

مجلة جامعة البعث

سلسلة العلوم الهندسية الميكانيكية
والكهربائية والمعلوماتية



مجلة علمية محكمة دورية

المجلد 45 . العدد 19

1444 هـ - 2023 م

الأستاذ الدكتور عبد الباسط الخطيب
رئيس جامعة البعث
المدير المسؤول عن المجلة

رئيس هيئة التحرير	أ. د. محمود حديد
رئيس التحرير	أ. د. درغام سلوم

مديرة مكتب مجلة جامعة البعث
بشرى مصطفى

عضو هيئة التحرير	د. محمد هلال
عضو هيئة التحرير	د. فهد شريباتي
عضو هيئة التحرير	د. معن سلامة
عضو هيئة التحرير	د. جمال العلي
عضو هيئة التحرير	د. عباد كاسوحة
عضو هيئة التحرير	د. محمود عامر
عضو هيئة التحرير	د. أحمد الحسن
عضو هيئة التحرير	د. سونيا عطية
عضو هيئة التحرير	د. ريم ديب
عضو هيئة التحرير	د. حسن مشرقي
عضو هيئة التحرير	د. هيثم حسن
عضو هيئة التحرير	د. نزار عبشي

تهدف المجلة إلى نشر البحوث العلمية الأصيلة، ويمكن للراغبين في طلبها

الاتصال بالعنوان التالي:

رئيس تحرير مجلة جامعة البعث

سورية . حمص . جامعة البعث . الإدارة المركزية . ص . ب (77)

. هاتف / فاكس : 2138071 31 963 ++

. موقع الإنترنت : www.albaath-univ.edu.sy

. البريد الإلكتروني : [magazine@ albaath-univ.edu.sy](mailto:magazine@albaath-univ.edu.sy)

ISSN: 1022-467X

شروط النشر في مجلة جامعة البعث

الأوراق المطلوبة:

- 2 نسخة ورقية من البحث بدون اسم الباحث / الكلية / الجامعة) + CD / word من البحث منسق حسب شروط المجلة.
 - طابع بحث علمي + طابع نقابة معلمين.
 - إذا كان الباحث طالب دراسات عليا:
يجب إرفاق قرار تسجيل الدكتوراه / ماجستير + كتاب من الدكتور المشرف بموافقة على النشر في المجلة.
 - إذا كان الباحث عضو هيئة تدريسية:
يجب إرفاق قرار المجلس المختص بإنجاز البحث أو قرار قسم بالموافقة على اعتماده حسب الحال.
 - إذا كان الباحث عضو هيئة تدريسية من خارج جامعة البعث :
يجب إحضار كتاب من عمادة كليته تثبت أنه عضو بالهيئة التدريسية و على رأس عمله حتى تاريخه.
 - إذا كان الباحث عضواً في الهيئة الفنية :
يجب إرفاق كتاب يحدد فيه مكان و زمان إجراء البحث ، وما يثبت صفته وأنه على رأس عمله.
 - يتم ترتيب البحث على النحو الآتي بالنسبة لكليات (العلوم الطبية والهندسية والأساسية والتطبيقية):
عنوان البحث .. ملخص عربي و إنكليزي (كلمات مفتاحية في نهاية الملخصين).
- 1- مقدمة
 - 2- هدف البحث
 - 3- مواد وطرق البحث
 - 4- النتائج ومناقشتها .
 - 5- الاستنتاجات والتوصيات .
 - 6- المراجع.

- يتم ترتيب البحث على النحو الآتي بالنسبة لكليات (الآداب - الاقتصاد - التربية - الحقوق - السياحة - التربية الموسيقية وجميع العلوم الإنسانية):
- عنوان البحث .. ملخص عربي و إنكليزي (كلمات مفتاحية في نهاية الملخصين).
- 1. مقدمة.
- 2. مشكلة البحث وأهميته والجديد فيه.
- 3. أهداف البحث و أسئلته.
- 4. فرضيات البحث و حدوده.
- 5. مصطلحات البحث و تعريفاته الإجرائية.
- 6. الإطار النظري و الدراسات السابقة.
- 7. منهج البحث و إجراءاته.
- 8. عرض البحث و المناقشة والتحليل
- 9. نتائج البحث.
- 10. مقترحات البحث إن وجدت.
- 11. قائمة المصادر والمراجع.
- 7- يجب اعتماد الإعدادات الآتية أثناء طباعة البحث على الكمبيوتر:
 - أ- قياس الورق 25×17.5 B5.
 - ب- هوامش الصفحة: أعلى 2.54- أسفل 2.54 - يمين 2.5- يسار 2.5 سم
 - ت- رأس الصفحة 1.6 / تذييل الصفحة 1.8
 - ث- نوع الخط وقياسه: العنوان . Monotype Koufi قياس 20
- . كتابة النص Simplified Arabic قياس 13 عادي . العناوين الفرعية Simplified Arabic قياس 13 عريض.
- ج . يجب مراعاة أن يكون قياس الصور والجداول المدرجة في البحث لا يتعدى 12سم.
- 8- في حال عدم إجراء البحث وفقاً لما ورد أعلاه من إشارات فإن البحث سيهمل ولا يرد البحث إلى صاحبه.
- 9- تقديم أي بحث للنشر في المجلة يدل ضمناً على عدم نشره في أي مكان آخر، وفي حال قبول البحث للنشر في مجلة جامعة البعث يجب عدم نشره في أي مجلة أخرى.
- 10- الناشر غير مسؤول عن محتوى ما ينشر من مادة الموضوعات التي تنشر في المجلة

11- تكتب المراجع ضمن النص على الشكل التالي: [1] ثم رقم الصفحة ويفضل استخدام التهميش الإلكتروني المعمول به في نظام وورد WORD حيث يشير الرقم إلى رقم المرجع الوارد في قائمة المراجع.

تكتب جميع المراجع باللغة الانكليزية (الأحرف الرومانية) وفق التالي:
آ . إذا كان المرجع أجنبياً:

الكنية بالأحرف الكبيرة . الحرف الأول من الاسم تتبعه فاصلة . سنة النشر . وتتبعها معترضة (-) عنوان الكتاب ويوضع تحته خط وتتبعه نقطة . دار النشر وتتبعها فاصلة . الطبعة (ثانية . ثالثة) . بلد النشر وتتبعها فاصلة . عدد صفحات الكتاب وتتبعها نقطة .
وفيما يلي مثال على ذلك:

-MAVRODEANUS, R1986- Flame Spectroscopy. Willy, New York, 373p.

ب . إذا كان المرجع بحثاً منشوراً في مجلة باللغة الأجنبية:

. بعد الكنية والاسم وسنة النشر يضاف عنوان البحث وتتبعه فاصلة، اسم المجلد ويوضع تحته خط وتتبعه فاصلة . المجلد والعدد (كتابة مختزلة) وبعدها فاصلة . أرقام الصفحات الخاصة بالبحث ضمن المجلة.
مثال على ذلك:

BUSSE,E 1980 Organic Brain Diseases Clinical Psychiatry News ,
Vol. 4. 20 – 60

ج . إذا كان المرجع أو البحث منشوراً باللغة العربية فيجب تحويله إلى اللغة الإنكليزية و
التقيد

بالبنود (أ و ب) ويكتب في نهاية المراجع العربية: (المراجع In Arabic)

رسوم النشر في مجلة جامعة البعث

1. دفع رسم نشر (40000) ل.س أربعون ألف ليرة سورية عن كل بحث لكل باحث يريد نشره في مجلة جامعة البعث.
2. دفع رسم نشر (100000) ل.س مئة الف ليرة سورية عن كل بحث للباحثين من الجامعة الخاصة والافتراضية .
3. دفع رسم نشر (200) مئتا دولار أمريكي فقط للباحثين من خارج القطر العربي السوري .
4. دفع مبلغ (6000) ل.س ستة آلاف ليرة سورية رسم موافقة على النشر من كافة الباحثين.

المحتوى

الصفحة	اسم الباحث	اسم البحث
36-11	د. زينة لطفي	تحسين ضغط وتصنيف ملفات الفيديو باستخدام تقنية التعلم الآلي
56-37	د. ايلى خليل	دراسة خوارزميات التشفير المسبق وجدولة المستخدمين في الشبكات الفضائية متعددة الحزم ومتعددة المجموعات
74-57	م. غنوة حسين	دراسة تأثير عمق الحراثة والسرعة الامامية في بعض المؤشرات الاستثمارية للجرار ماسي فيرغسون 375 ذو الدفع الأمامي
94-75	م. نور قرقماز د. كمال السلوم	التاريخ الإشكالي للقيمة الخالية في جافا: نقاط الضعف في طرق الإصلاح حتى الآن
114-95	م. غنوة حسين	دراسة تأثير المتغيرات المختارة (سرعة دوران مضرب تفتيت السماد-شكل الشفرة) في أداء موزعات السماد العضوي
146-115	م. عمار العلي د. محسن عبود	معالجة مشكلة الملفات الصغيرة في هادوب

تحسين ضغط وتصنيف ملفات الفيديو باستخدام

تقنية التعلم الآلي

الدكتورة: زينة ابراهيم لطفي

قسم الاتصالات - معالجة الاشارة الرقمية

كلية الهندسة - جامعة الرشيد الخاصة

ملخص البحث

يعد ضغط وتصنيف الفيديو إجراءً ضرورياً لاختزال البيانات ضمن الفيديو، اعتمد الباحثون على طرق مختلفة لتحويل ملفات الفيديو كبيرة الحجم إلى ملفات فيديو أقل حجماً ولكن واجهوا بعض المشاكل بالمقابل من حيث التأثير على جودة الصورة ومعدل النقل للبيانات وبالتالي فك الضغط يؤدي إلى ضياع في المعلومات وعدم القدرة على تحقيق الضغط في الوقت الفعلي بسبب عدم توفر مجموعة قواعد بيانات كافية. تم في هذا البحث اقتراح تقنية جديدة تعتمد على شبكة عصبونية تلافيفية كثيفة مع مرشح ثلاثي الأبعاد لتنفيذ عمليات الطي للمراقبة والتنبؤ على التوالي، بحيث تكون النواه (Kernel) قادرة على الانزلاق في ثلاث اتجاهات (عرض الإطار وارتفاعه والزمن) وتنفيذ الاختزال والتصنيف في الوقت الفعلي وتم تدريب هذه الشبكة باستخدام مجموعة قواعد البيانات University of Central Florida (UCF101) والتي تحوي على مختلف أنواع الفيديو من حيث الحجم والدقة. وتم اختبار أداء هذه التقنية من خلال معامل الدقة و معامل الخسارة ومصفوفة الارتباك لمختلف أنواع عينات الاختبار من مجموعة قواعد البيانات UCF101.

الكلمات المفتاحية: ضغط الفيديو، تصنيف الفيديو، اختزال الفيديو، معامل الدقة، معامل الخسارة، مصفوفة الارتباك.

Improved compression and classification of video files using machine learning technology

Abstract:

Compressing and classifying video is a necessary measure to reduce the data within the video. Researchers relied on different methods to convert large-sized video files into smaller video files, but they faced some problems in terms of affecting image quality and bit rate, and thus decompression leads to loss of information and inability. Real-time compression is achieved because there is not enough database set available. In this research, a new technique based on a dense convolutional neural network with a three-dimensional filter is proposed to implement folding operations for monitoring and prediction respectively, so that the kernel is able to slide in three directions (width, frame height, and time) and implement real-time reduction and classification. Training this network using the University of Central Florida (UCF101) database set, which contains various types of video in terms of size and resolution. The performance of this technique was tested through the accuracy coefficient, the loss coefficient, and the confusion matrix for various types of test samples from the UCF101 database set.

Keywords: video compression, video classification, video downsampling, accuracy coefficient, loss coefficient, confusion matrix.

- مقدمة -

يهدف ضغط وتصنيف الفيديو إلى تقليل حجم الملفات الضخمة للفيديو وذلك بتقليل كمية البيانات الموجودة ضمن كل إطار في الفيديو لتخزينه ونقله عبر قناة الاتصال محدودة السعة وهذا ما هو رائج للاتصال بين الافراد عن طريق وسائل التواصل الاجتماعي والتخاطب عبر الايميل. يتم ضغط الفيديو عن طريق استخدام عدة تقنيات وخوارزميات منها تقليدي مثل MPEG4 ، H.264 [1]. ومنها يستخدم الشبكات العصبونية والتعلم العميق يعتمد بشكل رئيسي على التحويل المويجي وتقدير الحركة وحذف المتكرر منها ضمن الأطر، وترميز هوفمان حيث يعتمد ضغط الفيديو على تقليل عدد الإطارات في الثانية، وتحديد سمات الصورة (التباين والسطوع)، وإزالة الصور المتكررة. حيث يمكن أن تستهلك البيانات غير المضغوطة مساحة كبيرة، وهو أمر غير مناسب لمحدودية مساحة أقراص التخزين وسعة قناة الاتصال. وتختلف هذه التقنيات في خوارزميات الضغط المستخدمة وجودة الصورة المستعادة. ويتم ضغط الفيديو بشكل تلقائي في بعض المنصات مثل YouTube وVimeo ، حيث يتم تحويل الملفات إلى صيغ مضغوطة لتسهيل عملية النقل والتشغيل. حيث أن أجهزة المعالجة التي تستقبل هذه الصيغ أقل كلفة مادية وأقل تعقيد حسابي.

1- هناك نوعين من الضياعات تحدث عند ضغط ملف الفيديو:

1-1 - Lossless ضغط الفيديو بدون فقدان معلومات [2] : وهي تقنية تخفض عدد البتات بدون فقدان المعلومات عن طريق التعرف على بينات كل إطار في الفيديو وحساب التكرار الإحصائي، وبالتالي فلا يتم فقدان أي من معلومات عند الاستعادة، ويتم استخدامها لأرشفة وتخزين مقاطع الفيديو archiving and storing videos.

1-2 - Lossy ضغط الفيديو مع فقدان معلومات [2]: وهي تقنية تخفض عدد البتات عن طريق إزالة المعلومات غير الضرورية أو الأقل أهمية، ويتم استخدامها لنقل

ومشاركة مقاطع الفيديو streaming and sharing videos، و بعد فك الضغط عند استقبال الفيديو لا تستعيد المعلومات في شكلها الأصلي.

2- الخوارزميات التقليدية الشهيرة في ترميز الفيديو:

تهدف مجموعة خوارزميات الترميز السابقة إلى ضغط الفيديو حسب التقنيات مع مراعاة جعل محتوى الفيديو متوافقاً من حيث اللاحقة mp4 ، .mov ، .wmv . للتعامل مع العديد من الأجهزة والمنصات والشاشات. لذلك فإن أفضل خوارزميات الترميز المستخدمة لضغط الفيديو هي:

2-1 الخوارزمية H.264:

هي خوارزمية ترميز الفيديو الأكثر استخداماً حيث تتيح هذه التقنية تشغيل الفيديو بأي لاحقة كانت، ولا تضر بجودة المحتوى، وهي معروفة أيضاً باسم Advanced Video Coding (AVC) ترميز الفيديو المتقدم [3].

2-2 الخوارزمية (High Efficiency video coding, or HEVC):

تم تطوير خوارزمية H.264 لتنتج خوارزمية (High-Efficiency Video HEVC Coding لتحسين ضغط الفيديو من خلال تجزئة أطر الفيديو بتقنية كتل الترميز CTB Coding Tree Blocks ويتم التوقع لكل إطار تالي بناءً على بيانات مشفرة مسبقاً تم حفظها بعد كل تنبؤ ويستخدم الترميز الإنتروبي. وبالتالي الحصول على ملفات أصغر حجماً وبجودة أفضل [4].

3-2 الخوارزمية VP9:

طُورت من قبل Google وتؤمن خوارزمية VP9 كفاءة ضغط مماثلة عند مقارنتها بـ HEVC ولكن من أهم مزايا هذه التقنية أنها تتعامل بسهولة مع المنصات والجوالات ذات أنظمة التشغيل Android الراجحة حالياً ومتصفح Chrome . [5] YouTube

3- علم الترميز باستخدام أدوات الرؤية الحاسوبية:

تعتبر الرؤية الحاسوبية (computer vision) من أهم مجالات علم الحاسوب تهدف إلى بناء تطبيقات ذكية قادرة على فهم محتوى الصور كما يفهمها الإنسان. [6] حيث من الممكن أن تأخذ بيانات الصور عدة أشكال كالصور المتعاقبة (فيديو)، المشاهد من عدة كاميرات، بيانات مأخوذة من جهاز تصوير طبي.

ولنقل تقنيات ترميز الفيديو إلى مستوى أعلى تمت الاستفادة من علم الرؤية الحاسوبية نظراً لقدرتها على الكشف البصري بأدق التفاصيل حيث تحاكي أدواته الطريقة التي تتظر بها العين البشرية إلى العالم الخارجي مما يعمل هذا على تحسين الترميز وفقاً لنوع المحتوى أو المناطق الموجودة في الصورة حيث تركز العين البشرية. لذلك بدلاً من مجرد حذف الإطارات المتطابقة في التقنيات التقليدية، فإن الترميز المبني على الرؤية الحاسوبية يستهدف الأهم ويتجاهل الباقي من خلال من أهم الخوارزميات التي تعتمد على الشبكات العصبونية:

3-1 خوارزمية Huffman and Discrete Cosine Transform (DCT):

وهي خوارزمية هجينة تعتمد على تقنيتين لضغط الفيديو، تتمثل المرحلة الأولى من Discrete Cosine Transform (DCT) تحويل الجيب المتقطع في تحقيق ضغط عالٍ مع الحفاظ على جودة إعادة فك ضغط الفيديو. وتقنية Huffman والتي بدورها تزيد نسبة الضغط. الهدف الرئيسي من هذه الخوارزمية هو تحقيق مستويات ضغط مختلفة وعالية للصور والفيديو باستخدام الخوارزمية الهجينة [13].

3-2 خوارزمية Wavelet التحويل المويجي: وهي خوارزمية تستخدم لضغط الفيديو

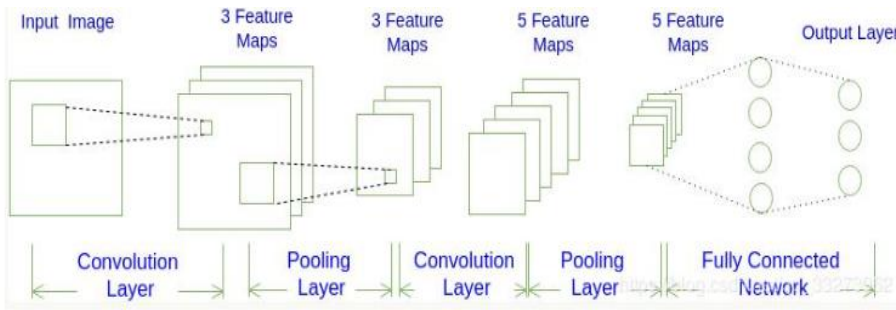
على نطاق واسع في العديد من التطبيقات وبشكل خاص المراقبة المستمرة التي تستهلك التخزين عن طريق النقاط الحركة، وبالتالي ضرورة ضغط الفيديو دون أي خسارة وفي

الوقت الفعلي. تظهر استخدام التحويل الموجي الأداء الأفضل أثناء المقارنة مع تحويل جيب المتقطع [15].

3-3 خوارزمية **Generative Adversarial Network GAN**: شبكة إعادة التوليد: وهي خوارزمية تستخدم لضغط الفيديو وإنشاء عمليات إعادة بناء من خلال أسلوب تنبؤ داخلي قائم على شبكة (GAN) لتعزيز دقة التنبؤ. مع قدرة فك الضغط غير الخطية الفائقة، حيث يعمل GAN على التنبؤ الصحيح عند فك الضغط [14].

4- بنية التقنية المقترحة:

تعتبر طبقات شبكات الطّي العصبونية هي طبقات تصغير الأبعاد ووظيفتها تصغير أبعاد الصورة، كما يبين الشكل (1) بنية الشبكة العصبونية التلافيفية التقليدية [7]. نستطيع الملاحظة بوضوح تغير أبعاد الصورة وبالتالي سمات هذه الصورة في شبكة الطّي الأمر الذي يجعل عملية تجميع سمات الصورة في هذه الحالة غير ممكناً نظراً للأحجام المختلفة للسمات.



الشكل (1) الشبكة العصبونية التلافيفية التقليدية. [7]

1-4 تم في هذا البحث اقتراح تقنية جديدة تعتمد على شبكة عصبونية تلافيفية كثيفة [8] كما هو موضح في الشكل (2) . ومرشح ثلاثي الأبعاد [9] كما هو موضح في الشكل (3) لتنفيذ عمليات الطي للمراقبة والتنبؤ على التوالي.

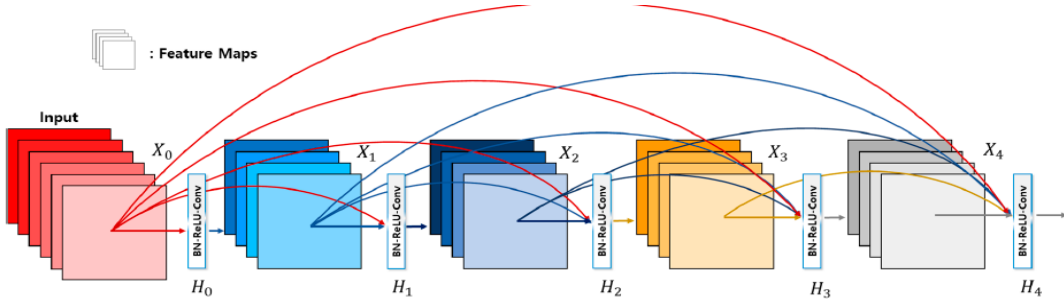
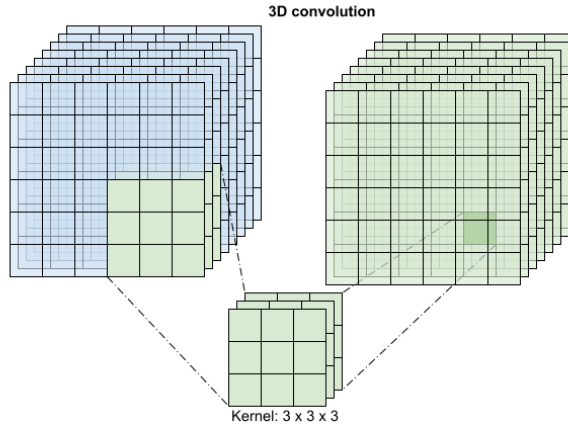


Figure 3. Architecture of five-layer densely connected convolution networks.

