toc354772413)

[آلية إعطاء عقدة ومفتاح معرف خاص بها 11](#_Toc354772414)

[انضمام ومغادرة العقد 11](#_Toc354772415)

[تسريع عملية البحث Acceleration of Lookups 12](#_Toc354772416)

[خصائص جدول الأصابع 13](#_Toc354772417)

[انضمام العقد باستخدام جدول الأصابع 14](#_Toc354772418)

[أسباب عدم الاستقرارية: العمليات المتزامنة والفشل 15](#_Toc354772419)

[عملية Stabilization بعد انضمام عقدة 15](#_Toc354772412)

[الاسترداد بعد الفشل 15](#_Toc354772413)

[بروتوكول Chrod من وجهة نظر رياضية 16](#_Toc354772414)

[نتائج التجارب 17](#_Toc354772415)

المصادر والمراجع ............................................................................17

**المؤلف / عبدالناصر سعد محمد الزوبه من مواليد 1987 ,**

**قرية الزوب \_ رداع \_ اليمن**

**التحق بالتعليم الاساسي في مدرسة الوحده بالزوب في قريته ثم ألتحق بالثانويه العامه في مدينة رداع , بعد ذالك انتقل الى العاصمه صنعاء لدراسه وتعلم اللغة الانجليزيه في المعهد الامريكي ( يالي ) لمدة عام .**

**بعد ذالك انتقل الى محافظه ذمار لمواصلة التعليم الجامعي في جامعة ذمار كلية علوم الحاسوب وتقنية المعلومات قسم تقنية المعلومات**



1. تجريد

رغم ما تتميز به أنظمة الند للند من مميزات عالية كثيرة منها:- من القدرة على الخزن الفائض, الديمومة, اختيار الخدمات القريبة, السرية, سرعة البحث, المصداقية,

إلا أن هنالك مشكلة في تطبيقات الند للند حيث انه كان من الصعب تحديد موقع العقد التي تحتوي على بيانات معينة بشكل كفؤ ولحل هذه المشكلة ظهر نظام (Chord p2p protocol)الذي يوزع المعلومات بين العقد المختلفة.

و الوظيفة الأساسية ل( Chord p2p protocol) هو إعطاء المفاتيح و تخطيط العقد و Chord يوفر توزيع سريع حسب الوظائف الأساسية لتخطيط المفاتيح التي ارتبطت بالعقد ,

موقع بياناتِ يُمْكِنُ أَنْ يطبق بسهولة على قمةِ الوتر من خلال مُشَارَكَة المفتاح مع كُلّ مادة بياناتِ، ويتم خْزنُ زوج(المفتاحَ / مادةِ بياناتِ) في العقدةِ .

يتألم Chordبشكل كفء بينما تَنضمُّ العُقَدُ إليها وتَتْركُ النظام، ويُمْكِنُ أَنْ يُجيبَ عن الاستفسارات حتى إذا حدث في النظامِ تغير بشكل مستمر النَتائِج مِنْ التحليلِ النظريِ، والمحاكاة، ترى من خلال التجاربَ بأنّ Chord قابل للتوسع.

وسنتحدث في موضوعنا هذا عن كل ما يخص أل( Chord p2p protocol)من حيث الأتي:-

♦ تعريفة.

♦عملة .

♦خصائصه .

♦مقارنته ببروتوكولات أخرى.

♦الخوارزمية التي يتبعها في عملة.

1. المقدمة

الأنظمة وتطبيقات الند للند عبارة عن تطبيق للأنظمة الموزعة بدون أيّ سيطرة مركزية، حيث أنَّ البرامج التي يدير كُلّ عقدة بشكل متكافئ في الوظيفة.

ورغم ماتتميز به أنظمة و التطبيقاتِ الند للندِ من مميزات كثيرة وعالية منها:القدرة على الخزن الفائض،الديمومة،اختيار الخدمات القريبة،السرية،سرعة البحث،المصداقية

يستعمل Chordأل Consistent Hashing (الثابت أو المتماسك)لتزويد العقد بالمفاتيح وذلك حينما تدخل عقد جديدة إلى النظام وتلك المفاتيح توزع بانتظام على العقد .

يحتفظ هذا النظام بمعلومات حول العقد الأخرى وكل عقدة تستقبل المعلومات من العقد الأخرى لذلك هو قابل لتوسيع ,عموما فأن بروتوكول (Chord p2p protocol)يجعل من عملية تحديد العقد وعملية البحث تتم بأسلوب اثر كفاءة .

3-[[1]](#footnote-2)Chord p2p protocol

Chord 1-3 و عمله:

Chord هو خوارزمية وبروتوكول جدولة التجزئة الموزعة للأنظمة الند للند.

وجدولة التجزئة الموزعة تخزن القيم الرئيسية الزوجية عن طريق تخصيص مفاتيح للحواسيب المختلفة تعرف بالعقد (nodes) .العقدة ستخزن القيم لكل المفاتيح التي هي مسئولة عنها .

ويقوم Chord بحل مشاكل تحديد مواقع البيانات المطلوبة في مجموعة من العقد الموزعة ، مع الأخذ بعين الاعتبار الوصول والمغادرة المتكررة للعقد.

Chord تحدد كيف ستخصص المفاتيح للعقد(nodes).وكيف العقد(nodes) ستكتشف القيمة للمفتاح المعطى بواسطة أول موقع للعقدة(node) المسئولة عن هذا المفتاح.

ويمكن تعريفه باختصار انه خدمة بحث لشبكة ند للند.

Chord واحدة من أربع بروتوكولات لجدولة التجزئة الموزعة.

أن العملية الرئيسية في اغلب أنظمة ند لند هو عملية تحديد مواقع البيانات فقط،

وهذا البروتوكول يدعم توفير عملية تحديد مواقع البيانات عبر عملية واحدة فقط ، إذا أعطي المفتاح فإنه يربط هذا المفتاح بالعقدة المطلوبة .

نستطيع القول أنها خوارزمية تعمل بين طرفين على مبدأ (Hash Table) .

Hash Table دائما يحتفظ بقيمتين وتسمى القيمة المفتاحية بحيث يتم إعطاء كل عقدة قيمة مفتاحيه وتقوم العقدة بتخزين القيم المفتاحية لبعض العقد الأخرى في الشبكة .

ودور هذا البروتوكول في الواقع هو كيفية إسناد القيم المفتاحية للعقد في الشبكة ، بالإضافة إلى كيفية اكتشاف القيمة من قبل العقدة بناء على المفتاح المعطى له سلفاً عن طريق البحث عن العقدة صاحبة ذلك المفتاح .

1. 2 خصائص Chord

* البساطة، والصحة، والأداء
* كل عقدة تحتاج إلى معلومات توجيه routing information عن بعض العقد فقط على الشبكة وليس كلها .
* تحل مشكلة البحث عبر إرسال رسائل إلى عقد أخرى سواء كانت ( بالتكرار أو الاستدعاء الذاتي ).
* تقوم بصيانة معلومات التوجيه routing information عند كل انضمام أو مغادرة إحدى العقد للنظام .

3-3 كيفية الربط بين العقد والقيم

* الطرق الاعتيادية توفر ربط مباشر بين المفاتيح والقيم .
* قد تكون القيم عناوين أو مستندات ، أو بيانات أخرى .
* برتوكول Chord يقوم بربط القيم عبر تخزين كل زوج من القيمة /المفتاح في عقدة للمفتاح الذي يرتبط بها .

3-4 قيود بروتوكول chord

* جدول Finger يتبع الطريقة الخطية للبحث عن المفاتيح أو القيم باتجاه عقارب الساعة، ومن المعلوم أن البحث بالطريقة الخطية يتطلب المرور بالعناصر عنصر عنصر.
* وبفرض وجود المفتاح الذي يتم البحث عنه في عقدة موقعها في أخر الحلقة عند الفحص باتجاه عقارب الساعة فإن هذا البحث يأخذ الكثير من الوقت، بذلك يخفض من كفاءة البروتوكول. لذا أصبح من المهم جداً إنقاص كمية الوقت المطلوب للبحث عن العقدة.
* هذا الأمر يجب أن يأخذ بعين الاعتبار لأن الهدف الأساسي من أي بروتوكول هو الكفاءة وسرعة البحث عن العقد المحتوية للمفاتيح.