الشكل (2) الشبكة العصبونية التلافيفية الكثيفة [8]

2-4 يعمل النموذج المُقترح على حجم الفيديو (الزمن، الارتفاع، العرض) من أجل كل موقع خرج، حيث يكون ناتج الطي ثلاثي الأبعاد للحجم الأصلي هو متجه واحد في حجم الخرج، كما هو موضح في الشكل (3).



الشكل (3) ناتج الطي ثلاثي الأبعاد للحجم الأصلي هو متجه واحد في حجم الخرج [9]

3-4 يعتمد النموذج المقترح على نموذج التعلم العميق (ResNet) [10] وهو مستخدم لتطبيقات الرؤية الحاسوبية تم تطويرها عن الشبكة العصبونية تلافيفية (CNN) Convolutional Neural Network وهي مصممة لدعم مئات أو آلاف الطبقات التلافيفية. ويتم إعادة تجميع كل كتلة تم إجراء عملية الطي عليها وإجراء التدريب على مجموعة بيانات التدريب لقواعد البيانات وحساب معاملات الدقة (accuracy) وقيم الخسارة (loss) لبيانات التدريب. ثم اختبار النتائج بعد ضغطها أي الطي ثلاثي الأبعاد للحجم الأصلي من خلال مجموعة الاختبار لقواعد البيانات وإجراء عملية مقارنة مع الكتل الأصلية من خلال المعاملات دقة الكشف والتذكر عند الاستدعاء.

5- تدريب النموذج المقترح :

1-5 تم تدريب النموذج المقترح باستخدام مجموعة قواعد البيانات UCF101 لمدة 50 Epochs حقبة زمنية باستخدام Keras وهي عبارة عن واجهة برمجة لتطبيقات الشبكة العصبونية حيث تستخدم واجهات برمجية تسمح للمستخدم بصنع نماذج التعلم الآلي الخاصة به. واستغرق تدريب النموذج المقترح ساعة كاملة على جهاز حاسوب نوع انتل core i7 12 GB of RAM والاستعانة بإمكانيات الحوسبة السحابية Cloud Computing عن طريق تحميل البرنامج المكتوب بلغة البايثون على منصة ال (Google Colabs) ونلاحظ النتائج بعد التطبيق على Google Colab كما يلي:

```

Epoch 1/50
38/38 [=====] - 67s 2s/step -
loss: 2.69 - accuracy: 0.093 - val_loss: 2.40 -
val_accuracy: 0.10
Epoch 2/50
38/38 [=====] - 60s 2s/step -
loss: 2.28 - accuracy: 0.1600 - val_loss: 2.38 -
val_accuracy: 0.14
Epoch 3/50
.
.
.
Epoch 49/50
38/38 [=====] - 59s 2s/step -
loss: 0.54 - accuracy: 0.8167 - val_loss: 1.02 -
val_accuracy: 0.63
Epoch 50/50
38/38 [=====] - 59s 2s/step -
loss: 0.48 - accuracy: 0.8533 - val_loss: 1.07 -
val_accuracy: 0.64

```

2-5 مقارنة عملية التدريب باستخدام الخوارزميات DCT و WAVELET و

GAN والنموذج المقترح على مجموعة عينات التدريب في مجموعة قواعد البيانات UCF101 وحساب قيم الدقة (accuracy) وقيم الخسارة (loss) لبيانات التدريب عند استخدام كل خوارزمية على حدى.

$$Accuracy = \frac{True_{positive} + True_{negative}}{True_{positive} + True_{negative} + False_{positive} + False_{negative}}$$

(1)

-تشير نتائج الدقة (accuracy) التي تقيس بشكل مباشر عدد التنبؤات الصحيحة أو الخاطئة.. يمكن حساب الدقة [12] كما يلي:

-تشير نتائج الخسارة (loss) إلى القيم الحقيقية وتقارنها مع القيم المتنبئ بها. وتدل العلاقة التالية على معامل الخسارة المستخدم في التعلم الآلي هي الخسارة التربيعية:

$$\mathcal{L}_{sq} = \frac{1}{n} \sum_{i=1}^n (\tilde{y}_i - y_i)^2 \quad (2)$$

في هذه الصيغة، y_i هي النتيجة الصحيحة و \tilde{y}_i النتيجة المتوقعة. والقسمة على n ، للتعويض عن حجم مجموعة قواعد البيانات المستخدمة.

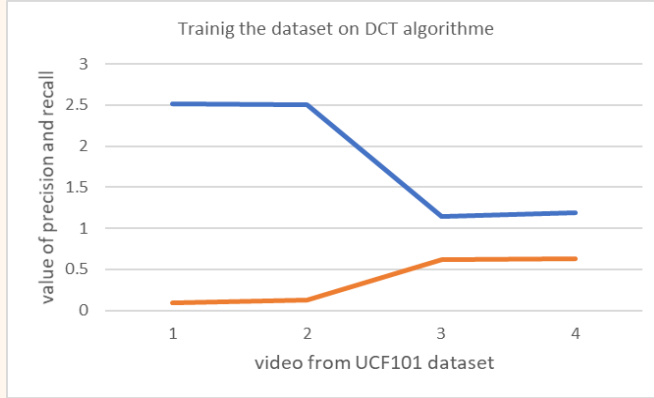
1-2-5 يبين الجدول (1) والمنحني البياني (1) قيم accuracy و loss لبيانات

التدريب في مجموعة قواعد البيانات باستخدام خوارزمية DCT .

الجدول (1) يبين قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات

لتقنية DCT

Traning	Loss	Accuracy
ApplyEyeMakeup	2.52	0.09
Archery	2.5	0.13
Basketball	1.14	0.62
BenchPress	1.19	0.63



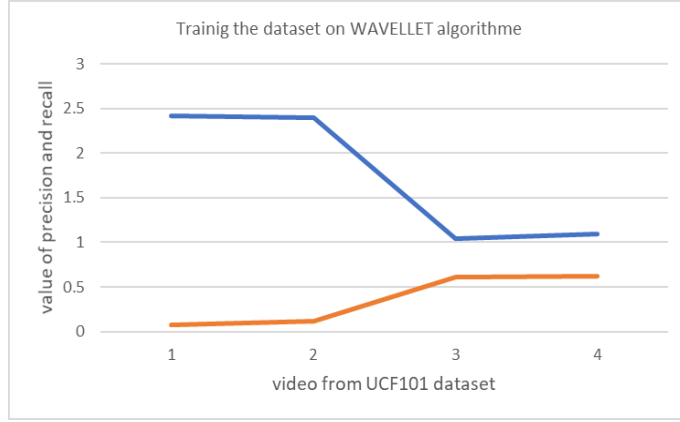
المنحني البياني (1) يبين قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات لخوارزمية DCT

2-2-5 - يبين الجدول (2) والمنحني البياني (2) قيم Accuracy و Loss لبيانات التدريب في مجموعة قواعد البيانات باستخدام خوارزمية wavelet في مجموعة قواعد البيانات .

الجدول (2) يبين قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات لتقنية

wavelet

Traning	Loss	Accuracy
ApplyEyeMakeup	2.42	0.079
Archery	2.4	0.119
Basketball	1.04	0.609
BenchPress	1.09	0.619



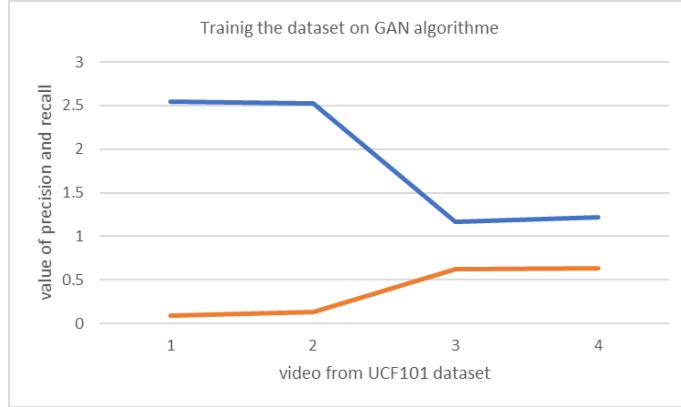
المنحني البياني (2) يبين قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات لخوارزمية wavelet

3-2-5 يبين الجدول (3) والمنحني البياني (3) قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات باستخدام خوارزمية GAN .

الجدول (3) يبين قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات

لتقنية GAN

Traning	Loss	Accuracy
ApplyEyeMakeup	2.55	0.09
Archery	2.53	0.13
Basketball	1.17	0.62
BenchPress	1.22	0.63



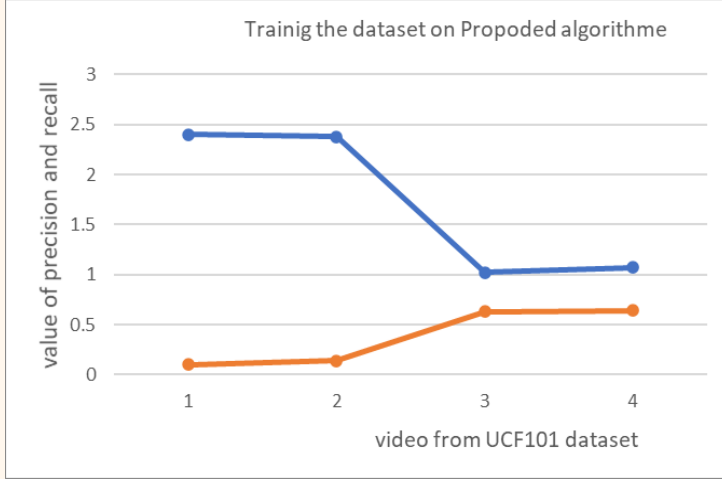
المنحني البياني (3) يبين قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات لخوارزمية GAN

4-2-5 - باستخدام النموذج المقترح في مجموعة قواعد البيانات UCF101. يبين الجدول (4) والمنحني البياني (4) قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات.

الجدول (4) يبين قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات

للمنحني المقترح

Traning	Loss	Accuracy
ApplyEyeMakeup	2.4	0.1
Archery	2.38	0.14
Basketball	1.02	0.63
BenchPress	1.07	0.64



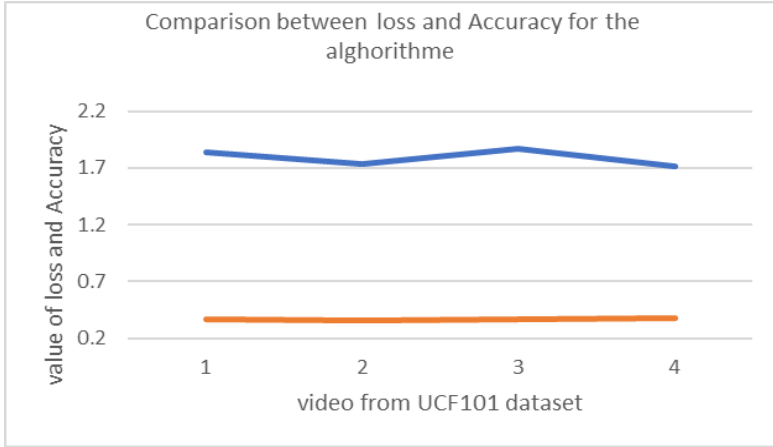
المنحني البياني (4) يبين قيم الدقة والخسارة لبيانات التدريب في مجموعة قواعد البيانات للنموذج المقترح

5-2-5- عند حساب المتوسط الحسابي لكل من خوارزمية من حيث loss و Accuracy كما هو مبين في الجدول (5). والمنحني البياني (5).

الجدول (5) المتوسط الحسابي للخوارزميات المستخدمة لعملية التدريب من حيث loss

و Accuracy

الخوارزمية المستخدمة للتدريب	Loss	Accuracy
DCT	1.84	0.37
Wavelet	1.74	0.35
GAN	1.86	0.36
Proposed	1.71	0.38



المنحني (5) المتوسط الحسابي للخوارزميات المستخدمة لعملية التدريب من حيث Accuracy و loss

6- اختبار النموذج والخوارزميات المستخدمة يتم حساب المعاملات:

-معامل التذكر عند الاستدعاء Recall : وهو النسبة بين عدد العينات الموجبة المصنفة بشكل صحيح على أنها موجبة إلى العدد الإجمالي للعينات الموجبة الصحيحة. تحدد قيمة ال Recall قدرة النموذج على اكتشاف العينات الإيجابية. وكلما زادت قيمته تم اكتشاف عينات أكثر إيجابية [12]. وفق العلاقة التالية :

$$\begin{aligned} \text{Recall} &= \text{True Positive} / \text{True Positive} + \text{False Negative} \\ &= \text{TP} / \text{TP} + \text{FN} \end{aligned} \quad (3)$$

-Precision وهي نسبة العينات الموجبة المصنفة بشكل صحيح (إيجابية حقيقية) إلى العدد الإجمالي للعينات الإيجابية المصنفة (سواء بشكل صحيح أو غير صحيح). تساعد الدقة في تصور موثوقية نموذج التعلم الآلي في تصنيف النموذج كاملاً على أنه إيجابي [11] وتعطى بالعلاقة التالية.

$$\begin{aligned} \text{Precision} &= \text{True Positive} / \text{True Positive} + \text{False Positive} \\ &= \text{TP} / \text{TP} + \text{FP} \end{aligned} \quad (4)$$

يعتبر معامل دقة الكشف Precision ومعامل التذكر عند الاستدعاء Recall هي مقاييس الأداء المستخدمة للتعرف على الأنماط والتصنيف في التعلم الآلي. بالاعتماد على المصطلحات الموجودة في المراجع [11] [12] تم تحديد معامل تقييم دقة الكشف Precision ومعامل تقييم التذكر عند الاستدعاء Recall على مستوى البكسل كما يلي:

حيث:

TP إيجابي حقيقي: عدد المرات التي يصنف فيها النموذج عينة إيجابية بشكل صحيح على أنها إيجابية.

FN سلبي كاذب: عدد المرات التي يصنف فيها النموذج بشكل غير صحيح عينة موجبة على أنها سلبية.

FP إيجابية خاطئة: عدد المرات التي يصنف فيها النموذج بشكل غير صحيح عينة سالبة على أنها إيجابية.

TN سلبي حقيقي: عدد المرات التي يصنف فيها النموذج عينة سالبة بشكل صحيح على أنها سالبة.

1-6 تم اختبار النموذج DCT و WAVELET و GAN و النموذج المقترح في google colab على فيديوهات اختبار من مجموعة قواعد البيانات UCF101 تم حساب Precision و Recall لمجموعة اختبار من مجموعة قواعد البيانات UCF101.

Precision –6

{'ApplyEyeMakeup': **0.42**

'ApplyLipstick': 1.0,

'Archery': **1.0**,

'BabyCrawling': 0.56

'BalanceBeam': 0.5,

'BandMarching': 0.66

'BaseballPitch': 0.9,

'Basketball': **0.83**

'BasketballDunk': 0.88

'BenchPress': **1.0**}

Recall

{'ApplyEyeMakeup': **0.9**,

'ApplyLipstick': 0.3,

'Archery': **0.6**,

'BabyCrawling': 0.9,

'BalanceBeam': 0.5,

'BandMarching': 0.8,

'BaseballPitch': 0.9,

'Basketball': **0.5**,

'BasketballDunk': 0.8,

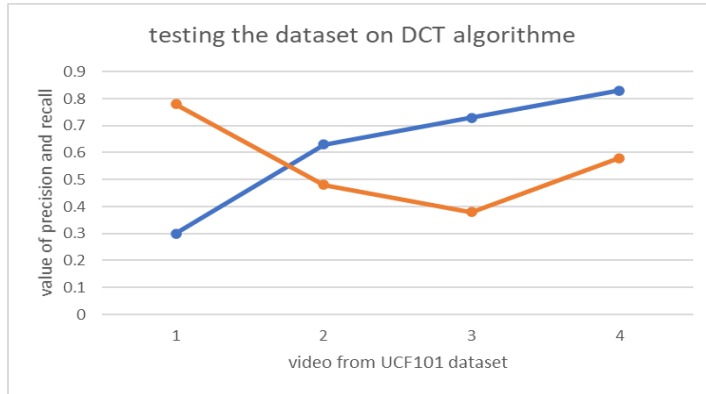
'BenchPress': **0.7**}

1-1-6 - يبين الجدول (6) والمنحني البياني (6) قيم Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام خوارزمية DCT .

الجدول (6) نتائج معامل Precision و Recall لبيانات الاختبار في مجموعة قواعد

البيانات بخوارزمية DCT

Video from UCF101	Precision	Recall
ApplyEyeMakeup	0.3	0.78
Archery	0.63	0.48
Basketball	0.73	0.38
BenchPress	0.83	0.58



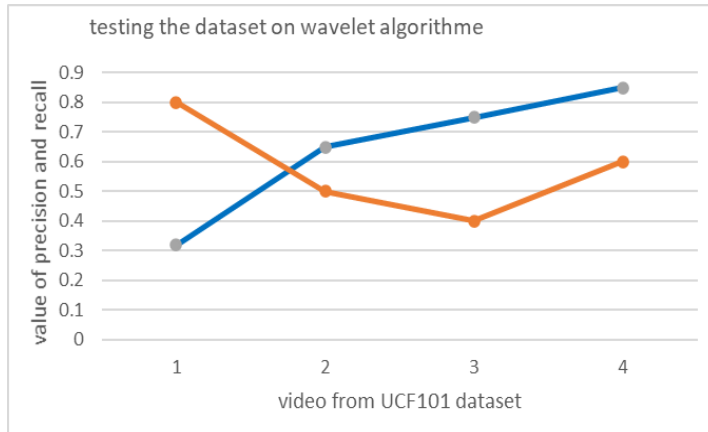
المنحني البياني (6) لقيم Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام DCT .

2-1-6- يبين الجدول (7) والمنحني البياني (7) قيم Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام خوارزمية Wavelet.

الجدول (7) نتائج معامل Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات

بخوارزمية Wavelet

Video from UCF101	Precision	Recall
ApplyEyeMakeup	0.32	0.8
Archery	0.65	0.5
Basketball	0.75	0.4
BenchPress	0.85	0.6

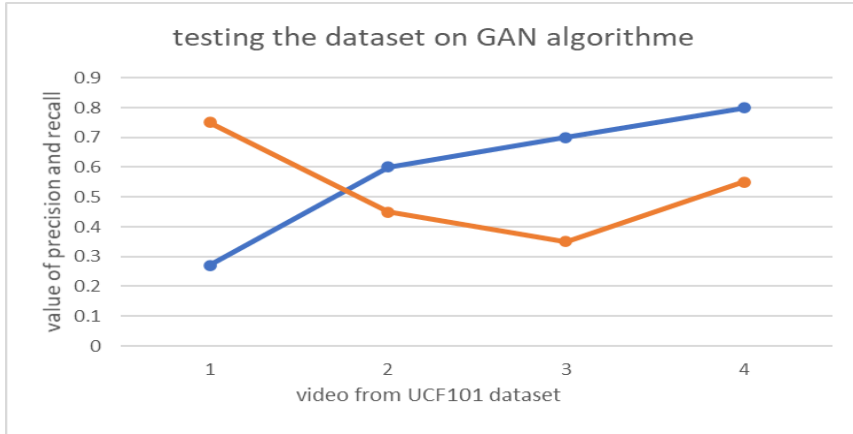


المنحني البياني (7) لقيم Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام Wavelet

6-1-3- يبين الجدول (8) والمنحني البياني (8) قيم Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام خوارزمية GAN.

الجدول (8) نتائج معامل Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات بخوارزمية GAN

Video from UCF101	Precision	Recall
ApplyEyeMakeup	0.27	0.75
Archery	0.6	0.45
Basketball	0.7	0.35
BenchPress	0.8	0.55

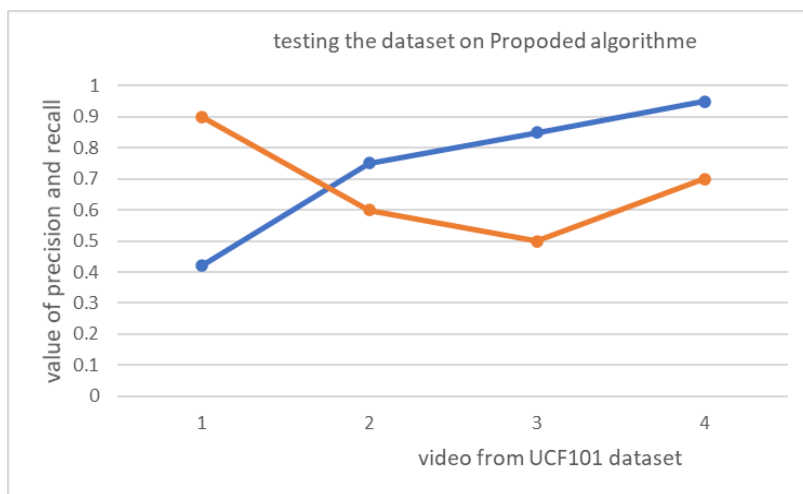


المنحني البياني (8) لقيم Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام GAN

4-1-6 يبين الجدول (9) والمنحني البياني (9) قيم Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام النموذج المقترح.