3-5 جداول Hash الموزعة( [distributed hash table](http://en.wikipedia.org/wiki/Distributed_hash_table))

* جداول الموزعة تستخدم للأنظمة الموزعة والتي تزودها بخدمة البحث .
* كل جدول يحوي ( المفتاح , القيمة ) وكل عقدة تعطى مفتاح وقيمة خاص بها .
* الجدول مسئول عن توليد المفاتيح وربطهم بالقيم الموجودة في العقد الموزعة ، وتتم بطريقة يمكن عن طريقها تفادي العرقلة الناتجة عن التغير في مجموعة المشاركين .
* جداول الهاش الموزعة يمكن أن تستخدم لبناء الشبكات الكبيرة لأنها توزع بانتظام خلال جميع العقد. وتتم بطريقة تسمح بزيادة عدد العقد في الشبكة دون حدوث تعارض أو مشاكل .

1. مقارنات:

Napster, Gnutella etc. vs. Chord 1-4

* مقارنة بالـ Napster التي لديها خادمات مركزية ، فإن Chord تتجنب نقاط التحكم والفشل المركزية عبر تقنية لامركزية .
* مقارنة بالـ Gnutella واستخدامها الكبير للـ broadcasts ، فإن chord تتفادى النقص في القدرة على التوسع باستخدام عدد قليل من المعلومات الهامة لأجل عملية التوجيه .

DNS vs. Chord 2-4

DNS

Chord

* يمكنها توفير نفس الخدمة حيث يكون الاسم هو المفتاح key والعنوان هو القيمة value
* لا تتطلب خوادم خاصة
* لا تفرض هيكل للتسميات
* يمكن أن تستخدم أيضاً لإيجاد البيانات الغير مرتبطة بآلات معينة
* تعمل على الربط بين اسم المضيف host name وعنوان الـ IP Address
* تعتمد على مجموعة من الخوادم الخاصة
* التسميات فيها تعكس الحدود الإدارية
* خاصة بإيجاد أسماء المضيفين أو الخدمات

5- مشاكل عالجها البروتوكول

* اللامركزية Decentralization: في نظام ند لند باستخدام Chord لا وجود لخادم مركزي أو نظير مميز. كل عقدة لها نفس الأهمية كأي عقدة أخرى . لذا فإن النظام متين جداً ، لكونه لا يحوي نقطة وحيدة من الفشل أو الإخفاق .
* الموفورية Availability : حيث يقوم بتعديل جداوله الداخلية تلقائياً من أجل ضمان أن العقد المسئولة عن المفاتيح يمكن إيجادها دائماً .
* القياسية Scalability : تكلفة البحث عن القيمة ذات نمو لوغاريتمي حسب عدد العقد في الشبكة، لذا يمكن استخدامه للأنظمة الكبيرة جداً .
* توازن الحمل Load balance : يستخدم Chord وظيفة الـ hash الثابتة لتخصيص مفاتيح للعقد ، يتم توزيع المفاتيح بشكل متساوي على العقد .
* التسمية المرنة Flexible naming : حيث لا توجد أي قيود على هيكل التسميات للمفاتيح، وتوجد مرونة عالية في كيفية ربط الأسماء بالمفاتيح دون قيود .

6-مثال لتطبيق يستخدم الـ Chord (النسخ التعاوني):

الشكل(1)

* في الشكل (1)هو عبارة عن تطبيق للنسخ التعاوني يستخدم بروتوكول Chord.
* الطبقة العليا في هذا التطبيق توفر واجهات للمستخدم .
* نظام الملفات File System في هذا التطبيق يربط العمليات في الطبقة العليا بعمليات المستوى الأدنى .
* كتلة التخزين Block Store في هذا التطبيق تستخدم Chord لتحديد العقدة المسئولة عن تخزين الكتلة المطلوبة ثم تتخاطب مع الخادم Server على تلك العقدة .