الجدول (9) نتائج معامل Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام النموذج المقترح

Video from UCF101	Precision	Recall
ApplyEyeMakeup	0.42	0.9
Archery	0.75	0.6
Basketball	0.85	0.5
BenchPress	0.95	0.7

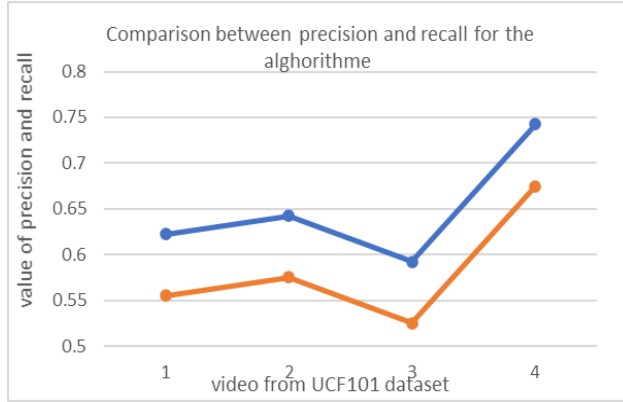


المنحني البياني (9) لقيم Precision و Recall لبيانات الاختبار في مجموعة قواعد البيانات باستخدام النموذج المقترح .

6-1-5- عند حساب المتوسط الحسابي لكل من خوارزمية من حيث Precision و Recall، كما هو مبين في الجدول (10).

الجدول (10). حساب المتوسط الحسابي لكل من خوارزمية من حيث Precision و Recall

الخوارزمية المستخدمة للاختبار	Precision	Recall
DCT	0.62	0.55
Wavelet	0.64	0.57
GAN	0.59	0.52
proposed	0.74	0.67



المنحني (10). حساب المتوسط الحسابي لكل من خوارزمية من حيث Precision و Recall

ويوضح الجدول نسبة التحسن في معاملات المستخدمة عند استخدام الخوارزمية المقترحة

الخوارزميات المستخدمة	Accuracy%	Precision%	Recall%
خوارزمية DCT	37	62	55
خوارزمية Wavelet	35	64	57
خوارزمية GAN	36	59	52
الخوارزمية المقترحة	38	74	67

النتائج:

1. تم تحسين معامل LOSS باستخدام الخوارزمية المقترحة بنسبة 6% .
بالمقارنة مع تقنية DCT.
2. كما نجد تحسن معامل Accuracy باستخدام الخوارزمية المقترحة بنسبة 35%
بالمقارنة مع تقنية DCT.
3. - تحسن معامل LOSS باستخدام الخوارزمية المقترحة بنسبة 2% بالمقارنة
مع تقنية Wavelet.
4. ونجد تحسن معامل Accuracy باستخدام الخوارزمية المقترحة بنسبة 37%
بالمقارنة مع تقنية Wavelet.
5. تحسن معامل LOSS باستخدام الخوارزمية المقترحة بنسبة 8% بالمقارنة مع
تقنية GAN.
6. تحسن معامل Accuracy باستخدام الخوارزمية المقترحة بنسبة 36% بالمقارنة
مع تقنية GAN.
7. تحسن معامل Precision باستخدام الخوارزمية المقترحة بنسبة 16% .
بالمقارنة مع تقنية DCT.
8. نجد تحسن معامل Recall باستخدام الخوارزمية المقترحة بنسبة 12% بالمقارنة
مع تقنية DCT.
9. - تحسن معامل Precision باستخدام الخوارزمية المقترحة بنسبة 13%
بالمقارنة مع تقنية Wavelet .
10. نجد تحسن معامل Recall باستخدام الخوارزمية المقترحة بنسبة 10%
بالمقارنة مع تقنية Wavelet.
11. تحسن معامل Precision باستخدام الخوارزمية المقترحة بنسبة 20%
بالمقارنة مع تقنية GAN.
12. نجد تحسن معامل Recall باستخدام الخوارزمية المقترحة بنسبة 15%
بالمقارنة مع تقنية GAN.

المراجع:

- [1] Valentim,J. Nunes , P. Pereira,F. 2002 , **Evaluating MPEG-4 video decoding complexity for an alternative video complexity verifier model.** IEEE Transactions on Circuits and Systems for Video Technology .Volume. 12. Issue. 11
- [2] KAVITHA,S. 2016, **A Survey on Lossless and Lossy Data Compression Methods.** International Journal of Computer Science & Engineering Technology (IJCSET). Vol. 7. No. 03.
- [3] Marpe.D and Thomas Wiegand.2006, **The H.264/MPEG4 Advanced Video Coding Standard and its Applications.** IEEE Communications Magazine. Volume: 44, Issue: 8. Page(s): 134 - 143.
- [4] Pali.V, Nandyal.S,2022. **Video Compression Using H.265 .2018,(HEVC-Main Profile).** International Journal of Applied Engineering Research, ISSN 0973-4562 Volume 17, Number 4 , pp. 427-435.
- [5] R`er`a`bek,M. Hanhart,P. Korshunov.P and Touradj Ebrahimi.2015. **Quality Evaluation of HEVC and VP9 Video Compression in Real-Time Applications.** IEEE Communications Magazine. 978-1-4799-8958-4/15.
- [6] Kaiming .H, Zhang. X; Ren. S and Jian Sun. 2016, **Deep Residual Learning for Image Recognition.** CVPR IEEE Xplor Conference on Computer Vision and Pattern Recognition. p. 770778.

- [7] Todt .E., Krinski. B. 2019, **Convolutional Neural Network - CNNVRI Group -Vision Robotic and Image**. Universidade Federal do Paraná.Federal University of Paran´a. DOI 10.1109/ICDIS.2018.00043.
- [8] Khah .S., Marteau. P.and Nicolas B´echet ,2018. **Intrusion detection in network systems through hybrid supervised and unsupervised machine learning process: a case study on the ISCX dataset**. IEEE International Conference on Data Intelligence and Security. DOI 10.1109/ICDIS..00043.
- [9] Sunkara .J, Navaneethasagari.E, Pradeep.D, Chaithanya.E, Pavani.D andSai Sudheer 2012. **A New Video Compression Method using DCT/DWT and SPIHT based on Accordion Representation**. Image Graphics and Signal Processing. DOI: 10.5815/ijigsp.2012.04.04.
- [10] Liang.J,2020. **Image classification based on RESNET**. Journal of Physics Conference Series 1634(1):012110 DOI:10.1088/1742-6596/1634/1/012110.
- [11] Tsai . V, 2006. **A Comparative Study on Shadow Compensation of Color Aerial Images in Invariant Color Models**. IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, VOL. 44, NO. 6.
- [12] Chung K L, Lin Y.R and Huang Y.H, 2009. **Efficient Shadow Detection of Color Aerial Images Based on Successive Thresholding Scheme** .IEEE TRANSACTIONS

ON GEOSCIENCE AND REMOTE SENSING, VOL. 47, NO. 2.

[13] Bharath K, Padmajadevi G, 2013. **Hybrid Compression Using DWT-DCT and Huffman Encoding Techniques for Biomedical Image and Video Applications.** IJCSMC, Vol. 2, Issue. 5, May.

[14] Yang R, Timofte R and Luc Van Gool, 2022. **Perceptual Learned Video Compression with Recurrent Conditional GAN.** IJCAI . 2109.03082.

[15] Gunanandhin S, Kalamani M, 2022 **Wavelet based Video Compression techniques for Industrial monitoring applications.** Journal of Physics, doi:10.1088/1742-6596/2272/1/012019.

دراسة خوارزميات التشفير المسبق وجدولة المستخدمين في الشبكات الفضائية متعددة الحزم ومتعددة المجموعات

الباحث: د. ايلي خليل

كلية: الهندسة المعلوماتية - الجامعة العربية الخاصة للعلوم والتكنولوجيا

الملخص

تلعب شبكات الأقمار الصناعية دورًا مركزيًا نحو تلبية متطلبات اتصالات الجيل الخامس 5G ، حيث لا يمكن تصور اتصال في أي وقت وفي أي مكان بدون وجود القمر الصناعي.

تعمل أنظمة الأقمار الصناعية من أجل تقديم حركة مرور بيانات تفاعلية ذات النطاق العريض على اعتماد نظام إشعاع متعدد الحزم ، مما يؤمن مزايا عديدة بالمقارنة مع إرسال حزمة عامة واحدة فأولاً ، استخدام مصفوفة هوائيات سيزيد نسبة الكسب إلى الضجيج مما يؤدي إلى زيادة معدل نقل البيانات لكل وصلة، ثانياً يمكن إرسال الرموز بنفس الوقت إلى مناطق مفصولة جغرافياً، مما يسمح بالاتصال متعدد الإرسال مكانياً، الى جانب أنه يسمح بإعادة استخدام النطاق الترددي المتاح في حزم منفصلة بما فيه الكفاية ، مما يؤدي إلى زيادة عرض النطاق الترددي للمستخدم مع الحفاظ على تداخل منخفض متعدد المستخدمين.

ندرس في هذا البحث خوارزميات التشفير المسبق (تشكيل الحزمة) المستخدمة لتأمين نظام متعدد الحزم الاشعاعية، ونقارن بينها ، كما ندرس أهمية جدولة المستخدمين ضمن الإطار الواحد

Studying the Recoding Algorithm and User Scheduling in Multibeam Multigroup Satellite Network

Anstract

SATELLITE communications will play a central role towards fulfilling next generation 5G communication requirements. As a matter of fact, anytime-anywhere connectivity cannot be conceived without the presence of the satellite segment. In order to deliver broadband interactive data traffic, satellite payloads are currently implementing a multibeam radiation pattern. The use of a multibeam architecture brings several advantages in front of a single global beam transmission [3]. First, since an array fed reflector is employed, the antenna gain to noise ratio can be increased leading to high gain of each beam return link achievable throughput. Second, different symbols can be simultaneously sent to geographically separated areas, allowing a spatially multiplexed communication.

In this paper, we study the pre-coding algorithms (beamforming) used to secure a multi-beam system, and compare them, as well as the importance of scheduling users within the same framework.

1- مقدمة:

إن الطلب المتزايد على الأقمار الصناعية ذات النطاق العريض ومعدلات نقل البيانات العالية، واعتماد الأقمار الصناعية كجزء من تقنية دعم انترنت الأشياء واتصالات جهاز الى جهاز M2M ، وتكامل شبكات الأقمار الصناعية مع شبكات الجيل الخامس ، دفع مقدمي خدمات الأقمار الصناعية نحو دراسة طرائق أكثر كفاءة لاستغلال عرض النطاق الترددي المتاح، ولهذا الغرض تستخدم أنظمة الأقمار الصناعية الجديدة وبشكل مشابه للشبكات الخلوية الأرضية حزم متعددة بدلاً من شعاع عالمي واحد في منطقة التغطية بحيث يتم الحصول على استخدام أعلى للطيف من خلال استخدام إعادة استخدام التردد الجزئي بين الحزم [1].

يؤمن اعتماد أنظمة الحزم المتعددة بعض الفوائد الإضافية، فأولاً يمكن لكل شعاع الحصول على كسب هوائي أكبر مقارنة بحالة الحزمة الواحدة ، كما أن الطيف المتاح في بنية الحزم المتعددة يمكن إعادة استخدامه مكانياً في الحزم المنفصلة ، و يمكن أن تدعم بنية الحزم المتعددة تعديلات مختلفة ومعدلات رمز لكل مستخدم حسب جودة رابط المستخدم (أي جودة الاتصال الساتل و المستخدم) كما يمكن تقديم الخدمة لعدة مستخدمين في وقت واحد في منطقة التغطية.

تكون بنية القمر الصناعي مجهزة بعدة مقسمات بحيث يتم إرسال المعلومات في نفس الوقت إلى حزم نقطية مختلفة على الأرض بنمط معين لإعادة استخدام التردد.

أن زيادة عدد الحزم سيجعل أداء النظام مقيداً بالمستوى المتزايد للتداخل بين المستخدمين بسبب الفصوص الجانبية للأشعة. يلزم لمعالجة هذه المشكلة الاعتماد تقنيات تشكيل الحزمة (Precoding) للساتل متعدد الحزم بهدف تخفيف التداخل بين الحزم من أجل تحسين الكفاءة الطيفية SE .

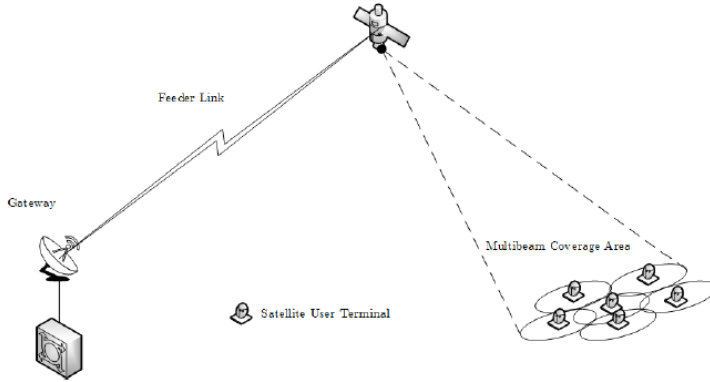
2- هدف البحث:

يهدف هذا البحث الى دراسة خوارزميات التشفير المسبق (تشكيل حزمة الاشعاع) في نظام اتصال فضائي متعدد الحزم ومتعدد المجموعات ومقارنة هذه الطرائق ودراسة أهمية جدولة المستخدمين ضمن الإطار الواحد وتأثير هذه الجدولة على خوارزميات التشفير المسبق .

3- نظام القمر الصناعي متعدد الحزم :

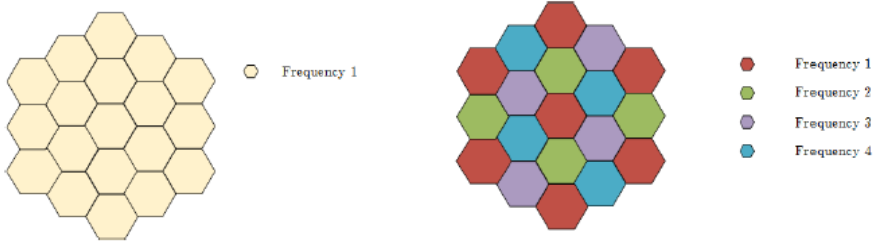
يوضح الشكل (1) بنية نظام اتصالات فضائية متعدد الحزم من طرف إلى طرف (End to End) من البوابة (GW) إلى طرفية المستخدم (UTs) ، حيث يمكن ملاحظة أن UTs تستقبل وتنقل المعلومات من خلال مخطط هوائي متعدد الحزم ، والذي يتم تغذيته بواسطة وصلة التغذية (Feeder link) . يجب أن يكون عرض النطاق الترددي المتاح في وصلة التغذية كبيرًا بما يكفي لدعم إعادة استخدام التردد المعتمدة للحزم المستخدمة. يسمح توليد عدة حزمة وارسالها الى الأرض بواسطة السائل بتدفقات معلومات مختلفة بمعدلات مختلفة ترسل نحو كل حزمة على حدة ، جنبًا إلى جنب مع التشفير والتشكيل التكيفي (ACM) لمطابقة ظروف القناة الأساسية.

إن إعادة استخدام التردد عبر الحزم النقطية المتعددة يولد تداخلًا في نفس القناة بين الحزم المجاورة. في الواقع ، نظرًا لعدم وجود طريقة لتأمين حزم معزولة تمامًا ، فقد تم تصميم تداخل مخطط بعناية بينها ، مما يؤدي إلى التحكم بنسبة اشارة الناقل الى التداخل.



الشكل (1) بنية نظام اتصالات فضائية متعدد الحزم

وبالتالي ، فإن إشارة الإرسال نحو حزمة معينة تُشع جزئياً إلى الحزم المجاورة المقابلة من خلال الفصوص الجانبية للهوائي. على الرغم من أن سويات الاستطاعة المستقبلية للحزم المجاورة ليست كبيرة مثل مستوى الحزمة نفسها ، إلا أن التداخل الناتج يتراكم ، وبالتالي تعاني روابط الاتصال من تدهور نسبة الإشارة إلى التداخل والضجيج (SINR) التي يمكن تحقيقها، و من أجل التعامل مع هذا التداخل داخل النظام ، يحاول مشغلو الأقمار الصناعية ومصنعوها فصل نطاقات التردد للحزم المجاورة بحيث يتم تخفيف التداخل، حيث يتم بشكل شائع تقسيم نطاق التردد إلى أربعة نطاقات فرعية بالنسبة لأنظمة الأقمار الصناعية عالية الإنتاجية (HTS) high throughput satellite ، بحيث تحتوي الحزم المجاورة الأقرب على نطاقات تردد منفصلة كما هو مبين في الشكل (2) و يتم تقديم درجة أخرى من الفصل بين الحزم الموضعية من خلال استخدام الاستقطاب المتعامد [2].



الشكل (2) اعادة استخدام التردد في شبكات الاقمار الصناعية

على الرغم من أن إدارة الطيف المقتصد هذه تخفف بشكل كافٍ من مشكلة التداخل متعدد الحزم ، إلا أنها تمنع توسيع سعة النظام باستخدام كل عرض النطاق المتاح في كل حزمة. لذلك ، لتحقيق قفزة هائلة في خدمات معدل البيانات المقدمة ، من المستحسن استخدام أكثر قوة للطيف. والجدير بالذكر أن هذه الرؤية قد تم تعزيزها أيضًا في النظام الخلوي عن طريق تحويل النموذج من خلية واحدة إلى أنظمة متعددة الخلايا [3].

ونتيجة لذلك ، فإن إعادة استخدام التردد العالي يجب أن تندمج جنبًا إلى جنب مع التقنيات المناسبة لتخفيف التداخل و هي الطريقة المناسبة للمضي قدمًا في هذا السيناريو .

نفترض لدينا نظام أقمار الصناعية واسع الحزمة الترددية يوفر خدمة للمستخدمين الثابت عبر حزم متعددة. تقوم مصفوفة الهوائيات بتحويل إشارات التغذية N إلى K شعاع مرسل.

تم جدولتها N مستخدم لكل حزمة في كل فترة زمنية، باستخدام التعدد عبر تقسيم الزمن (TDM) ، وتم اعتماد إعادة استخدام التردد بالكامل لتحسين الكفاءة الطيفية.

تعطى القناة الفضائية متعددة الحزم وفق [4] :

$$H = \varphi A \quad (1)$$

حيث φ تمثل تأثيرات تغير الطور بسبب مسارات انتشار مختلفة بين القمر الصناعي والمستخدمين

و A يمثل نموذج اشعاع مصفوفة الهوائيات متعددة الحزم.

نفترض أن بيئة الارسال هي يوجد خط نظر مباشر بين القمر الصناعي والمستخدمين (Line of site) وبالتالي و نظراً لأن تباعد هوائيات الأقمار الصناعية صغير نسبياً مقارنة بمسار الانتشار الطويل ، وبالتالي فرق الطور بين كل الهوائيات سيعد متطابقاً بالتالي ستكون ϕ عبارة عن مصفوفة قطرية مع قيم قطرية تعرف كما يلي:

$$\varphi_{i,i} = e^{j\varphi_i}, i = 1, \dots, K$$

وتعطى قيم A عند العمود ، والصف n كما يلي:

$$a_{K,n} = \frac{\sqrt{G_R G_{K,n}}}{4\pi \frac{d_K}{\lambda} \sqrt{K T_R B_w}}$$

حيث G_R هو ربح هوائي الاستقبال عند المستخدم و $G_{K,n}$ هو الربح بين الهوائي n و المستخدم K و d_K هي المسافة بين القمر الصناعي و المستخدم و T_R هي درجة حرارة الفضاء و K هو ثابت بولتزمان

تعطى الإشارة المستقبلية عند مدخل المستخدم وفق العلاقة:

$$y_k = h_k w_k x_k + \sum_{j \neq k} h_k w_j x_j + n_k \quad (2)$$

حيث W هي مصفوفة تشكيل الحزمة الـ Precoding Matrix .

و x_k هي البيانات المرسله للمستخدم K

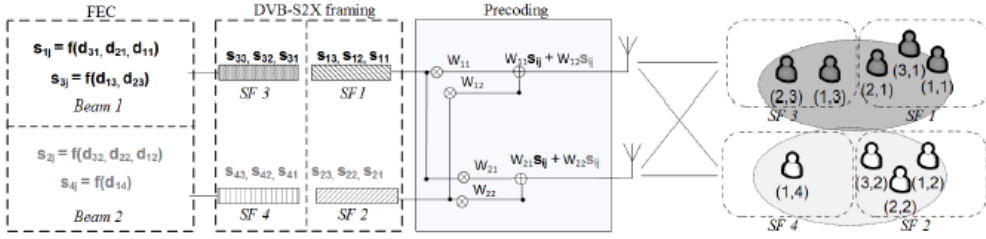
تعطى الاستطاعة المشعة من كل هوائي وفق العلاقة :

$$P_n = \sum_{K=1}^{N_t} w_k w_k^H \quad (3)$$

حيث N_T عدد هوائيات الارسال

يتم تشكيل الأطار المرسل بحيث يدعم عدة مستخدمين في نفس الوقت وترسل البيانات لمجموعة المستخدمين هذه معاً كما هو موضح في الشكل (3)

تم تحسين تصميم الطبقة المادية ليتوافق مع ضجيج محدود ، مع تأخيرات انتشار مفرطة و خفوت شديدة وهو ما يميز القناة فضائية في بروتوكول DVB-s2x لذلك اعتمد تشفير تصحيح الخطأ الأمامي الطويل Long FCE و تقنيات تخفيف الخفوت المعتمدة على تصميم طبقة الاتصال المتكيفة (Adaptive coding and modulation – ACM ، واعتماد تقنيات تشكيل الحزمة .



الشكل (3) تشكيل الاطار في بروتوكول DVB-s2X المكون من عدة مستخدمين وتجميعهم في مجموعات

(التشفير المسبق) لتأمين ارسال متعدد الحزم عبر استخدام خوارزميات متعددة سنبينها فيما يلي:

4- تقنية الإجبار الصفري (ZF) zero-forcing:

تقوم هذه الخوارزمية بزيادة نسبة الإشارة المستقبلة إلى التداخل، ويتم حساب العمود k في مصفوفة الكشف W كما يلي [5]:

$$w_k^{ZF} = \underset{w_k \in \mathcal{C}^{M \times 1}}{\operatorname{argmax}} \frac{\text{عاطتسا لخادتلا نيب نيمدختسلا}}{\text{فراشإلا قديفملا}} = \underset{w_k \in \mathcal{C}^{M \times 1}}{\operatorname{argmax}} \frac{\rho_u |w_k^H h_k|^2}{\rho_u \left| \sum_{j=1, j \neq k}^K w_k^H h_j \right|^2} \quad (4)$$

حيث ρ_u متوسط نسبة الإشارة إلى الضجيج SNR للوصلة الصاعدة

و h_k هي مصفوفة القناة للإشارة المفيدة و h_j هي مصفوفة القناة للمستخدم j

و لتحقيق العلاقة السابقة يتم اختيار العمود $\{w_k\}$ لمصفوفة فك الترميز كما يلي:

$$w_k h_k \neq 0 \quad . \quad w_k h_j = 0 \quad \forall j \neq k$$

و بمعنى آخر يتم اختيار أعمدة W ، بحيث يتم تخمين الإشارة المفيدة، وتكون متعامدة مع باقي اشارات المستخدمين في النظام. ويتم تحقيق هذا الشرط عن طريق إيجاد المعكوس الوهمي pseudo inverse لمصفوفة القناة H :

$$W^{ZF} = H(H^H H)^{-1} \quad (5)$$

حيث H^H هي التحويل الهرميتي لمصفوفة القناة H (وهي تعني المرافق العقدي للمنقول) وبالتالي نحصل على إشارة خالية من التداخل، ومن هنا جاءت التسمية؛ حيث أن هذه الخوارزمية تجبر الإشارات المتداخلة لتساوي الصفر.

في الواقع ، نظرًا لأن الحد الأقصى من الاستطاعة المستقبلية عند المستخدم k هو ρ ، فإن المخطط الأمثل هو أن يتم تسليم كل هذه الاستطاعة بواسطة الإشارة المطلوبة. في هذه الحالة يضاف العامل $\beta_{ZF} = \sqrt{\rho}$ لضبط الاستطاعة عند المرسل فتصبح قيم الأوزان بالشكل [6] :

$$W^{ZF} = \beta_{ZF} H(H^H H)^{-1} \quad (6)$$

تم تطوير هذه الطريقة بهدف التعامل مع تواجد عدد من المستخدمين في نفس الشعاع وذلك بالاعتماد على طريقة الفراغ الخالي (null-space) والتي تعتمد على استخدام تحليل القيمة المفرد SVD لمصفوفة القناة لكل المجموعات عدا المجموعة التي ينتمي لها المستخدم [6] .

5- خوارزمية الخطأ التربيعي المتوسط الأدنى MMSE:

تقوم هذه الخوارزمية بتكبير الإشارة المستقبلية بالنسبة لكل من الضجيج والتداخل SINR. ويتم ذلك من خلال تصغير مربع الخطأ المتوسط الأدنى بين الإشارة المخمنة الناتجة

عن الكاشف، وشعاع الرموز المرسله X . أي يتم إيجاد مصفوفة كاشف MMSE كما يلي [7] :

$$W^{MMSE} = \operatorname{argmin} \mathbb{E}\{\|W^H y - X\|^2\} \quad (7)$$

$$W^{MMSE} = \operatorname{argmin} \sum_{k=1}^K \mathbb{E}\{|w_k^H y - s_k|^2\}$$

$$W^{MMSE} = (HH^H + \frac{1}{\rho_u} I_K)^{-1} H$$

حيث $S = [s_1 \ s_2 \ \dots \ s_K]^T$ هو شعاع الرموز المرسله من K مستخدم

و I_K هي مصفوفة واحديه ذات K بعد

و ρ_u متوسط نسبة الإشارة إلى الضجيج SNR للوصلة الصاعده

يتم اضافة نفس العامل الذي أضيف في طريقة الأجرار الصفري β_{ZF} فتصبح العلاقة بالشكل [8] :

حيث I_K هي مصفوفة واحديه

$$W^{MMSE} = \beta_{ZF} (HH^H + \frac{1}{\rho_u} I_K)^{-1} H \quad (8)$$

6- خوارمية تكبير الـ SINR (SRM) :

نظرًا لأن سيناريو الإرسال المتعدد المجموعات يستلزم المرونة لتعظيم معدل النظام الإجمالي من خلال تقديم خدمة مختلفة المستويات بين المجموعات ، فالإرسال متعدد

المجموعات بأقصى ارسال بيانات ممكن يهدف إلى زيادة الحد الأدنى من SINR في كل مجموعة بالتوازي مع تعظيم مجموع المعدلات من كل المجموعات. يتم تحقيق ذلك عن طريق تقليل SINR للمستخدمين بظروف أفضل من أسوأ مستخدم من مجموعتهم. ويمكن كتابة الخوارزمية كما يلي [9] :

$$\begin{aligned} & \max \sum_{i=1}^{N_u} \log(1 + \gamma_i) \\ & \text{s.t. : } \min \gamma_i \\ & \text{and } \sum_{K=1}^{N_t} w_k w_k^H \leq P_n \end{aligned} \quad (9)$$

حيث γ_i هي نسبة الاشارة المفيدة الى اشارات التداخل والضجيج وتعطى بالعلاقة
:[10]

$$\gamma_i = \frac{|w_k^H \square_m|^2}{\sum_{l \neq k} |w_l^H \square_m|^2 + \sigma_m^2} \quad (10)$$

وهي مسألة أمثلية محدبة يتم حلها بالاعتماد على CVX Tool باستخدام الماتلاب

7- جدولة المستخدمين :

تغطي أنظمة الأقمار الصناعية متعددة الحزم عادةً مساحات شاسعة من خلال قمر صناعي واحد يتعامل مع مجموعة كبيرة من المستخدمين الراغبين في الخدمة. لذلك ، يعمل نظام الأقمار الصناعية في بيئة ذات عدد كبير من المستخدمين .

إن عنوانه أكثر من مستخدم في نفس الإطار يزيد من تعقيد الترميز حيث يجب مراعاة المواقع المكانية المختلفة يجب مراعاتها في معاملات التشفير الواحدة ، فالمستخدمين الذين ينتمون إلى نفس الشعاع سيكون لهم قنوات مختلفة ، مما يؤدي إلى ضعف أداء الترميز المسبق . ومع ذلك، يمكن لمصمم النظام اختيار المستخدمين بحيث يكون لديهم قنوات مماثلة . في ظل هذا السياق ، لا يجب على الجدول فقط اعتبار مستويات SINR لتجميع المستخدمين كما هو الحال في معيار [11] DVB-S2- حيث تتم جدولة المستخدمين بناءً على طلب الحركة وجودة القناة في جداول DVB - S2 متشابهة نسبياً من حيث مستخدم SINR في نفس الإطار - ولكن أيضاً أخذ شعاع القناة بعين الاعتبار

يعد اكتشاف أفضل المستخدمين الذين يتم تقديمهم في وقت معين على الفور مع الأخذ في الاعتبار أنه سيتم تشفيرهم مسبقاً بنفس أوزان الترميز المسبق مشكلة صعبة للغاية . والسبب هو أنه بعد تأثير التشفير المسبق ، تختلف مستويات SINR لكل مستخدم بحيث يصبح التخصيص المسبق للتشكيل والتشفير (modcod) قديماً . لذلك ، فإن التحسين هو عملية تكرارية تتطلب حساباً مستمراً .

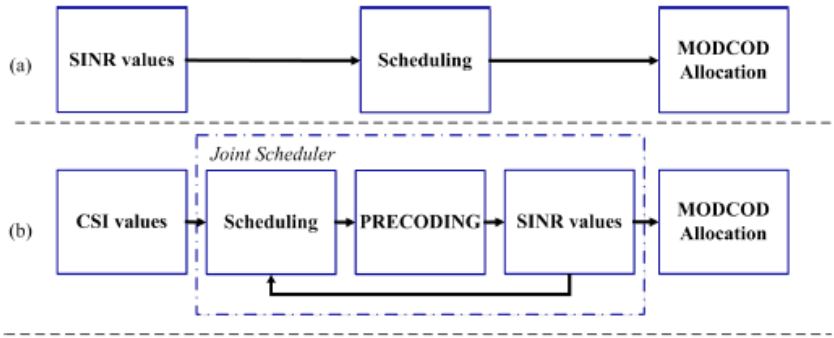
تغطي أنظمة الأقمار الصناعية متعددة الحزم عادةً مساحات شاسعة بواسطة قمر صناعي واحد يخدم مجموعة كبيرة من المستخدمين ، لذلك يعمل نظام الأقمار الصناعية في بيئة كبيرة متعددة المستخدمين .

تعتمد جدولة المستخدم في معايير الأقمار الصناعية الحالية على طلب الحركة وجودة القناة [3] ، وبالتالي فإن جداول DVB-S2 متشابهة نسبياً من حيث مستخدم SINR في نفس الإطار ويتم استخدام وضع طبقة ارتباط محدد (بافتراض adaptive coding and modulation - ACM) .

يوضح الشكل 4 (a) مخططاً للعمليات الضرورية التي يتم إجراؤها عند المرسل للأنظمة التقليدية.

يتمثل الاختلاف الأساسي مع الأنظمة التقليدية في أن CSI لكل مستخدم أصبح الآن منتجاً ذو N_t بعد بدلاً من قيمة SINR واحدة . (أي أن القناة CSI أصبحت مختلفة لكل مستخدم فبالنتيجة أصبحت مصفوفة مؤلفة من N_t قيمة بدلاً من أن تكون لمستخدم واحد) .

في لغة اتصالات MU -MIMO ، يمكن قياس مستوى التشابه بين المستخدمين من حيث تعامد قنوات المتجه المعقدة . لتعظيم التشابه بين متجهين ، نحتاج إلى تعظيم الـ Projection و الذي هو حاصل الضرب النقطي للمتجهين . على العكس من ذلك ، لتحقيق أقصى قدر من التعامد ، يجب تصغير الـ Projection .



الشكل (4) (a) الجدولة بالاعتماد على SINR فقط (b) الجدولة بالاعتماد على

SINR و شعاع القناة

فبالاستفادة من طبيعة الإرسال متعدد الحزم متعدد المجموعات ، تم تطوير سياسة جدولة المستخدم متعددة الطبقات في [9]، حيث نظرًا لأن جميع المستخدمين المجدولين المشتركين يتم خدمتهم من خلال الوضع الذي يفرضه أسوأ مستخدم في كل مجموعة ،

فسيتم تحقيق خسائر كبيرة في الأداء من منظور تصميم النظام من خلال مجموعة المستخدمين العشوائية هذه .مع التسليم بأن CSI متاحة بسهولة في جانب الإرسال ، حيث إنها شرط أساسي لتطبيق إدارة التداخل ، يمكن تحقيق أمثلية النظام بأي معنى مطلوب بواسطة طرق جدولة متقدمة .تعتمد هذه الطرق، كما هو موضح في الشكل 4 (b) ، إلى CSI الدقيق .يجب معالجة افتراضات CSI غير الكاملة في الامتدادات المستقبلية لهذا العمل [9].

8- النتائج:

سنجري الآن محاكاة لنظام شبكة فضائية متعددة الحزم ومتعددة المجموعات بالاعتماد على برنامج الماتلاب، يتألف النظام من 245 شعاع و تمت الدراسة من أجل التردد 20GHz وعرض حزمة لكل مستخدم 500MHz وارتفاع القمر الصناعي 35780 كم أي اعتمدنا نوع و ربح هوائيات الإرسال 50dBi و ربح هوائيات الاستقبال 30 dBi و استطاعة الإرسال الكلية $P_n = -40dbW$ ومعامل التموج (Roll off Factor) $a = 0.2$

سنجري عملية المحاكاة باستعمال خوارزميات التشفير المسبق الثلاث في كلا الحالتين مع اعتماد جدولة للمستخدمين وفي حالة عدم وجود جدولة للمستخدمين .

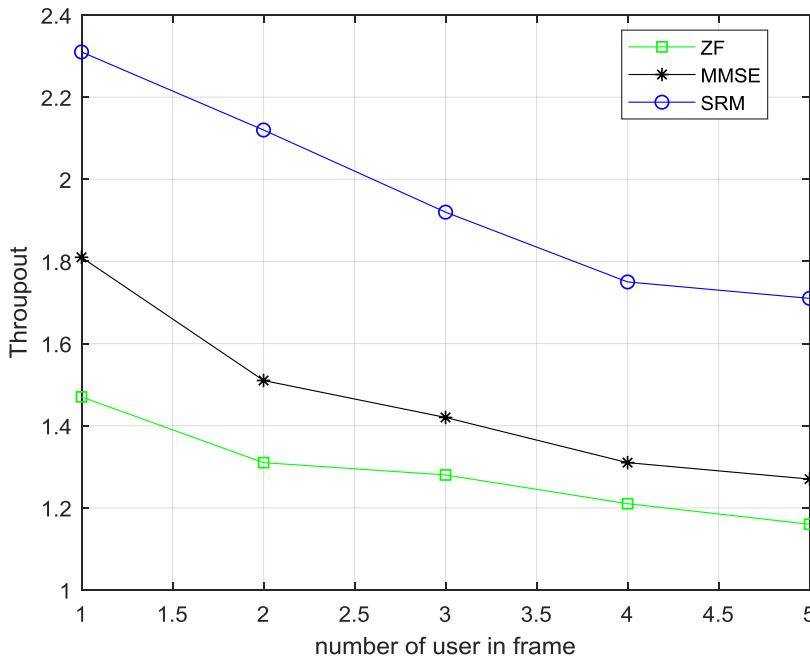
يبين الشكل معدل الإرسال الكلي في الشعاع الواحد مقابل عدد المستخدمين في نفس الإطار وذلك في حال عدم وجود جدولة للمستخدمين ضمن الإطار الواحد .

يعطى معدل نقل البيانات (Throughput) في معيار DVB-s2x وفق العلاقة [11] :

$$R = \min \frac{2B}{1+a} f_{DVB-s2x}(SINR_K) \quad (11)$$

حيث $f_{DVB-sex}(\cdot)$ هو تابع يعطي الكفاءة الطيفية وفق معيار DVB-S2X من أجل قيم SINR وهذا التابع متغير بحسب قيم SINR وحسب أنماط التعديل الرقمي المستخدم ويعطى مخطط يبين قيم التابع وفق كل حالة ويمكن الاطلاع على المخططات من [12]

نلاحظ من الشكل (4) أن طريقة SRM تقدم الأداء الأفضل بالمقارنة مع خوارزميتي ZF و MMSE وذلك في حالة عدم وجود جدولة للمستخدمين داخل الإطار

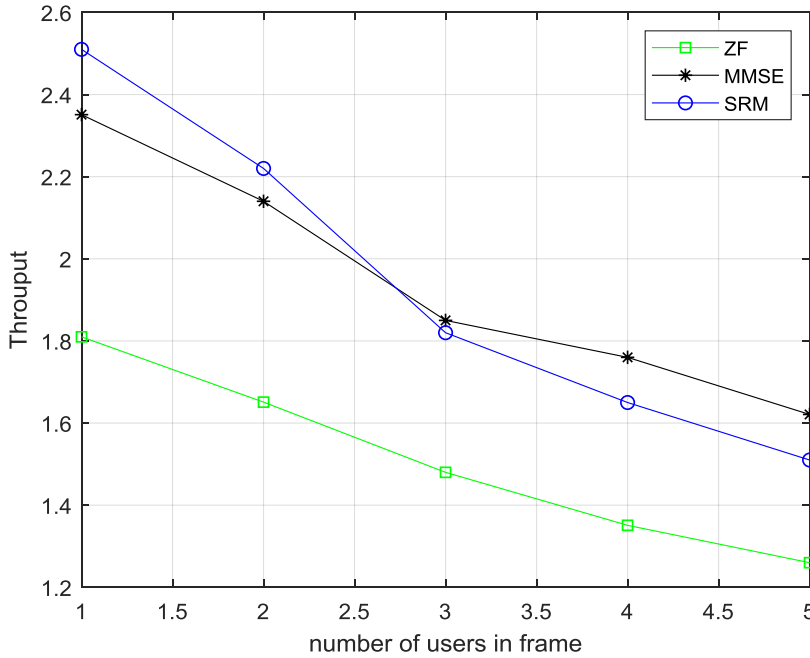


الشكل (4) معدل نقل البيانات الكلي مقابل عدد المستخدمين في إطار الواحد عند عدم وجود جدولة للمستخدمين

بينما يبين الشكل (5) معدل الارسال الكلي في الشعاع الواحد مقابل عدد المستخدمين في الاطار الواحد وذلك في حال وجود جدولة للمستخدمين ضمن الإطار الواحد .

ونلاحظ أولاً تحسن النتائج في كل الخوارزميات مع وجود جدولة للمستخدمين بالمقارنة عن الحالة السابقة عند عدم وجود جدولة.

كما نلاحظ أن طريقة SRM تقدم الأداء الأفضل من أجل عدد مستخدمين أقل من 3 في الأطار الواحد بينما قدمت طريقة MMSE الأداء الأفضل عند زيادة عدد المستخدمين أكثر من 3 في الإطار الواحد [9].



الشكل (5) معدل نقل البيانات الكلي مقابل عدد المستخدمين في إطار الواحد عند وجود جدولة للمستخدمين

9- المراجع:

[1] V. Joroughi, M. A. Vázquez, and A. I. Pérez-Neira, "Generalized multicast multibeam precoding for satellite communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 2, pp. 952–966, Feb. 2017.

[2] M. A. Vazquez *et al.*, "Precoding in Multibeam Satellite Communications: Present and Future Challenges," in *IEEE Wireless Communications*, vol. 23, no. 6, pp. 88-95, December 2016, doi: 10.1109/MWC.2016.1500047WC.

[3] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, Y. Wei, "Multi-Cell MIMO Cooperative Networks: A New Look at Interference," *IEEE J. Select. Areas*

[4] C. Qi and X. Wang, "Precoding Design for Energy Efficiency of Multibeam Satellite Communications," in *IEEE Communications Letters*, vol. 22, no. 9, pp. 1826-1829, Sept. 2018, doi: 10.1109/LCOMM.2018.2855935.

[5] Yong Soo Cho ; Jaekwon Kim ; Won Young Yang ; Chung G. Kang," MIMO-OFDM Wireless Communications with MATLAB", Wiley-IEEE Press 2010.

[6] Y. Silva and A. Klein, "Linear Transmit Beamforming Techniques for the Multigroup Multicast Scenario," *Vehicular Technology, IEEE Transactions on*, vol. 58, no. 8, pp. 4353–4367, Oct 2009.

[7] Mosa Ali Abu-Rgheff, "5G Physical Layer Technologies" Wiley-IEEE Press 2019

[8] G. Taricco, "Linear Precoding Methods for Multi-Beam Broadband Satellite Systems," in *European Wireless 2014; 20th European Wireless Conference; Proceedings of*, May 2014, pp. 1–6.

[9] D. Christopoulos, S. Chatzinotas, and B. Ottersten, "Multicast Multigroup Precoding and User Scheduling for Frame-Based Satellite Communications, *Wireless Communications, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[10] V. Joroughi, M. R. B. Shankar, S. Maleki, S. Chatzinotas, J. Grotz and B. Ottersten, "Robust Precoding Techniques for Multibeam Mobile Satellite Systems," *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, Marrakesh, Morocco, 2019, pp. 1-8, doi: 10.1109/WCNC.2019.8885862.

[11] ETSI EN 302 307-2 V1.1.1 (2014-10), “Digital video broadcasting (DVB); second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broad-ban satellite applications. Part 2: DVB-S2 Extension (DVB-S2X), european broadcasting union (EBU).”

[12] Hugo Meric , “Optimal DVB-S2 Spectral Efficiency with Hierarchical Modulation” , [arXiv:1411.5098v1](https://arxiv.org/abs/1411.5098v1) , Cornell University .

دراسة تأثير عمق الحراثة والسرعة الامامية في بعض المؤشرات الاستثمارية للجرار ماسي فيرغسون 375 ذو الدفع الأمامي

¹الباحثة: م. غنوة علي حسين

المخلص

أجريت تجربة حقلية في أحد الحقول الزراعية بمنطقة يحمور التابعة لمحافظة
طرطوس لحساب كل من استهلاك الوقود، الإنتاجية والانزلاق لوحدة الحراثة (جرار
ماسي فيرغسون 375 + محراث القرصي القلاب) في تربة طينية. استخدم تصميم
القطاعات العشوائية حيث مثلت الألواح الرئيسة أعماق الحراثة وبمستويين (15,25)
[سم]، بينما مثلت الألواح الثانوية السرعات الامامية للجرار وبثلاث مستويات
(2.66-3.82-5.71) [كم/سا] وبثلاثة تكررات وحللت النتائج احصائياً. وبينت ما
يلي: تفوق العمق (15) [سم] مقارنة بالعمق (25) [سم] وحقق التداخل بين العمق [15]
[سم] مع السرعة (5.71) [كم/سا] أعلى قيمة للإنتاجية العملية وبلغت
(1.119) [دونم/ساعة] وبأقل استهلاك للوقود (16.4) [لتر/هكتار] وحقق التداخل
بين العمق (15) [سم] والسرعة (2.66) [كم/سا] أقل قيمة للانزلاق [6.01%].

الكلمات المفتاحية: استهلاك الوقود ، أعماق وسرع الحراثة ، الانزلاق ، الإنتاجية.