7-أساسيات بروتوكول Chord

* يحدد كيف يتم إيجاد مواقع المفاتيح.
* ويحدد كيف يتم انضمام العقد إلى النظام.
* وكيف يتعافى من الفشل أثناء مغادرة العقد.

Consistent Hashing 1-7

دالة الهاش Hash Function تقوم بإعطاء كل عقدة ومفتاح معرّف خاص باستخدام دالة هاش أساسية مثل:

ID(node) = hash(IP, Port)

ID(key) = hash(key)

7-2 خصائص عملية Hashing الثابتة ((Consistالمفاتيح.g :

* تعمل على توازن الحمل( (Balances Load:

حيث تعطي جميع العقد تقريباً نفس العدد من المفاتيح .

* عندما تنضم العقدة رقم n إلى الشبكة، فالمطلوب فقط O(1/n) من المفاتيح التي يجب أن يتم تغيير مكانها وليس جميع المفاتيح.

7-3آلية إعطاء عقدة ومفتاح معرف خاص بها:

شكل (2)

إن آلية إعطاء كل عقدة ومفتاح معرف خاص بها تكون كالتالي:

في البداية يتم إعطاء كل عقدة معرف خاص رقم 1ومن ثم يتم ربط المفتاح رقم 1 بالعقدة رقم 1 في دائرة المعرفات.

أما المعرف رقم 2 يعطى المفتاح رقم 3 ويتم توجيه المفتاح رقم 3 من خلال المعرف إلى العقدة رقم 3 حتى يتم الارتباط بها.

المفتاح رقم 6 يعطى المعرف رقم 6 الذي سيقوم بتوجيه إلى العقدة رقم 0.

7-4 انضمام ومغادرة العقد:

شكل (3)

بمجرد انضمام العقدة رقم 7 إلى النظام يتم تحويل المفتاح رقم 6 إلى العقدة 7 للارتباط بها حيث إنها الأقرب

وبمجرد خروج العقدة 1 يقوم المعرف بتسليم المفتاح للعقدة الوريثة لها 3 .

7-5 صلابة موقع المفتاح Scalable Key Location

* فقط كمية صغيرة من معلومات التوجيه Routing Information تكفي لتطبيق Consistent Hashing في الشبكة .
* كل عقدة تحتاج فقط لمعرفة العقدة اللاحقة لها Successor Node في الدائرة .
* الاستعلامات لمعرّف معين يمكن تمريره في الدائرة عبر مؤشرات العقد اللاحقة Successor Pointers .

7-6 تسريع عملية البحث Acceleration of Lookups

* كيف يتم تسريع عملية البحث في بروتوكول chord؟
* يتم تسرع عملية البحث بإضافة معلومات توجيه Routing Information إضافية .
* كل عقدة تضيف جدول توجيه Routing Table فيه “m ” مدخل (entry) على الأكثر

(حيث n= 2m ) ويسمى بجدول الأصبع Finger Table .

* المدخل رقم i في الجدول للعقد n يحتوي على قيمة أول عقدة s والتي تلحق العقدة n بأعلى الأقل 2i-1 في دائرة المعرّفات (الجدول في الشكل(2) يوضح ذلك).
* *s* = successor(n + 2i-1) % 2
* وتسمى s بالأصبع رقم i للعقدة n

شكل(4)

7-7خصائص جدول الأصابع

* كل عقدة تخزن معلومات حول عدد صغير من العقد الأخرى ، وليس جميعها، وكلما كانت العقدة قريبة منها كلما علمت عنها معلومات أكثر .
* جدول الأصابع لعقدة ما عموماً لا يحتوي على معلومات كافية لتحديد اللاحق Successor لمفتاح ما k
* الاستعلامات المتكررة للعقد التي تسبق المفتاح المطلوب ستؤدي في الأخير إلى اللاحق Successor لهذا المفتاح .