¹ ماجستير - مكننة زراعية - كلية الهندسة التقنية - جامعة طرطوس.

A study of the effect of plowing depth and forward speed on some investment indicators of the front wheel drive Massey Ferguson 375 tractor

²Eng:Ghenwa ali hussen

Abstract

A field experiment was conducted in an agricultural field in Yamour region of Tartous governorate to calculate the fuel consumption, productivity and slippage of a tillage unit (Massey Ferguson 375 tractor + disc plow) in clay soil. The random plots design was used, where the main panels represented plowing depths with two levels (15.25) [cm], and the secondary panels represented the forward speed of the tractor with three levels (5.71-3.82-2.66) [km/h] with three replications, and the results were analyzed statistically. And it showed the following: The depth [(15) cm] was superior compared to the depth (25) [cm]. Dunum / hour [with the lowest fuel consumption [(16.4) liters / hectare], and the overlap between the depth [(15) cm] and the speed [(2.66) km / h] achieved the lowest (slip value. (6.01%

Keywords: fuel consumption, plowing depths and speeds, slippage, productivit.

² master- Agricultural Mechanization -Faculty of Technical Engineering

1- المقدمة

يعرف الجرار الزراعي بأنه المصدر الرئيسي والاساسي للقدرة اللازمة لتشغيل الآلات للقيام بالعمليات الزراعية المختلفة سواء كانت عمليات ثابتة (عمليات الري والدراس.) أو العمليات التي تتطلب التنقل من مكان لآخر أثناء أدائها (الحراث، البذر، العزيق....) (الرويشدي و آحرون،2015)

تتعدد أنواع الجرارات بتعدد الوظائف وقدرة محركاتها لكنها تشترك جميعها في الأجزاء الرئيسية: المحرك، الهيكل، القابض، صندوق السرعات، المحور الخلفي، عجلة التوجيه، جهاز النقل النهائي.

تتلخص الوظائف الأساسية للجرار: جر أو سحب الآلات الزراعية، قطر المقطورات الزراعية لنقل العمال والمحاصيل والاسمدة، إدارة الآلات الزراعية مثل مضخات الري وآلات الاعلاف، نقل الحركة الى الآلات الزراعية مع جريها بأن واحد، رفع أو خفض الآلات الزراعية عن طريق أجهزة ميكانيكية أو هيدروليكية (محمد وآخرون،2003)

هناك العديد من العوامل الزراعية المختلفة التي تؤثر على المواصفات الفنية للجرارات وخاصة استهلاك الوقود مثل نوع ونسيج التربة، رطوبة التربة، حجم الجرار، والعلاقة بين الجرار والآلة المستخدمة لذلك، قيم استهلاك الوقود للجرار مقاسة بطرق مختلفة ليست ثابتة بل تختلف من جرار لآخر (1993,Sorensen, Nielsen)

أكد (العاني، 2000) أن زيادة السرعة العملية للوحدة الميكانيكية (الجرار + المحراث) مع ثبات العمق قد رافقتها زيادة معنوية للانزلاق وقد ارجعوا السبب في ذلك إلى إن زيادة السرعة العملية قد أدت إلى زيادة مقاومة السحب وتقليل تماسك عجلات الجرار مع التربة ومن ثم زيادة قيمة الانزلاق.

توصل (الطائي، 1999) إلى أن زيادة السرعة الأمامية للوحدة الميكانيكية من 3.8 إلى 5.4 [كم/سا] سوف يؤدي إلى زيادة النسبة المئوية للانزلاق من 7.7% إلى 9.3% وقد

عزى السبب في ذلك إلى أن زيادة السرعة الأمامية سوف تؤدي إلى زيادة قيمة الانزلاق، وذلك بسبب الضياعات المرنة في الإطارات المطاطية، وتشوهات التربة عند سطح التماس.

أكد (Mathes et al, 1988) و(أمين وآخرون، 1992) و(الجراح، 1998) بأن زيادة السرعة العملية للجرار تؤدي إلى خفض معدل استهلاك الوقود لوحدة المساحة [لتر/هكتار] وقد أرجعوا السبب في ذلك إلى أن زيادة السرعة العملية تعني استغلال قدرة الجرار بشكل أمثل وتقليل الزمن اللازم لإنجاز عملية الحراثة لوحدة المساحة ومن ثم انخفاض كمية الوقود المستهلكة.

استنتج (Bukhari et al, 1990) بأن العوامل المؤثرة في استهلاك الوقود هي السرعة العملية، حيث إن زيادتها يسبب انخفاض قيمة التماسك بين العجلات وسطح الاستناد مما أدى إلى زيادة قيمة للانزلاق.

أكد (الطحان، 1995) أن كمية الوقود المستهلكة تختلف باختلاف السرعة العملية للجرار، حيث استنتج أن استهلاك الوقود يقل بزيادة السرعة وعمق الحراثة، ويعود السبب في ذلك إلى عدم استغلال قدرة الجرار الاستغلال الأمثل عند العمل بالسرعة البطيئة، مما يؤدي إلى حدوث هدر في الطاقة، أما في السرعة العالية فإن هذه الطاقة تستغل بصورة أفضل وتؤدي إلى زيادة في الإنتاجية.

وجد (عبطان، 2000) أن زيادة عمق الحراثة من 15 إلى 25 [سم] قد زاد معنوياً في معدل استهلاك الوقود من 32.24 إلى 43.89 لتر/هكتار أي بنسبة زيادة مقدارها 36.14% وعزى سبب ذلك إلى أن زيادة عمق الحراثة أدت إلى زيادة في حجم التربة المثارة ومن ثم الحاجة إلى طاقة أكبر لإنجاز هذا العمل.

بين (الشريفي والجبوري، 2011) أن هناك تأثيراً معنوياً للسرعة العملية للجرار في مؤشر الاستهلاك النوعي للوقود [ل/هـ]. حيث أدت زيادة السرعة العملية للجرار من

2.028 إلى 3.210 ثم إلى 4.955 [كم/سا] إلى انخفاض بمعدل الاستهلاك النوعي للوقود من 4.824 إلى 2.467 ثم إلى 10.698 [لتر/هكتار] على التوالي، وعزي سبب ذلك إلى أن زيادة السرعة العملية للجرار تعني استغلال قدرة المحرك بشكل أمثل وبالتالي تقليل الزمن اللازم لإنجاز العمل، ومن ثم انخفاض كمية الوقود المستهلكة.

أكد (الرويشدي وجاسم، 2015) أن استهلاك الوقود في بعض الظروف يقل مع زيادة السرعة الأمامية للجرار، حيث أن زيادة السرعة من 0.38 إلى 0.85 [متر/ثانية] نتج عنه انخفاض في الوقود المستهلك من 13 إلى 9.49 [لتر/هكتار]، وقد أرجعوا السبب في ذلك إلى زيادة السرعة العملية تؤدي إلى استغلال قدرة الجرار بشكل أمثل، وتقليل الزمن اللازم لإنجاز عملية الحراثة، ومن ثم انخفاض كمية الوقود المستهلكة.

أكد كل من (الطائي، 1999) ، (الخفاجي، 2001) ان السرعة العملية تؤثر تأثيراً كبيراً في الإنتاجية العملية إذ لاحظوا وجود علاقة طردية بين السرعة والإنتاجية العملية. وجد (الصباغ، 1990) بأن الزيادة في السرعة العملية هي إحدى أهم مركبات الإنتاجية العملية لذلك عند زيادة السرعة تحدث زيادة في الإنتاجية العملية. وجد (العاني، 1995) بأنه هناك تأثير معنوي لأعماق الحراثة في الإنتاجية العملية لمعدات الحراثة؛ إذ إن هناك انخفاض في الإنتاجية العملية عند زيادة عمق الحراثة؛ والسبب في ذلك يعود إلى الزيادة في قيمة الانزلاق بزيادة العمق مما يؤدي إلى انخفاض الإنتاجية.

وجد (الجراح، 1998) أن هناك انخفاض في الإنتاجية عند زيادة عمق الحراثة والسبب في ذلك يعود إلى الزيادة في قيمة الانزلاق بزيادة العمق، مما يؤدي إلى انخفاض السرعة العملية والتي هي أحد عوامل الإنتاجية.

2- أهمية وأهداف البحث

- 1- دراسة تأثير السرعة العملية والعمق في بعض مؤشرات الأداء الحقلية لوحدة الحرثة والتي شملت (الانزلاق، استهلاك الوقود و الإنتاجية)
- 2- تحديد أفضل سرعة عملية وعمق حرثة واللذان يؤديان لتحسين المؤشرات الفنية لوحدة الحرثة.

3- مواد وطرائق البحث

أجريت تجربة حقلية في أحد الحقول الزراعية بمنطقة يحمور في تربة طينية (الطين%54- السلت %14-الرمل %32) وذات محتوى رطوبي (15-17)% تم إجراء التحاليل الكيميائية في محطة بحوث بيت كمونة التابعة لمركز البحوث العلمية الزراعية (لحساب استهلاك الوقود والانزلاق والإنتاجية للجرار مع المحراث القرصي القلاب الشكل (1).



الشكل(1): وحدة الحرثة قبل البدء بالتجربة.

دراسة تأثير عمق الحراثة والسرعة الامامية في بعض المؤشرات الاستثمارية للجرار ماسي فيرغسون 375 ذو الدفع الأمامي

3-1- مواصفات المحراث القرصي القلب

جدول (1) مواصفات المحراث القرصي القلب المستخدم في التجربة

الواحدة	القيمة	المؤشر	التسلسل
[سم]	63	قطر القرص	1
[سم]	9.5	تقعر القرص	2
[درجة]	25	زاوية الميل	3
[سم]	66.5	العرض الشغال	4
[درجة]	50	زاوية القرص	5
[كغ]	340	الوزن	6
عدد	3	عدد الابدان	7

3-2- مواصفات الجرار المستخدم

جدول (2) بعض المواصفات الفنية للجرار المستخدم في التجربة

الواحدة	القيمة	المؤشر	التسلسل
-	ماسي فيرغسون 375	نوع الجرار	1
عدد	4	عدد الأسطوانات	2
[حصان]	75	القدرة الحصانية للمحرك	3
عدد	8 (4عادية، 4 بطيئة)	عدد السرعات الأمامية	4
عدد	2 (1عادية ، 1 بطيئة)	عدد السرعات الخلفية	5
[دورة/دقيقة]	2200	عدد الدورات القصوى	6
[كغ]	3000	الوزن الكلي للجرار	7
-	ديزل	نوع المحرك	8
-	جيدة جداً	حالة الجرار	9

استعمل في تنفيذ التجربة طريقة القطاعات المنشقة-المنشقة (Split-Split Block Design) وفق تصميم القطاعات العشوائية الكاملة (RCBD). إذ اشتمل البحث على دراسة عاملين (السرعة والعمق) وبثلاث مكررات للمعاملة الواحدة، حيث مثلت الألواح الرئيسية أعماق الحراثة وكانت (15-25) [سم] على التوالي ، ومثلت الألواح الثانوية السرعة الأمامية وكانت (2.66 ، 3.82 ، 5.71) [كم/سا] وبثلاث مكررات.

3-3- حسابات التجربة:

أولاً: حساب السرعة النظرية والعملية للوحدة الميكانيكية:

تم الاختبار بتشغيل وحدة العمل (الجرار مع المحراث) في حقل التجربة دون حراثة والمحراث يكاد يلامس الأرض لحساب الزمن النظري ولمسافة 30 متراً، مع ترك مسافة 10 أمتار من بداية خط العمل للوصول إلى الاستقرار في سرعة الجرار، وحسبت السرعة النظرية للسرعات المنتخبة (أولى، ثانية، ثالثة) وكانت قيمتها للسرعات المختارة 2.66-3.82-5.71 [كم/سا]، على التوالي وتم حسابها وفق المعادلة الآتية (عزت ومحمد علي، 1979).

$$V_t = \frac{S_t}{T_t} \times 3.6$$

إذ إن:

S_t : المسافة [متر].

T_t : الزمن النظري [ثانية].

في حين تم تشغيل (الجرار مع المحراث) في حقل التجربة وحسب الأعماق المختارة والسرعة المختارة وحسب التصميم المستخدم لحساب الزمن العملي 30متر مع ملاحظة ترك مسافة 10 امتار من بداية خط العمل للوصول الى حالة الاستقرار في السرعة والعمق المطلوبين وبعد تسجيل البيانات المطلوبة أجريت عليها الدراسات المطلوبة.

$$V_p = \frac{S_p}{T_p} \times 3.6$$

إذ إن :

S_p : المسافة [متر].

T_p : الزمن الفعلي [ثانية].

ثانياً: حساب قيمة الانزلاق:

يتم حساب قيمة الانزلاق من خلال المعادلة :

$$\delta_p = \frac{V_t - V_p}{V_t}$$

إذ إن :

δ_p : قيمة الانزلاق [%].

V_T : السرعة النظرية [كم/ساعة].

V_p : السرعة العملية [كم/ساعة].

ثالثاً: استهلاك الوقود (QF) (Fuel Consumption):

تم استخدام المعادلة الآتية للحصول على كمية الوقود المصروفة لوحدة المساحة (الجراح
وآخرون، 1998).

$$FU.C = (Q_d / T_E) \times 3.6$$

إذ أن:

$FU.C$: كمية الوقود المستهلكة لوحدة الزمن [ل/سا].

Q_d : كمية الوقود المستهلكة خلال المعاملة [مل].

T_E : الزمن الفعلي خلال المعاملة [ثانية].

إنّ نتائج استهلاك الوقود (لتر/ساعة) تمثل كمية الوقود المستهلكة في أثناء القيام بعملية الحراثة فقط دون توقف. ولغرض حساب كمية الوقود المستهلكة في أثناء التوقفات والاستدارة يتم قسمة قيم استهلاك الوقود في المعادلة السابقة على (80%) أي بزيادة مقدارها (20%) (الجراح، 1988).

رابعاً: الإنتاجية العملية (Practical Productivity):

تعرف الإنتاجية العملية بأنها أقصى إنتاجية يحتمل الحصول عليها عندما تعمل الآلة على (100%) من الوقت بالسرعة المحددة لها وبكامل عرضها (الطحان، 1991). يعبر عنها أيضاً بالأداء الفعلي للآلة في الحقل خلال مدة زمنية محددة، وتقاس بوحدات مساحة مقسومة إلى وحدات زمن مثل (هكتار/ساعة). أكد (البناء، 1990) أن الإنتاجية العملية تكون أقل من الإنتاجية النظرية وارجع سبب ذلك إلى عدة أسباب أهمها شكل وطبيعة الحقل واختلاف طريقة التنفيذ والمقاومات الطارئة في الحقل فضلاً عن مهارة العامل القائم بالحقل.

تم حساب الإنتاجية العملية لكل نظام من أنظمة الحراثة ولكل سرعة عملية وللمكررات جميعها باستخدام المعادلة الآتية (الطحان وآخرون، 1991).

$$P_p = 0.4 \times W_p \times V_p \times f_f$$

إذ أن:

P_p = الإنتاجية العملية [دونم/سا].

W_p : العرض الشغال الفعلي [م].

f_f : معامل استغلال

V_p : السرعة العملية [كم/ساعة].

4-نتائج الاختبار

4-1- عند عمق حراثة 15سم.

يبين الجدول (3) متوسط الانزلاق واستهلاك الوقود والإنتاجية العملية عند عمق حراثة 15سم:

جدول (3) العلاقة بين متوسط قيمة الانزلاق واستهلاك الوقود والإنتاجية العملية عند عمق حراثة 15سم

المتوسط العام			الإنتاجية [دونم/ساعة]	استهلاك الوقود [لتر/هكتار]	الانزلاق[%]	التكرار	السرعة [كم/ساعة].
الإنتاجية	استهلاك الوقود	الانزلاق					
0.550	26.38	6.01	0.500	23.5	5.1	تكرار 1	2.66
			0.540	27.24	5.73	تكرار 2	
			0.610	28.4	7.2	تكرار 3	
0.772	20.22	8.1	0.690	15	6.4	تكرار 1	3.82
			0.816	18.1	7.6	تكرار 2	
			0.810	27.56	10.2	تكرار 3	
1.119	16.4	10.87	0.950	13.1	9.2	تكرار 1	5.71
			1.207	15.5	11.1	تكرار 2	
			1.200	20.6	12.31	تكرار 3	

4-2- عند عمق حراثة 25سم

يبين الجدول (4) متوسط الانزلاق واستهلاك الوقود والإنتاجية العملية عند عمق حراثة 25سم:

جدول (4) العلاقة بين متوسط الانزلاق واستهلاك الوقود والإنتاجية العملية عند عمق حراثة 25سم.

المتوسط العام			الإنتاجية [دونم/ساعة]	استهلاك الوقود [لتر/هكتار]	قيمة الانزلاق [%]	التكرار	السرعة [كم/ساعة]
الإنتاجية	استهلاك الوقود	الانزلاق					
0.541	29.09	7.62	0.473	25.4	5.4	تكرار 1	2.66
			0.550	30.2	8.3	تكرار 2	
			0.600	31.67	9.16	تكرار 3	
0.756	23.53	10.01	0.700	21.5	7.9	تكرار 1	3.82
			0.748	25.2	9.83	تكرار 2	
			0.820	23.89	12.3	تكرار 3	
1.043	18.41	12.8	0.810	16	10.2	تكرار 1	5.71
			0.976	18.03	11.11	تكرار 2	
			1.343	21.2	17.1	تكرار 3	

5- الدراسة الإحصائية للنتائج

استخدم في التحليل الإحصائي برنامج excel2016، لمقارنة الفروقات بين المتوسطات بحساب قيمة أقل فرق معنوي عند LSD عند مستوى دلالة 5%، كما درست العلاقة الارتباطية وقوتها بين بعض الصفات المدروسة.

5-1- تأثير سرعة العمل وعمق الحراثة في الصفات المدروسة

لتقييم مقدار استهلاك الوقود وقيمة الانزلاق والإنتاجية العملية ولمقارنة عمق الحراثة الأفضل عند عند السرعة المدروسة، أُجريت تجربة باستخدام تصميم العشوائية الكاملة بواسطة برنامج excel2016، ولأجل المقارنات البعدية وتحليل تباين التجربة أُجري اختبار LSD وفق العلاقة التالية [إسماعيل، 2021]

$$LSD = Tdfe0.05 \sqrt{\frac{2MSe}{m}}$$

أولاً- تأثير أعماق الحراثة في استهلاك الوقود ونسبة الانزلاق والإنتاجية العملية

جدول (5) حساب الفروقات المعنوية

أعماق الحراثة [سم]	قيمة الانزلاق [%]	استهلاك الوقود [لتر/هكتار]	الإنتاجية [دوم/ساعة]
15	8.32	21	0.814
25	10.14	23.67	0.780
LSD0.05	0.15	0.22	0.021

يلاحظ من الجدول (5) إلى أن قيمة الانزلاق، واستهلاك الوقود والإنتاجية العملية قد تأثر معنوياً بزيادة عمق الحراثة حيث تفوق عمق الحراثة 15 [سم] معنوياً على العمق 25 [سم] وحقق أقل قيمة لانزلاق العجلات 8.32% وأقل استهلاك للوقود 21 [لتر/هكتار] وأعلى إنتاجية عملية 0.814 [دونم/ساعة].

ثانياً-تأثير عمق الحراثة وسرعات العمل في استهلاك الوقود ونسبة الانزلاق

جدول (6) حساب الفروقات المعنوية

العمق [سم]	السرعة [كم/ساعة]	نسبة الانزلاق [%]	استهلاك الوقود [لتر/هكتار]	الإنتاجية العملية [دونم/سا]
15	2.66	6.01	26.38	0.550
	3.82	8.1	20.22	0.772
	5.71	10.87	16.4	1.119
25	2.66	7.62	29.09	0.541
	3.82	10.01	23.53	0.756
	5.71	12.8	18.41	1.043
LSD0.05				
		0.26	0.37	0.036

يوضح لنا الجدول (6) تفوق التداخل بين العمق 15 [سم] والسرعة الاولى 2.66 [كم/سا] معنوياً حيث سجلت أقل قيمة للانزلاق 6.01 [%] وتفوق التداخل بين العمق 15 [سم] مع السرعة 5.71 [كم/سا] مسجلا اقل كمية لاستهلاك الوقود 16.4 [كم/سا].

وتفوق التداخل بين العمق 15 [سم] مع السرعة 5.71 [كم/سا] وحقق أعلى قيمة
للإنتاجية 1.119 [دونم/ساعة]

ومن خلال الجداول (5+6) تبين إحصائياً أن العمق 15 [سم] قد حقق أقل قيمة
لاستهلاك الوقود وأقل قيمة للانزلاق وأعلى قيمة للإنتاجية العملية مقارنة مع العمق [25سم].

الاستنتاجات:

1- أدت زيادة السرعة العملية من 2.66 [كم/سا] إلى 3.82 [كم/سا] ثم إلى 5.71 [كم/سا] إلى زيادة كل من الانزلاق والإنتاجية وبفارق معنوي على مستوى 0.05 وإلى انخفاض استهلاك الوقود.

2- أدت زيادة عمق الحراثة من 15 [سم] إلى 25 [سم] إلى زيادة كل من النسبة المئوية للانزلاق واستهلاك الوقود وبفارق معنوي على مستوى 0.05.

3- كان للتداخل بين عمق الحراثة والسرعة العملية دور هام في التأثير على المؤشرات المدروسة ، حيث تفوق التداخل بين العمق 15 [سم] والسرعة 2.66 [كم/سا] معنوياً على بقية المعاملات معطياً أقل قيمة للانزلاق، في حين تفوق التداخل بين العمق 15 [سم] والسرعة 5.71 [كم/سا] على بقية التداخلات معطياً أقل قيمة لاستهلاك الوقود.

(1) التوصيات:

1- إجراء عملية الحراثة على الأعماق القليلة والسرعة العالية لما حققته هذه المعاملات أعلى قيمة للمؤشرات المدروسة.

2- إجراء دراسات أخرى بأعماق وسرعات مختلفة لوحدة الحراثة لدراسة مؤشرات فنية أخرى غير مدروسة للجرار

3- القيام بدراسة مماثلة ولكن لنوع آخر من الجرارات والمفاضلة بينهما.

المراجع References:

- 1- البنا، عزيز رمو. (1990). معدات تهيئة التربة. مديرية دار الكتب للطباعة والنشر، وزارة التعليم العالي والبحث العلمي، جامعة الموصل، ص 3-44.
- 2- الطائي، محمد سالم يونس. (1999). كفاءة أداء أنظمة الحراثة في إزالة الطبقة المحراثية تحت ظروف الزراعة الديمية. رسالة ماجستير، كلية الزراعة والغابات، جامعة الموصل.
- 3- العاني، فراس سالم خلف. (2000). أداء الجرار المسرف (DT-75) مع المحراث المطرحي الرباعي القلاب (134) وتأثير تداخلهما في بعض الصفات الفيزيائية للتربة. رسالة ماجستير، قسم المكننة الزراعية، كلية الزراعة، جامعة بغداد، العراق، ص 69.
- 4- الجراح، مثنى عبد المالك نوري. (1998). تحميل الساحبة بنوعين من المحارث وقياس المؤشرات الخاصة باستهلاك الوقود تحت ظروف الزراعة الديمية. رسالة ماجستير، قسم المكننة الزراعية، كلية الزراعة والغابات، جامعة الموصل، العراق.
- 5- الخفاجي، إياد جميل جبر. (2001). دراسة بعض المؤشرات الاستغلالية للجرار ماسي فيركسن (FM-399) مع المحراث تحت التربة وكفاءته في تحسين بعض الصفات الفيزيائية للتربة. رسالة ماجستير، قسم المكننة الزراعية، كلية الزراعة، جامعة بغداد.
- 6- الرويشدي، زينة؛ جاسم، عبد الرزاق. (2015). دراسة بعض المؤشرات الفنية والاقتصادية للوحدة الميكانيكية باستخدام سرع مختلفة. مجلة العلوم الزراعية العراقية. العدد (6)، المجلد (46)، ص 1060-1068.
- 7- الطحان، ياسين هاشم؛ مدحت عبد الله حميدة؛ ومحمد قدرى عبد الوهاب. (1991). اقتصاديات وإدارة المكنائن والآلات الزراعية، وزارة التعليم العالي والبحث العلمي، جامعة الموصل، العراق.

- 8- الشريفي، صالح كاظم علوان؛ وموسى عبد شوجه الجبوري. (2011). دراسة تأثير نوع المحراث ورطوبة التربة والسرعَة العملية للوحدة المكنية في بعض مؤشرات الأداء والصفات الفيزيائية للتربة. المؤتمر العلمي الخامس لكلية الزراعة، مجلة تكريت للعلوم الزراعية، جامعة تكريت، ص 695-703.
- 9- الصباغ، عبد الرحمن أيوب. (1990). الساحبات ومعدات مكننة البساتين. قسم المكننة الزراعية، جامعة بغداد، وزارة التعليم العالي والبحث العلمي، العراق.
- 10- الطائي، محمد سالم يونس. (1999). كفاءة أداء أنظمة الحرّاة في إزالة الطبقة المحراثية تحت ظروف الزراعة الديمية. رسالة ماجستير، كلية الزراعة والغابات، جامعة الموصل.
- 11- الطحان، ياسين هاشم؛ سعد الدين محمد أمين. (1995). تأثير سرع الحرّاة في الأداء الحقلّي للمحراثين المطرحي والقرصي. مجلة زراعة الرافدين. ص 77-80.
- 12- العاني، رفعت نامق عبد الفتاح. (1995). دراسة تأثير السرعَة العملية العالية وأعماق مختلفة للحرّاة على بعض المؤشرات الاستغلالية للمحراث المطرحي القلاب مع الجرار عنتر (71) في منطقة أبي غريب. مجلة العلوم الزراعية العراقية. العدد (2)، المجلد (26)، ص 256-261.
- 13- عبطان، أحمد عبد علي. (2000). تأثير الحرّاة العميقة في إنتاجية بعض المحاصيل الحقلية. رسالة ماجستير، كلية الزراعة، جامعة بغداد.
- 14- عزت، عبد السلام محمود؛ ولطفي حسين محمد علي. (1979). الساحبات الزراعية، وزارة التعليم العالي والبحث العلمي، جامعة بغداد، العراق.
- 15- محمد، سيد عمران؛ عبد العال؛ زكي تايب. (2003). محاضرات في الجرارات والآلات الزراعية (دراسات عليا). قسم الهندسة الزراعية، كلية الزراعة، جامعة القاهرة
- 16- اسماعيل، ريم، (2021). مقرر الإحصاء وتصميم التجارب. جامعة طرطوس، كلية الهندسة التقنية.

المراجع الأجنبية:

1-Bukhari, S.C.1990 Effect of different speeds on the performance of mold board plow. Agri. Mech. In Asia, Africa and Latin America. Vol. 21, No . 1,, 21-24

2-Nielsen, V. and C.G.Sorensen, 1993. Technical fram Management a program for calculation of Work Requirement, workcapacitym work Budget. Danish Institute of Agricultural Engineering: Report No.53, 124p

3-Mathes, R.K.; W.F. Watson; I.W. Sarelle and D.L. Sirois. 1988 -17 *Effect of load and speed on fuel consumption of a Rubber-tired skidder*. Trans of ASAE. Vol. 31, N°. 1, 37-39

التاريخ الإشكالي للقيمة الخالية في جافا: نقاط الضعف في طرق الإصلاح حتى الآن

الأستاذ الدكتور كمال السلوم*
المهندسة نور قرقماز**

ملخص

تعاني لغة جافا من خطأ القيمة الخالية (null pointer exception)، إذ أن القيمة الخالية (null) هي واحدة من أكثر المفاهيم البرمجية المثيرة للجدل، فبعد أن تم خلقها للمساعدة في تهيئة القيم الجديدة الغير مألوفة للغة، تبين أن تكلفة صيانتها تفوق بكثير الجدوى من ابتكارها. وأصبحت بعض الفرق البرمجية الكبرى تكرر قدر كبيراً من جهودها لتلافي المشاكل الناتجة عن القيمة الخالية، باعتبار أن استخدامها غير الصحيح يسبب خطأ القيمة الخالية (null pointer exception) الذي يؤدي بدوره إلى انهيار النظام البرمجي ككل. إن موضوع القيمة الخالية ذو أهمية بالغة منذ ظهورها وحتى أيامنا هذه، ونستعرض في هذه المقالة أبرز طرق معالجتها في جافا، وأسباب عدم نجاح هذه الطرق في إصلاح المشكلة بشكل جذري.

الكلمات المفتاحية: القيمة الخالية، التتويجات، فواحص القيمة الخالية، مكدس المسك، الأكواد السابقة، التوثيق، أنماط إصلاح الأخطاء، احتمالية احتواء المؤشر على القيمة الخالية.

*أستاذ في قسم هندسة البرمجيات ونظم المعلومات - كلية الهندسة المعلوماتية - جامعة البعث - حمص - سورية

**معيدة موفدة داخلياً لصالح جامعة تشرين (طالبة دكتوراه) في قسم هندسة البرمجيات ونظم المعلومات - كلية الهندسة المعلوماتية - جامعة البعث - حمص - سورية

The Problematic History of the Null Value in Java: Drawbacks of Fixing Approaches Till Now

Prof. Dr. Kamal AlSalloum*

Eng. Nour Qarqamaz**

Abstract

Java suffers from the null pointer exception, that's because the null value is one of the most controversial concepts in programming, even though it was created to help configure new unfamiliar values, it turned out to be costly in maintenance. Many large programming teams put their efforts to fix the null problems, because careless null usage causes null pointer exception which then causes the software to crash. The null value has been a sensitive topic till this day, we showcase the ways to fix it in java, and the reasons why these ways are incapable of fixing it radically.

Key Words: Null, Annotations, Null Checks, Catch Stack, Legacy Code, Documentation, Bug Fix Patterns, Nullness.

*Full Professor, Department of Software Engineering and Information Systems, Faculty of Informatics Engineering, Al Baath University, Homs, Syria.

**Postgraduate (Doctorate), Department of Software Engineering and Information Systems, Faculty of Informatics Engineering, Al Baath University, Homs, Syria.

1. مقدمة:

إن معاناة مطوري لغة جافا مع خطأ القيمة الخالية ليست موضوعاً حديثاً، وقد تعددت المحاولات لمكافحة هذا الخطأ منذ وقت طويل، فقد اقترح FÄHNDRICH و LEINO في [1] حلاً للمشكلة وهو نظام نمطي (Type System) لتقسيم المؤشرات (references) بين المتحولات التي من الممكن أن تحوي القيمة الخالية (null) والمتحولات التي لا تحويها، ولكن كان على المطور أن يضع التوجيهات (annotations) بشكل يدوي، الأمر الذي لم يكن عملياً أبداً بما يخص الأكواد السابقة (Legacy Code).

قام SUMMERS و MÜLLER في [2] بتقييم عدة مخططات لتهيئة الأغراض (Object Initialization Schemas)، المخطط الأول هو "Raw Types" [1]، ويفرض هذا المخطط قيوداً على حقل ما بحيث عندما نخزن قيمة "non-null" فيه، فلا يمكن له أن يحمل القيمة "null" مجدداً، والذي لم يقدم أداءً جيداً. المخطط الثاني هو "Delayed Types" [3]، وهو يشبه تماماً مخطط "Raw Type" مع إمكانية تحديد الوقت الذي يصبح فيه حقل ما غير قابل لتخزين القيمة "null". المخطط الثالث هو "Masked Types" [4]، وفيه لا يمكن استخدام صف ما قبل أن نقوم بتهيئة كل عناصره وكل عناصر أبنائه.

اقترح SERVETTO et al. في [5] النمط "PlaceHolders and Placeholder Types" كبديل للقيمة الخالية، وذلك لتهيئة نمط معطيات من النوع الدائري (circular data structure) وهي نمط معطيات تكون آخر قيمة فيه مرتبطة بأول قيمة. وتكون حوامل المكان (PlaceHolders) بديلاً للقيمة "null"، وبدلاً من رمي استثناء "NPE" يتم رمي استثناء "PlaceholderException".

قدم CHALIN و JAMES في [6] دراسة موسعة يعترزون فيها بأن كل الحقول هي افتراضياً "non-null"، أي أن فواحص القيمة الخالية "nullness checker" تستخدم التتويه "@NonNull" بشكل افتراضي، ما عدا المواقع التي من الممكن أن تتغير تبعاً للتدفق (flow-sensitively refined)، كالمتحويلات المحلية (local variables) مثلاً. ولكن هذا ينطبق على الكود الذي كتبه المطور فقط (Checked Code) أي على المكونات الأكثر أهمية في كود المطور، ولا يشمل الكود المتعلق بالمكتبات المستخدمة مثلاً (Unchecked Code).

نقوم في هذا البحث بعرض أبرز الطرق المستخدمة لحل مشكلة خطأ القيمة الخالية، ونلقي الضوء على نقاط ضعفها أو الأماكن التي لا تكون فيها بنفس الفعالية في الأداء.

2. هدف البحث:

تخفيف المعاناة التي يسببها خطأ القيمة الخالية لمطوري لغة جافا، وترك المجال لهم للتركيز على أمور منطقية في الكود وضمان أن هذا الكود ينفذ المتطلبات الرئيسية المرجوة منه، إذ أن خطأ القيمة الخالية موضوع حيوي وقديم، وكثير من الفرق البرمجية حاولت تقديم حل جيد وفعال، ولكن نوضح في هذا البحث أن أبرز هذه الحلول لا تزال قاصرة عن حل المشكلة المطروحة بشكل كامل، ولذلك نسلط الضوء على نقاط الضعف الكائنة في هذه الحلول بهدف التركيز عليها وتلافيها في الحلول المستقبلية.

3. مواد وطرق البحث:

نقوم باستعراض طرق إصلاح خطأ القيمة الخالية (Null Pointer Exception) على مجموعة من أكواد جافا مفتوحة المصدر.

4. أساليب حل خطأ القيمة الخالية:

أولاً: بدأت أوائل المحاولات لإصلاح خطأ القيمة الخالية بفكرة التتويهاات (annotations)، إذ قدّم Papi و Ernst في [7] عام 2007 تتويبه @NonNull للمرة الأولى. وفكرة التتويهاات هذه أصبحت مستخدمة على نطاق واسع فيما بعد لاكتشاف الأخطاء المرتبطة بـ "null". كما اقترح Loginov et al. في [8] عام 2008 تتويهاات مرتبطة بـ "null"، للتأكد من صحة وسلامة نتائج التحليل.

فضّل HUBERT في [13] استخدام تتويبه @NonNull على تتويبه @Nullable، إذ أن تتويبه @NonNull هو عبارة عن كلمة مفتاحية ترفق قبل نمط واسم الحقل الذي يراد به أن لا يحمل القيمة الخالية في أي مرحلة من حياته. مثال:

```
@NonNull Age age;
```

بطبيعة الحال لا يمكن لحقل العمر أن لا يحمل قيمة ما (على الأقل 0)، وتعتبر العبارة البرمجية السابقة عن ذلك بالضبط. ويتم فهمها من قبل المترجم (compiler) بأن age لا يقبل القيمة "null" أبداً، أي لا يتم إسنادها إليه بأي مرحلة من مراحل الكود بعد الانتهاء من تنفيذ الباني، وبالتالي نضمن عدم ظهور خطأ القيمة الخالية بسبب استخدام الحقل age.

ثانياً: ثاني أبرز الطرق لحل خطأ القيمة الخالية هي أنماط إصلاح الأخطاء (Bug-Fix Patterns) [9]، إذ يستند المؤلفون إلى فكرة أساسية هي أن الكشف المبكر عن الأخطاء البرمجية يلعب دوراً هاماً في التيقن من جودة البرمجيات قبل استخدامها (deployment)، ولذلك كان من الضروري معرفة سبب ظهور الأخطاء وسبل إصلاحها. ويعتقدون أيضاً بأن التنقيب (mining) عن الإصلاحات السابقة للأخطاء،

يمكن أن يقدم معرفة جيدة بشأن الأخطاء الجديدة التي يمكن أن تحدث، وإمكانية إصلاحها.

يرتكز عملهم على التتقيب في تاريخ الشيفرة (change history) الخاص بـ 717 مشروع مفتوح المصدر بلغة جافا (Java)، ليستخلصوا ما دعوه بأنماط إصلاح الأخطاء (bug-fix patterns). وتم التحقق اليدوي (manual inspection) من العديد من الأخطاء، وذلك لفهم السياق (context) الذي يحويها، والأسباب (reasons) وراء حدوثها!

فمثلاً، تم التوصل إلى أن غياب فواحص القيمة الخالية (missing null checks)، وغياب عملية التهيئة (missing configurations)، حالتان متكررتان جداً، وتؤديان إلى أخطاء (bugs) من الممكن العثور عليها وإصلاحها بشكل أوتوماتيكي.

ويتم عادةً التعامل مع الأبحاث المعنية بأنماط التغيير (change patterns) بالتصنيف اليدوي (manual categorization) أولاً، ومن ثم القيام أوتوماتيكياً بعد (automatic count) حالات (instances) هذه الأنماط المعرفة.

وتم الاعتماد على أنه لدينا كم هائل من البيانات (data) في تاريخ الأخطاء (change history) للعديد من المشاريع البرمجية. ويحوي هذا التاريخ، الكثير والكثير من إصلاحات الأخطاء (bug-fixes) المتعددة النوع (type) والحجم (size)، وإن التتقيب في هذه الإصلاحات بطريقة منظمة (systematic) يمكن أن يؤمن رؤيةً معمقةً للكيفية التي بها تنشأ هذه الأخطاء، ويتم إصلاحها! إذ أننا نعتقد أن 40% من الأخطاء متكررة [9]، وبالإمكان بالتالي تحديد مواقعها وإصلاحها أوتوماتيكياً.

يعتبر خطأ فقدان اختبار القيمة الخالية (missing null checks) واحداً من أكثر الأخطاء تكرراً، ويوسم هذا الخطأ بصفة الخطير إذ أنه السبب عادةً في استثناء القيمة الخالية (Null Pointer Exception) الذي يقود للانهيارات (crashes). كما أنه من الأسباب الأخرى التي تؤدي لأخطاء جسيمة، هي أن بعض الأغراض (objects) تحتاج لتهيئة فورية بعد إنشائها مباشرةً.

• الإجراءات المتبعة:

تقسم العملية المتبعة للحصول على أنماط إصلاح الأخطاء إلى ثلاث مراحل منفصلة [9]:

بناء القاعدة البرمجية (software corpora).

تحليل الشيفرة المصدرية (source code).

الفحص اليدوي (manual inspection).

1- تجميع المشاريع البرمجية بلغة جافا:

تقتضي المرحلة الأولى على تجميع القاعدة البرمجية الصحيحة، وهنا، هي عبارة عن مشروع مفتوح المصدر بلغة جافا من GitHub [9]، بمساعدة أداة تستعلم GitHub عن أكبر وأشهر مشاريع جافا المتوفرة، ومن ثم، عمل نسخة محلية من هذه المشاريع.

2- تحليل الشيفرة المصدرية وتاريخ التغييرات:

في المرحلة الثانية، نحلل [9] الشيفرة المصدرية وتاريخ التغييرات للمشاريع البرمجية السابقة كلها، حيث يجب أن يمر كل واحد من المشاريع السبعمئة وسبعة عشر المدروسين وتاريخ الأخطاء (change history) الخاص به بما يلي:

(a) إيجاد مواطن (commits) إصلاح الأخطاء: تم إعراب (parse) تاريخ الأخطاء، واستخراج التكرارات (revisions) التي تتوافق مع bug-fix معين. بكلمات أخرى، كل موطن (commit) يتوافق مع bug-fix إذا كانت رسالة الموطن (commit message) تحوي الكلمة المفتاحية "fix" أو أحد مشتقاتها (fixed, fixated, ... , etc).

(b) استخراج إصلاح الخطأ: تم استخراج نسختين من كل طريقة (method) تغيرت فعلاً، نسخة قبل الإصلاح (buggy method)، ونسخة بعده (fixed method).

(c) إجراء عملية مقارنة: بين الطريقة قبل وبعد الإصلاح لاستخراج الشيفرة التي تغيرت تماماً.

نسمي "bug hunk" أقل عدد من أسطر الشيفرة التي تحوي خطأ، ونسمي "fix hunk" أقل عدد من أسطر الشيفرة التي تم فيها إصلاح الخطأ. نصل بين كل "bug hunk" و "fix hunk" بالسلسلة المحرفية ">>>>", لنشكل مخطط تغيير الشيفرة (code transition layout) [9]، على سبيل المثال:

```
Callback.handle(row);>>>>callback.apply(row);
```

(d) إزالة الهوية (anonymization): نطبق هنا ثلاثة أمور:

نستبدل كل كلمة (word) (اسم متحول، اسم طريقة،...) بالحرف "T"، باستثناء كلمات جافا المفتاحية مثل null،int،return،....، ثم نستبدل كل رقم بالصفـر "0"، ونستبدل كل سلسلة من الفراغات والانتقالات (white space string) (tabs, new lines,) (spaces) بفراغ واحد (one space). وبهذا نحصل على الأنماط المجهولة (anonymized patterns)، مثال:

T.T(T);>>>>T.T(T);

3- الفحص اليدوي:

في المرحلة الثالثة، وبعد جمع كل المعلومات الضرورية، نقوم [9] بتجميع تغيرات الكود في مجموعات، حسب أنماط الإصلاح. ولذلك، ومن أجل كل نمط (fix pattern)، لدينا ما يلي:

- عدد مرات الظهور (occurrences).
- عدد المشاريع حيث ظهر النمط.
- مقاطع الكود الحرفية (concrete) التي تمثل هذا النمط.

ومن ثم، نقوم بترتيب الأنماط حسب عدد مرات ورودها في المشاريع ككل. بكلمات أخرى، كلما ازداد عدد مرات ظهور نمط ما في مشاريع مختلفة، كلما ازدادت أهميته (significance). ثم نبدأ بفحص الحالات (instances) لكل من الأنماط الأكثر أهمية، من خلال قراءة الـ "buggy code" الفعلي، وكيف تم إصلاحه، ورسالة موطن التنفيذ الموافقة للإصلاح، وكذلك تقارير الأخطاء (bug reports) كلما أمكن.

بعد كل ذلك، يبرز خطأ القيمة الخالية كواحد من أكثر الأخطاء وروداً وذلك بسبب غياب فواحص القيمة الخالية (Missing Null Checks)، ويكون الحل بالإضافة على الشيفرة (Addition changes)، حيث نضيف فاحص للقيمة الخالية (null check) على الغرض المفحوص (checked object)، مثل `if(T==null)` أو `if(T!=null)`.

→ `if(T!= null)`

تظهر هذه الأخطاء في 48% من المشاريع المفحوصة، وهذا يجعلها من المشاكل الجدية والخطيرة للغاية.

ثالثاً: يتم الاعتماد في هذه الطريقة لإصلاح خطأ القيمة الخالية على مكدمس المسك (Catch Stack) [10]، وهو عبارة عن مكدمس يحوي جميع الاستثناءات (Exceptions) المحتمل حدوثها في الكود، ويتم تعبئته من خلال جعل الكود ينفذ (Run) وملاحظة استثناء ما، عندها يتم تحديد موقع هذا الاستثناء في الكود وإضافته إلى المكدمس. فلمعرفة إذا كان استثناء معين سيتم مسكه في لحظة معينة أثناء التنفيذ، نحدد فيما إذا كان نمط (Type) هذا الاستثناء متوافقاً مع أحد الأنماط المخزنة في المكدمس.