7-8انضمام العقد باستخدام جدول الأصابع

7-9 مغادرة العقد باستخدام جدول الأصابع

7-10أسباب عدم الاستقرارية: العمليات المتزامنة والفشل

* يُستخدَم بروتوكول الاستقرار الأساسي Basic Stabilization Protocol من أجل إبقاء المؤشرات إلى العقد اللاحقة nodes’ successor pointers مُحدثة، وهذا يضمن صحة عمليات البحث
* هذه المؤشرات يمكن استخدامها للتحقق من صحة مدخلات جدول الأصبع Finger Table Entries
* كل عقدة تجري عملية Stabilize بشكل منتظم لإيجاد العقد الجديدة التي انضمت للشبكة

7-11 عملية Stabilization بعد انضمام عقدة

* العقدة n تنضم إلى الشبكة
  + n تجعل السابق لها predecessor=nil
  + n تجعلns لاحق لها successor
  + n تخبر ns بأنها سابق لها predecessor
  + ns تجعل n لاحق لها predecessor
* العقدة np تجري عملية Stabilize 
  + np تسأل ns عن السابق لها predessor (وهو بالطبع n)
  + np تجعل n لاحق لها successor
  + np تخبر n بأنها أصبحت لاحق لها
  + n تجعل np لاحق لها predecessor
* أصبحت جميع مؤشرات السوابق واللواحق صحيحة

7-12الاسترداد بعد الفشل

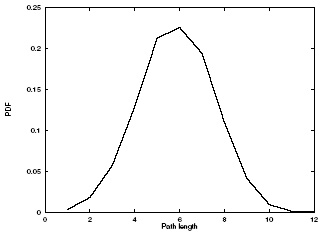
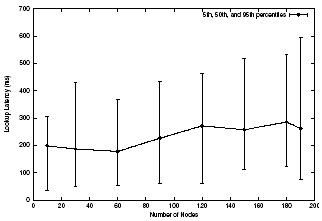
* الخطوة الرئيسية في عملية الاسترداد هي إبقاء مؤشرات اللواحق successor pointers صحيحة
* ولتحقيق ذلك، فإن كل عقدة تبقي قائمة لواحق successor-list فيها كل اللواحق القريبة منها
* إذا وجدت عقدة n أن اللاحق لها حصل له فشل، فإنها تستبدله بأقرب لاحق لها في القائمة
* عملية ا Stabilize تصحح مدخلات جدول الإصبع و مدخلات قائمة اللواحق successor-list التي تؤشر إلى العقدة التي حصل لها الفشل
* الأداء يتحسس بتكرار مرات انضمام ومغادرة العقد

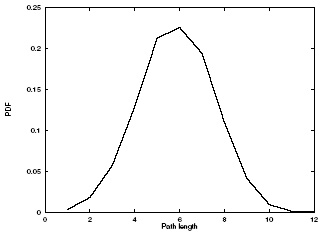
8- بروتوكول Chrod من وجهة نظر رياضية

* كل عقدة مسئولة عن حوالي K/N مفتاح ( حيث N عدد العقد و K عدد المفاتيح)
* عندما تنضم عقدة أو تغادر شبكة فيها n عقدة، فأنه فقط O(K/N) مفتاح سيتغير موقعه وليس جميع المفاتيح الموجودة
* عملية البحث تحتاج عدد رسائل يساوي O(log N).

9-نتائج التجارب

* التأخير Latency ينمو ببطء مع مجموع عدد العقد .
* طول مسار البحث تقريباً يساوي ½ log2N
* بروتوكول Chord يتميز بالمتانة والقوة أمام حالات الفشل في عدة عقد





المصادر والمراجع

http://www.mpi-inf.mpg.de/departments/d5/teaching/ws03\_04/p2p-data/11-18-writeup1.pdf\*

http://en.wikipedia.org/wiki/Chord\_(peer-to-peer)\*\*

\*\*\*http://pdos.csail.mit.edu/papers/chord:sigcomm01/chord\_sigcomm.pdf

1. تعريف وعمل : chord ويكيبيديا الموسوعة الحرة<http://en.wikipedia.org/wiki/Chord_(peer-to-peer)> [↑](#footnote-ref-2)