يتم بناء هذا المكدمس عن طريق حقن (inject) مجموعة من التعليمات البرمجية على الكود الأصلي، وهي تتم بإضافة استدعاءات لطرق (method calls) وهي [10]:

```
catchStack.add();    catchStack.remove();
```

في بداية ونهاية جسم (body) كتلة `try-catch` في الكود، كما في الشكل (1).

```
// before transformation
try{
  // codeA
} catch (TypedException te){
  // codeB
}
// after NPEfix transformation
int tryId = catchStack.getFreeId();
try{
  catchStack.add(tryId, TypedException.class, AnotherTypedException
    .class);
  // codeA
} catch (TypedException te){
  catchStack.remove(tryId);
  // codeB
} finally {
  catchStack.remove(tryId);
}
```

الشكل (1): بناء مكس المسك [10]

وتكون آلية العمل على مرحلتين: الأولى (كشف مواقع حدوث الخطأ): بعد مطابقة استثناء ما مع أحد الاستثناءات المخزنة في مكس المسك، يتم تقييم المتحولات الموجودة في الكتل (طرائق، بواني، ...) التي تحوي هذا الاستثناء، بحيث يتم اختبار فيما إذا كانت قيمة هذا المتحول هي null أم لا، عن طريق مجموعة من الاختبارات التي تضاف على الكود الأصلي كما يوضح الشكل (2).

```
//before modification
o.doSomething();

// after NPEfix transformation
checkForNull(o).doSomething();

// with static method
public static Object checkForNull(Object o){
    if (o == null) // null dereference detected
        if (cannotCatchNPE())
            switch (STRATEGY) {
                case s1b: return getVar(currentMethod());
                ...
            }
    return o;
}
```

الشكل (2): تقييم قيم المتحولات بعد المطابقة مع مكدس المسك [10]

الثانية (إصلاح الأخطاء المكتشفة في المرحلة الأولى): إذا كان الاستثناء الناتج عن استخدام متحول ما معالماً من قبل المطور، فيتم تجاهله، أما عكس ذلك يتم إصلاح الخطأ بوحدة من الطرق التالية:

1) تخطي الاستخدام ككل.

2) استبدال القيمة الخالية للمتحول بقيمة أخرى هي أحد اثنين:

a. قيمة موجودة ومعروفة مسبقاً بحيث تكون متوافقة مع نمط المتحول المراد إصلاحه.

b. قيمة جديدة يتم خلقها مباشرةً: وذلك اعتماداً على نمط القيمة التي نحن بصدد استبدالها.

يتم تكرار كل ما سبق من أجل كل استثناء تم تخزينه في مكدس المسك حتى يتم إصلاحها جميعاً.

رابعاً: يتم الكشف عن الأخطاء بالاعتماد على حكمة الآخرين، إذ تبين الأبحاث السابقة أن محاولة الوصول إلى محتويات المؤشرات (dereferencing) المتعلقة بما ترده استدعاءات الطرائق، هي السبب الرئيسي لظهور خطأ القيمة الخالية [9]، بما أن المطورين لا يتوقعون بسهولة، أن طريقة ما سترد null!

ولزيادة الأمر سوءاً، وجدنا أن هناك توثيقاً قليلاً جداً للحالات، فيما إذا كانت طريقة ما ترد null فعلاً أم لا (nullness)، ونزعم بأن قابلية أن ترد طريقة ما null أم لا هو موضوع حيوي للغاية، ويقدم معلومات هامة للمطورين كي يتفادوا هذه الفئة من الأخطاء [11].

ندرس "method nullness" ونرى كم مرة تكون نتيجة طريقة ما مفحوصة ضد القيمة null، هي null فعلاً! وندعو هذه العملية "method nullability"، والتي يمكنها أن تفيد في مفهوم "method nullness" [11].

مفهوما "Method Nullness" و "Method Nullability":

تظهر الطرائق التي تعيد null كقيمة معادة، على أنها السبب الرئيسي وراء خطأ القيمة الخالية [9]، حيث أن 70% من القيم التي تحتاج أن تفحص بـ "null check" تعود من استدعاءات الطرق [12]. ويشير ذلك إلى أن المطورين هم غالباً غافلون عن "nullness" الطريقة التي يقومون باستدعائها. تدل "Method nullness" فيما إذا كانت الطريقة ترد null (nullable)، أو إذا كانت لا ترد null مطلقاً وبتاتاً (non-null). وقد يبدو من السهل بما يكفي للمطورين أن يتعاملوا بأريحية ومنطق مقبول مع "nullness" الطرائق في مشاريعهم، ولكن، تحليل الطرائق في كيانات برمجية خارجية، هو ليس أبداً بهذه السهولة!

وعندما يريد المطورون فحص القيم المعادة لطرق في كيانات خارجية، يواجهون عادةً ثلاث سيناريوهات [11]:

(1) يفترضون أن الطريقة لا ترد null، وبالتالي يتعاملون مع الخطأ "NPE" في مرحلة لاحقة، إن ورد!

بالرغم من أنه يمكن استخدام هذه التقنية أثناء عملية التطوير، حيث يتم اكتشاف أخطاء "NPE" وإصلاحها في مرحلة الاختبار (testing)، ولكن يبقى أن نضع بعين الاعتبار أن بعض هذه الأخطاء قد تنجح في الوصول إلى المنتج النهائي مسببةً انهيار النظام.

(2) يتبعون أسلوباً دفاعياً يقضي بإضافة "null check" دوماً، للتخلص من خطر ظهور خطأ القيمة الخالية.

تجعل هذه التقنية الشيفرة البرمجية تغص بالكثير والكثير من فواحص القيمة الخالية، الأمر الذي يؤدي لصعوبة كبيرة في فهم الكود وصيانته.

(3) يقومون بتفحص الشيفرة المصدرية وقراءة توثيق الطريقة (method documentation)، ومحاولة فهم ومنطقة القيمة التي تردها، ومن ثم، إضافة "null check" عند الحاجة.

ولكن، غالباً ما يكون التوثيق غائباً، وإنه من الصعب والمعقد أن نقوم بإعادة هندسة طرائق غير معروفة!

ونجمع النتائج التي حصلنا عليها لكل طريقة في المكتبة (library method)، في معيار (measure) معرف كما يلي [11]:

$$\text{Nullability}(\text{Method}) = \frac{\text{CheckedDereferenced}(\text{Method})}{\text{Dereferences}(\text{Method})}$$

يعني هذا (عدد المرات التي تم فيها استدعاء الطريقة مفحوصة، من أصل عدد مرات الاستدعاء دون فحص)، ويلعب هذا المعيار دور الوسيط لمفهوم "Method nullness"، ويعبر عن مقدار الثقة بأن طريقة معينة سترد null، حيث أن "nullability" قيمتها صفر تعني أن الطريقة لا ترد null أبداً، مثال، هذه الطريقة تستخدم في الكيانات المختلفة دوماً دون "null check"، وبالتالي، "nullness" هذه الطريقة هو "non-null". بشكل معاكس، تشير "nullability" بقيمة غير صفرية (non-zero) إلى أن الطريقة هي "nullable".

أي أننا نعتمد على الأسلوب الذي استخدم به المطورون طريقة (method) ما في ملفاتهم البرمجية، لنتمكن من استخدام نفس الطريقة في ملفاتنا نحن، أي نستفيد من حكمتهم وخبرتهم.

خامساً: قامت شركة "META" [14] بتطوير أداة تحليل ستاتيكية تدعى "Nullsafe"، والتي تقوم بتتبع خطأ القيمة الخالية في جافا عن طريق استخدام "Java Compiler API" التي تضيف مساراً إضافياً يقوم بتجميع أخطاء القيمة الخالية، ويستخدم نوعين من قواعد المعطيات في التحليل:

- Abstract Syntax Tree (AST): والتي تقدم الهيكلية اللغوية للكود.
- Control Flow Graph (CFG): والذي يقدم مخططاً تدقيقياً للكود.

وتقسم مراحل تنفيذ الأداة "Nullsafe" إلى:

- مرحلة الاستدلال عن النوع (Type Inference): وفيها يتم اكتشاف المواطن من الكود التي يتم فيها التعامل مع القيمة الخالية.

- مرحلة التحقق من النوع (Type Check): وفيها يتم التحقق فيما إذا كان الكود يقوم بعمل استخدام خاطئ للقيمة الخالية (كاستدعاء قيمة خالية، أو تمرير قيمة خالية كوسيط، ...إلخ).

5. النتائج ومناقشتها:

نوضح هنا الأسباب الكامنة وراء عدم فاعلية كل من الطرق السابقة في تقديم حل عملي ودقيق وشامل لمشكلة القيمة الخالية.

أولاً: من أكبر نقاط ضعف طريقة التتويجات في إصلاح خطأ القيمة الخالية هو أنها أولاً لا تراعي ملفات الأكواد السابقة (Legacy Code) التي لا تتبع بطبيعة الحال مفهوم التتويجات، وبذلك نكون قد استثنينا مقداراً مهولاً من أخطاء القيمة الخالية التي لا يمكن حلها حسب هذه الطريقة.

وثانياً، تلزم هذه الطريقة المطورين ببذل الوقت والجهد للتفكير بكل حقل هل هو "nullable" أم لا، وهذا عبء غير ضروري تفرضه هذه الطريقة على المطور.

ثانياً: من أكبر نقاط ضعف أنماط إصلاح الأخطاء هي أنها حتى تعمل تحتاج إلى تاريخ التغييرات الخاص بالكود (Code Change History)، ما يعني أنه في حال عدم توفره فلا يمكن لهذه الطريقة أن تقوم بالإصلاحات المطلوبة، وهذا حال الأكواد الحديثة التي لم يمض وقت طويل على بداية تطورها، أي أنها لا تملك تاريخاً بالتغييرات التي طرأت عليها، إن وجدت أصلاً حتى اللحظة. وكذلك هو حال الأكواد المكتوبة من قبل المطورين المبتدئين الذين يفتقرون إلى الخبرات البرمجية التي تحميهم من ظهور خطأ القيمة الخالية في ملفات أكوادهم، ما يجعلهم دون حماية ضد هذا النوع الخطير من الأخطاء البرمجية.

ثالثاً: أبرز نقاط ضعف مكدس المسك هي أولاً أن عملية بناء مكدس المسك تفرض حملاً زائداً على الكود باعتبار أن كل متحول في المناطق المشتبه بظهور خطأ القيمة الخالية فيها يتم اختباره فيما إذا كان null أم لا عن طريق حقن مجموعة من الاختبارات عليه. وثانياً هي أن الحل عن طريق استبدال القيمة الخالية بقيمة أخرى قد يشكل خطأ منطقي على الكود، لأن المطور قد لا يتمكن من معرفة أين تم الإصلاح وما هي القيمة التي تم استخدامها في عملية الاستبدال.

رابعاً: من أبرز نقاط الضعف في الاعتماد على حكمة الآخرين هي أولاً أنها تعتمد بشكل كامل على التوثيق (Documentation) للكود المطلوب إصلاحه، ولا تغطي في هذه الحالة الأكواد التي لا تملك توثيقاً، وثانياً أن التوثيق يعتبر مهملًا من قبل معظم المطورين، ونادراً ما يكون مكتوباً بشكل واضح ودقيق.

خامساً: تقوم الأداة "Nullsafe" بإضافة مجموعة جديدة من القواعد اللغوية (semantic rules) على جافا، وطبعاً لا تخضع الأكواد المكتوبة قبلاً بلغة جافا لهذه القواعد، ما يجعل الأداة عاجزة عن حل مشكلة خطأ القيمة الخالية في هذا المجال من الأكواد.

6. الحل المقترح:

أفضل الحلول هي التي تحمي الكود من ظهور خطأ القيمة الخالية مع المحافظة على السياق المنطقي للبرنامج، بالإضافة إلى إشراك المطور في إجراء التعديلات التي يراها مناسبة في الحالات التي تستدعي ذلك بالضرورة. فمثلاً لو أخذنا التقنية "NPEFix" وأوجدنا طريقة مناسبة لإشراك المطور بدلاً من استبدال بعض القيم بأخرى اعتباطية، لكان ذلك حلاً أفضل للمشكلة المطروحة.

7. الاستنتاجات والتوصيات:

1. لا يوجد حتى الآن حل شامل وسهل التطبيق ودقيق لمشكلة القيمة الخالية. باعتبار أن التنويهات غير موجودة في الأكواد السابقة فهي إذاً غير شاملة، ولم تكن أنماط إصلاح الأخطاء سهلة التطبيق بسبب اعتمادها على تاريخ التغييرات، وتعتبر عملية بناء مكس المسك ذات تعقيد كبير، وإن الاعتماد على حكمة الآخرين ليس دقيقاً بسبب ضعف عملية التوثيق.
2. بعد قول ذلك، اتخاذ خطوة في طريق الحل أفضل من الوقوف دون حراك. فكل من الطرق السابقة تقلل من حجم الضرر وتقدم إمكانية كبيرة للتطوير مستقبلاً.
3. يختلف الأسلوب المتبع للدفاع عن النظام البرمجي ضد القيمة الخالية باختلاف النظام نفسه. أي في حال كان النظام البرمجي قديماً وكبيراً نسبياً وله تاريخ مقبول من الإصدارات، فمن المنطقي استخدام أنماط إصلاح الأخطاء لحمايته، أما في حال كان النظام قيد الإنشاء ويقوم على تطويره مطورون كفاء (Senior Developers)، فننصح باستخدام التنويهات. أخيراً وليس آخراً، في حال كان النظام صغيراً ولا يهمننا التعقيد الزمني يفضل استخدام مكس المسك.

8. المراجع:

- [1] FÄHNDRICH M, and LEINO K, 2003- **Declaring and checking non-null types in an object-oriented language**. In Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA), pages 302–312. ACM, 2003.
- [2] SUMMERS A, and MÜLLER P, 2011- **Freedom before commitment: a lightweight type system for object 93initialization**. In Object-Oriented Programming, Systems, Languages, and Applications OOPSLA), pages 1013–1032. ACM.
- [3] FÄHNDRICH M, and XIA S, 2007- **Establishing object invariants with delayed types**. In Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA), pages 337–350. ACM, 2007.
- [4] QI X, and MYERS A, 2009- **Masked types for sound object initialization**. In Principles Of Programming Languages (POPL), pages 53–65. ACM.
- [5] SERVETTO M, MACKAY J, POTANIN A, and NOBLE J, 2013- **The billion-dollar fix**. In European Conference on Object-Oriented Programming (ECOOP), pages 205–229. Springer.
- [6] CHALIN P, and JAMES P, 2007- **Non-null references by default in java: Alleviating the nullity annotation burden**. In European Conference on Object-Oriented Programming (ECOOP), pages 227–247. Springer.
- [7] PAPI M, and ERNST M, 2007- **Compile-time type-checking for custom type qualifiers in Java**. In Companion to the 22Nd ACM SIGPLAN Conference on Object-oriented Programming Systems and Applications Companion, ser. OOPSLA '07. New York.
- [8] LOGINOV A, YAHAV E, CHANDRA S, FINK S, RINETZKY N, and NANDA M, 2008- **Verifying dereference safety via expanding scope analysis**. In Proceedings of the International Symposium on Software Testing and Analysis, ser. ISSTA '08. New York.
- [9] OSMAN H, LUNGU M, and NIERSTRASZ O, 2014- **Mining Frequent Bug-Fix Code Changes**, IEEE Conference on Software Maintenance, Reengineering and Reverse Engineering (CSMR-WCRE), Software Evolution Week.

[10] CORNU B, DURIEUX T, SEINTURIER L, and MONPERRUS M, 2015- **NPEfix: Automatic Runtime Repair of Null Pointer Exceptions in Java.**

arXiv:1512.07423v1 [cs.SE]

[11] LEUENBERGER M, OSMAN H, GHAFARI M, and NIERSTRASZ O, 2017- **Harvesting the Wisdom of the Crowd to Infer Method Nullness in Java,** IEEE International Conference on Software Maintenance and Evolution (ICSME), 17-22 Sept. Shanghai, China.

[12] OSMAN H, LEUENBERGER M, LUNGU M, and NIERSTRASZ O, 2016- **Tracking Null Checks in Open-Source Java Systems,** IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER), Osaka, Japan.

[13] HUBERT L, 2010- **A Non-Null Annotation Inferencer for Java Byte Code,** arXiv: 1007.3183v1 [cs.PL]

[14] PIANYKH A, ZORIN L, and LYUBARSKIY D, 2022- **Retrofitting Null-Safety Onto Java at Meta,** <https://engineering.fb.com/2022/11/22/developer-tools/meta-java-nullsafe>

دراسة تأثير المتغيرات المختارة (سرعة دوران مضرب تفتيت السماد - شكل الشفرة) في أداء موزعات السماد العضوي

¹الباحثة: م. غنوة علي حسين

المخلص

تم إجراء البحث لمعرفة خواص المتغيرات المختارة (سرعة دوران مضرب التفتيت - شكل الشفرة) وأثرها في تشغيل آلة نثر السماد العضوي من حيث جودة التفتيت واستهلاك الوقود، حيث تم دراسة أشكال مختلفة لشفرات مضارب التفتيت (V-C-J-L-مشرشر) وذلك عند ثلاث سرعات مختلفة للمضرب (180-215-225 دورة/دقيقة) حيث بينت النتائج بأن الشفرات التي على شكل حرف (V مشرشر) قد حققت أقل جودة تفتيت وأكبر متوسط استهلاك للوقود مع زيادة السرعة الأمامية للدوار مقارنة بالشفرات الثلاث الباقية، التي حققت أفضل جودة تفتيت عند الشفرات على شكل حرف J تليها C ثم L، وأقل متوسط استهلاك للوقود كان عند الشفرات التي على شكل حرف C، ثم J، ثم L على التوالي.

الكلمات المفتاحية: شفرات مضارب تفتيت السماد - آلة نثر السماد - جودة التفتيت - استهلاك الوقود

¹ ماجستير - مكنة زراعية - كلية الهندسة التقنية - جامعة طرطوس.

A study of the effect of selected variables (rotation speed of the compost crushing paddle - blade shape) on the performance of the organic fertilizer spreaders

²Eng:Ghenwa ali hussen

Abstract

The research was conducted to find out the properties of the selected variables (the rotation speed of the pulverizing paddle - the shape of the blade) and their impact on the operation of the compost spreader in terms of pulverizing quality and fuel consumption. 180-215-225 revolutions/min), where the results showed that the V-shaped blades (serrated) achieved the lowest pulverization quality and the highest average fuel consumption with the increase in the front speed of the rotor compared to the remaining three blades, which achieved the best pulverization quality when the letter-shaped blades J followed by C, then L, and the lowest average fuel consumption was at the blades in the shape of the letter C, then J, and then L, respectively.

Keywords: Compost paddle blades - manure spreader- shredding quality- fuel consumption

² master- Agricultural Mechanization -Faculty of Technical Engineering

1-المقدمة

تتصف التربة غير الخصبة بأنها ذات نسج خفيفة عالية النفاذية للماء وبالتالي فان قابليتها على الاحتفاظ بالعناصر الغذائية منخفضة جداً. لذلك فالنبات في مثل هذه التربة غير قادة على سد حاجتها من العناصر الغذائية وهذا هو السبب الرئيسي في الإضافات الكبيرة للأسمدة لسد حاجة النباتات. [1]

عموماً فإن الأساس المعوّل عليه في نجاح زراعة النباتات هو عملية التحضير الجيد للتربة، يعدّ تحضير الأرض من العمليات المهمة للزراعة ويعتمد على عملية الحرث التي يتم إجراؤها، حيث يؤدي تحسين الحرث الى تهيئة أفضل للأرض. تتضمن عملية الحرث تكسير سطح التربة إلى عمق معيّن وتخفيف كتل التربة باستخدام معدات مثل المحراث الحفار، والقلاب (المطرحي، القرصي)، وغير ذلك ولكنها تتطلب عمليات متعددة. [2] والحرث الأولى والثانوية تعتمد على الميكنة الزراعية وفي أيامنا هذه يتم توجيه اهتمام كبير نحو الآلات الزراعية التي تعمل بالطاقة فمثلا تضاف الأسمدة العضوية بواسطة موزعات السماد خاصة بأشكالها المتعددة التي تعتمد بدورها على تفتت السماد وخلطه بالتربة وتعتمد جودتها على نوع الشفرات المستخدمة للمضرب تفتت السماد وسرعة الدوران. [2]

يتم تقييم أداء آلة نثر السماد العضوي الصلب من خلال تحديد معدّل أداء الآلة، كفاءة توزيع السماد، تجانس توزيع السماد، سلاسة العمل بين وحدات الآلة، الوثوقية. [3] قام [4] بتطوير واختبار آلة نثر السماد مقطورة بجرار وذلك بتطبيق سرعات أمامية مختلفة (0.41, 0.69, 1.12) m/s، مع إنتاجية (1.05-3.06-2.06) kg/s، وبيّنت هذه الدراسة أنّه مع زيادة السرعة الأمامية للجرار انخفض معدّل نثر السماد.

قام [5] بدراسة حركة السماد على موزع الأسمدة بهدف تطوير عملية نثر الأسمدة لتحسين انتظام توزيع السماد وكانت النتائج أن حركة جزيئات السماد العضوي على سطح شفرات محور التوزيع تتعلق بشكل كبير بسرعة دوران المحور ونصف قطر المحور مع الشفرات، وارتفاع المحور عن حصىرة العربة.

أشار [6] عند دراسته تأثير المتغيرات المختارة (السرعة الأمامية - سرعة الدوار - عدد ريش الشفرات) على كثافة السماد، انخفاض الكثافة مع زيادة السرعة الأمامية، وعدد الريش لكل شفرة من شفرات الدوار، أن أقل كثافة كانت (0.510-0.515) g/cm^3 ، تم الحصول عليها عند سرعة دوار (7.86 m/s) وثلاث ريش لكل شفرة.

قام [7] بتصميم نموذج لآلة نثر السماد العضوي وأكد على زيادة الإنتاجية بنسبة 150%، بالإضافة الى مساحتها الأقل في الحقل، ليست بحاجة لخبرة من العمال والاستفادة من بقايا السماد بإنتاج الغاز الحيوي.

درس [8] عدم انتظام نثر السماد العضوي نتيجة تفاعله مع الجزء الدوار الذي يفتته، وحركة السماد المتناثر. بينت الدراسة أن درجة التفتيت تزداد مع زيادة سرعة دوران الجزء الدوار وبناء على هذه النتائج قام بتطوير نماذج رياضية لتحسين أداء الآلة.

أشار [9] بدراسته على تأثير سرعات مختلفة لمأخذ القدرة الخلفي للجرار على مؤشرات عمل آلة نثر الأسمدة القرصية مبيناً أن سرعة دوران مأخذ القدرة الخلفي للجرار (750 rpm) أعطى أفضل نتائج لعمل الآلة وأفضل انتظام لتوزيع السماد.

أجرى [10] بحثاً يتناول التحكم بعرض العمل وجودة التسميد في السماد العضوي بالطرد المركزي، فتم تصميم برنامج محاكاة يأخذ بعين الاعتبار جميع العوامل المؤثرة في تشغيل الآلة للحصول على أفضل أداء.

2- أهمية وأهداف البحث:

يهدف البحث إلى تقييم تأثير أشكال مختلفة من شفرات مضارب تقنيت السماد في آلة نثر السماد العضوي الصلب على بعض مؤشرات الأداء، وذلك عند ثلاث سرعات مختلفة للدوار، وبالتالي تكمن أهمية البحث في اختيار أفضل نوع من الشفرات المستخدمة مع أفضل سرعة دورانية لمضرب التقنيت، والتي تحقق أفضل مؤشرات لآداء آلة النثر بما يضمن أيضاً تحقيق أفضل إنتاجية.

3- مؤشرات البحث:

1- جودة التقنيت ب (mm).

2- استهلاك الوقود (لتر/هكتار).

4- منهجية البحث

- دراسة تحليلية: حيث تم دراسة وتحليل الدراسات المرجعية المتعلقة بالآلات التسميد ومؤشرات عملها.

- دراسة تجريبية: حيث تم تجريب واختبار الشفرات المستخدمة على مضرب تقنيت السماد العضوي عند سرعات مختلفة للمضرب ومقارنتها من حيث بعض مؤشرات الأداء لاختيار الأفضل منها.

5- حساب المؤشرات المدروسة:**5-1- جودة التقنيت:**

تم تقييمها عن طرق غريال قياس درجة التقنيت الذي يتألف من أربع غرابيل متدرجة الأقطار من الأعلى إلى الأسفل كما هو موضح في (الشكل 1).



الشكل(1): جهاز قياس درجة تفتيت السماد/مخبري

- الغربال الأول: بقطر (40 [cm]) عدد الثقوب / 12 / وقطر الثقب [mm] 40.
 - الغربال الثاني: بقطر (30 [cm]) عدد الثقوب / 18 / وقطر الثقب [mm] 29.
 - الغربال الثالث: بقطر (25[cm]) عدد الثقوب / 24 / قطر الثقب [mm] 25.
 - الغربال الرابع: بقطر (15 [cm]) وهي مصممة لتجمع عليها الأتربة المتساقطة.
- تم حساب جودة التفتيت لكل نوع من أنواع الشفرات من العلاقة: [11]

$$D \% = \sum m \leq 25 \setminus \sum m * 100 \quad (1)$$

$\sum m$.: وزن العينة الكلي [kg].

5-2- استهلاك الوقود (QF) (Fuel Consumption):

تعد كمية الوقود المصروفة لإنجاز أي عملية زراعية أحد المؤشرات الأساسية في تقييم كفاءة الآلة، إذ إنّ كمية الوقود المستهلكة من قبل الآلة تعتمد على عوامل عدة منها

القدرة الحصانية للمحرك، ونوع الوقود، والزمن اللازم لإنجاز العملية الزراعية، ونوع العملية الزراعية. [11]

$$FU.C = \frac{Q_d}{T_E} \times 3.6 \quad (2)$$

حيث أن:

$FU.C$: كمية الوقود المستهلكة لوحدة الزمن (ل/سا).

Q_d : كمية الوقود المستهلكة خلال المعاملة (مل).

T_E : الزمن الفعلي خلال المعاملة (ثانية).

6- الدراسة العملية وتقييم التجربة:

أجريت دراسة لتقييم تأثير أشكال مختلفة من شفرات مضرب تفتيت السماد وسرعة دورانه على جودة التفتيت واستهلاك الوقود لآلة نثر السماد العضوي الصلب الموجودة في مزرعة زاهد في محافظة طرطوس، وقد تم دراسة النتائج عند نفس مواصفات آلة نثر السماد العضوي. الشكل(2).



الشكل(2): آلة نثر السماد العضوي

الجدول (1): بعض مواصفات آلة نثر السماد المستخدم

3.60م	طول العربة
1.2م	عرض العربة
1.42م	العرض التصميمي

تم تشغيل الآلة على سرعة أمامية ثابتة 4.5 [كم/سا] حيث بلغ عرض العمل 2 [م] ومسافة اختبار

50 [م] لكل معاملة ،استخدم أربع أنواع من شفرات مضرب تفتيت السماد وهي :

- النوع الأول: الشفرات على شكل حرف V مشرشر كما في الشكل (3).

- النوع الثاني: الشفرات على شكل حرف L كما في الشكل (4).

- النوع الثالث: الشفرات على شكل حرف C كما في الشكل (5).

النوع الرابع: الشفرات على شكل حرف J كما في الشكل (6).

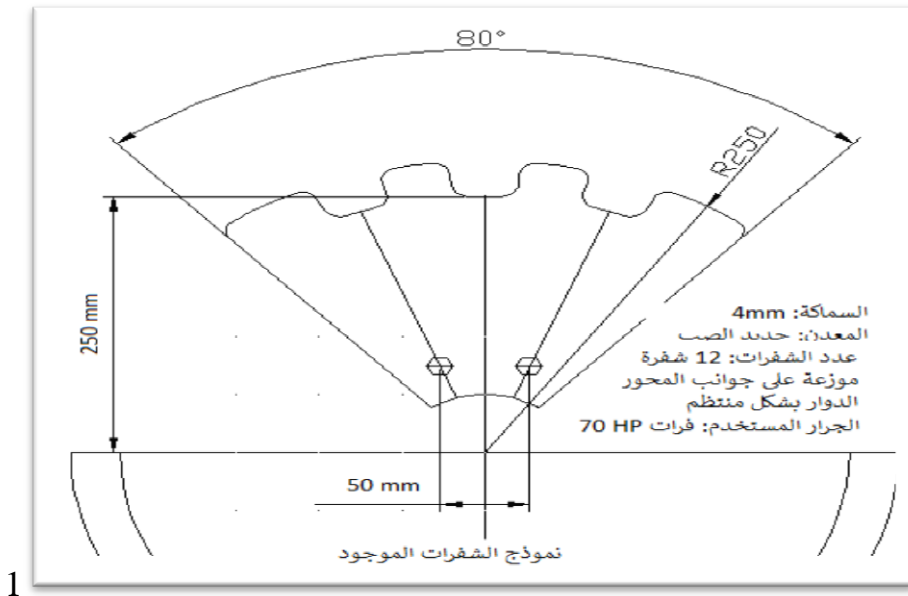
وبثلاث سرعات للمضرب (180 ، 215 ، 225) دورة/ دقيقة. بلغت عدد المعاملات 12

معاملة ولثلاث مكررات، حيث بلغ عدد القطع التجريبية 36.

أخذت العينات من القطع التجريبية بشكل عشوائي ومن مساحة 5 م² لتقييم جودة التفتيت

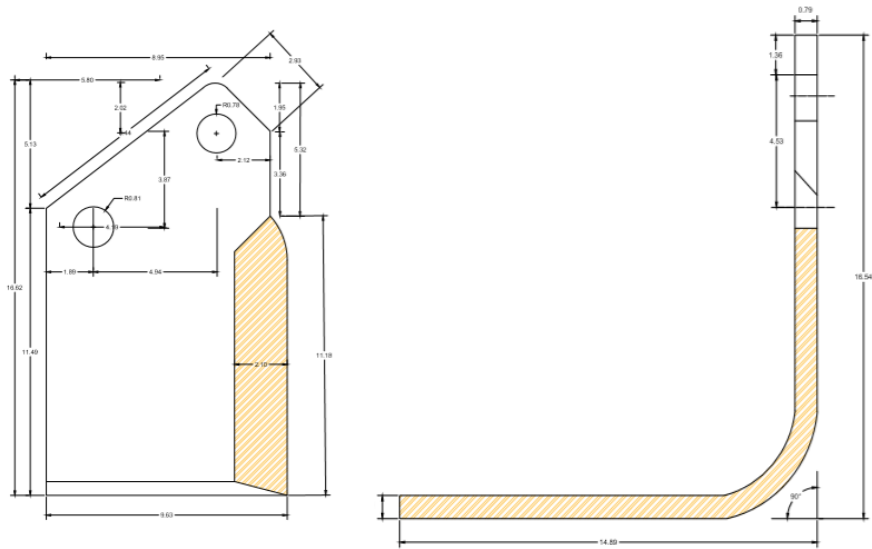
بواسطة غربال قياس درجة التفتيت [مم] . كما تم تقدير الاستهلاك الساعي للوقود [ل/سا]

وبدلالة معدل أداء الآلة [ه/سا] تم تحديد الاستهلاك النوعي للوقود [ل/ه] .



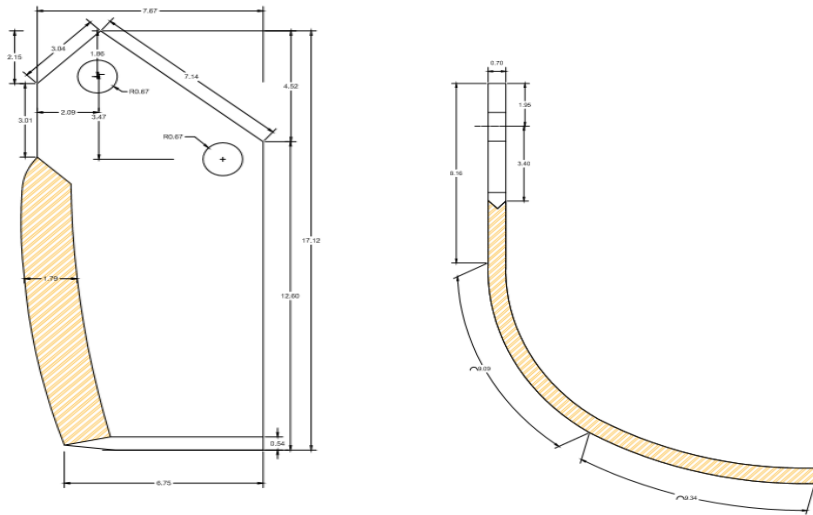
1

الشكل(3): مسقط أمامي للشفرات على شكل حرف V مشرشر.

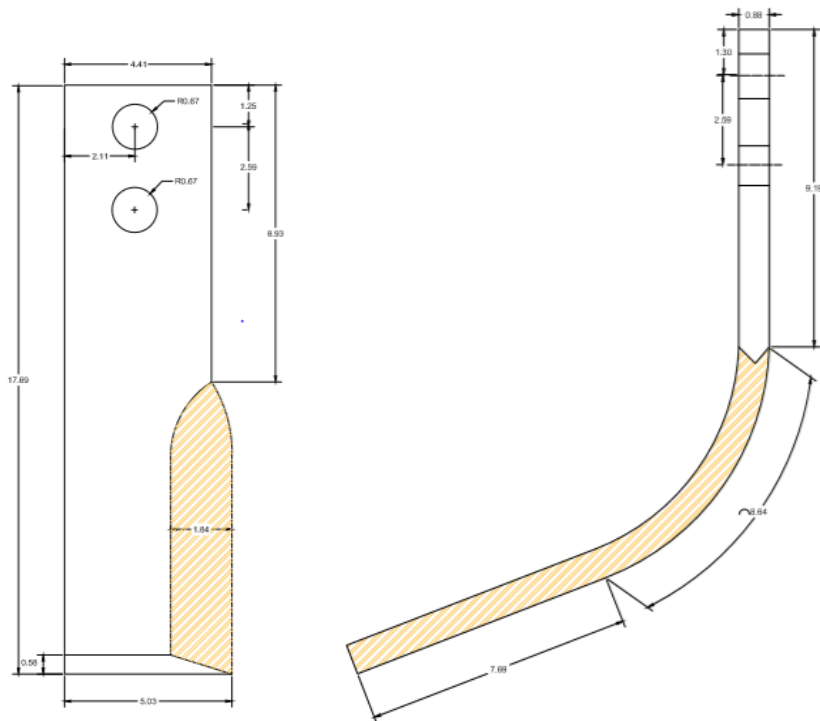


الشكل (4): مسقط للشفرات على شكل حرف L.

دراسة تأثير المتغيرات المختارة (سرعة دوران مضرب تفتيت السماد-شكل الشفرة) في أداء موزعات السماد العضوي



الشكل (5): مسقط للشفرات على شكل حرف C.



الشكل (6): مسقط للشفرات على شكل حرف J

7-النتائج والمناقشة

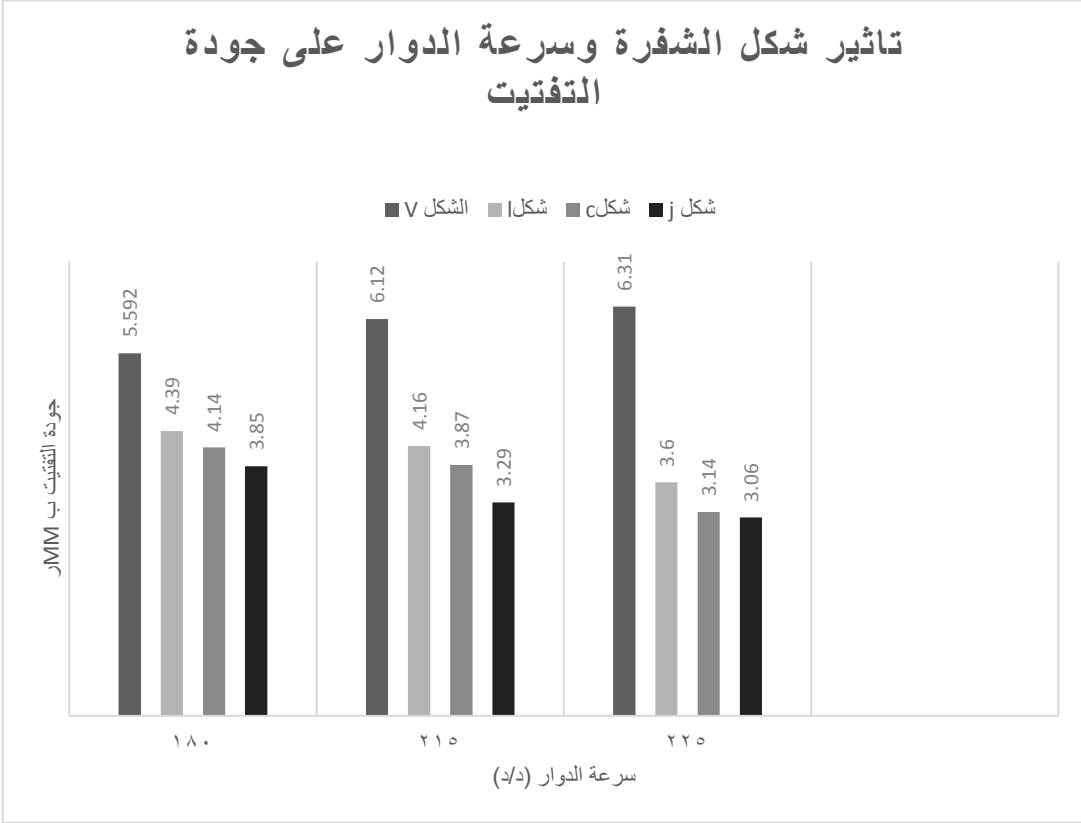
7-1-تأثير شكل الشفرة وسرعة مضرب التفتيت على جودة التفتيت:

تم قياس جودة التفتيت لكل نوع من أنواع الشفرات من خلال العلاقة (1) ودونت النتائج كما هو موضح بالجدول (2) ووضعت النتائج على شكل مخطط كما هو موضح في الشكل (7)

الجدول (2) تأثير شكل الشفرة وسرعة الدوار على جودة التفتيت.

جودة التفتيت ب (mm) حسب شكل الشفرة				سرعة الدوار (د/د)
J	C	L	مشرشر V	
3.85	4.14	4.39	5.59	180
3.29	3.87	4.16	6.12	215
3.06	3.14	3.60	6.31	225
3.4	3.71	4.05	6.0	المتوسط

تأثير شكل الشفرة وسرعة الدوار على جودة التفتيت



الشكل(7): تأثير شكل الشفرة وسرعة الدوار على جودة التفتيت.

يلاحظ من الشكل(7) أن درجة التفتيت كان الأفضل في حال الشفرة على شكل z (3.4mm) تليها الشفرات على شكل c(3.7mm) و أخيرا الشفرات على شكل a (4.05mm)، لأنه كلما قل قطر حبيبات السماد تحقق عنده أفضل تفتيت مع زيادة السرعة الدورانية لمضرب التفتيت بينما حققت الشفرات على شكل حرف V مشرشر أقل جودة لدرجة تفتيت مقارنة بباقي الشفرات حيث بلغ متوسط قطر كتل السماد عندها(6mm) أي انها حافظت على اكبر قطر لكتل السماد وقلة جودة التفتيت مع

زيادة السرعة الدورانية للمضرب . ومنه نبين التأثير الإيجابي لمساحة القطع حيث كلما زادت مساحة القطع لكتل السماد تحقق عندها افضل جودة تفنيت.

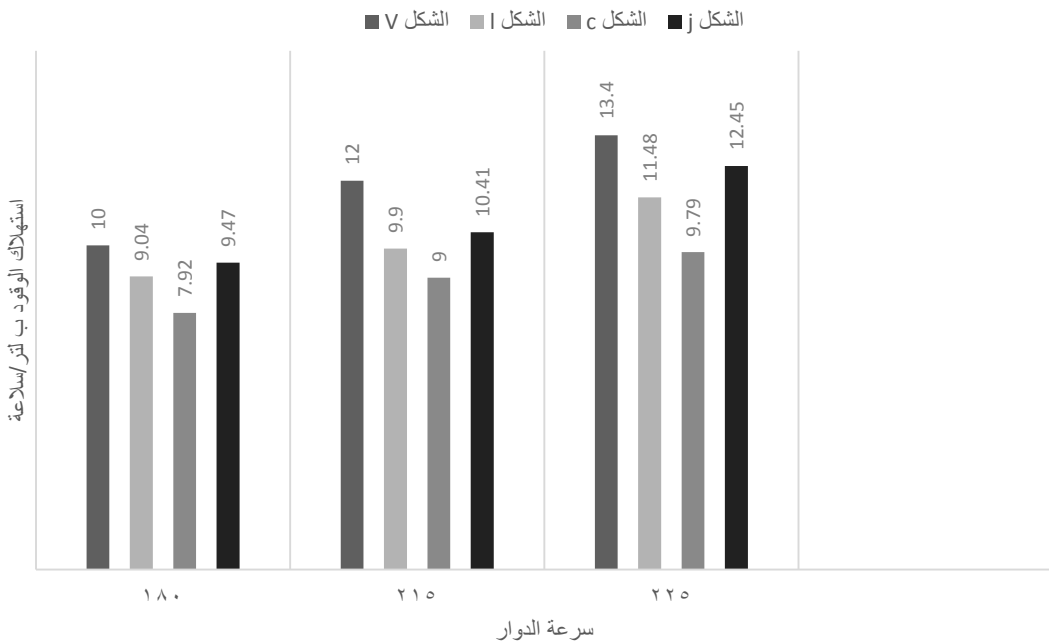
7-2- تأثير شكل الشفرة وسرعة الدوّار على استهلاك الوقود:

تم حساب استهلاك الوقود للجرار لكل نوع من أنواع الشفرات من خلال العلاقة (2) ودونت النتائج كما هو موضح بالجدول (3) ووضعت النتائج على شكل مخطط كما هو موضح في الشكل (8).

الجدول (3) تأثير شكل الشفرة وسرعة الدوار على استهلاك الوقود

استهلاك الوقود ب (لتر/ساعة) حسب شكل الشفرة				سرعة الدوار(د/د)
J	C	L	مشرشر	
9.47	7.92	9.04	10	180
10.1	9	9.9	12	215
12.45	9.79	11.48	13.4	225
10.67	8.90	10.14	11.8	المتوسط

تأثير شكل الشفرة وسرعة الدوران على استهلاك الوقود



الشكل (8): تأثير شكل الشفرة وسرعة الدوران على استهلاك الوقود.

يلاحظ من المخطط البياني الموضح في الشكل (8)، أنه كان متوسط استهلاك الوقود أقل في حالة استخدام الشفرات ذات الشكل C (8.90 لتر/ساعة)، في حين كان متوسط استهلاك الوقود للشفرات ذات الشكل L (10.14 لتر/ساعة)، وللشفرات ذات الشكل J (10.67 لتر/ساعة). على التوالي وجميع اشكال الشفرات قد حققت استهلاك للوقود أقل من متوسط استهلاك الوقود للشفرات على شكل حرف V مشرشر (11.8 لتر/ساعة).

بينما يلاحظ زيادة استهلاك الوقود مع زيادة سرعة الضرب لجميع أشكال الشفرات، وذلك بسبب حقيقة أنه مع زيادة سرعة الضرب يتطلب الأمر مزيداً من الطاقة المقدمة من عمود الإدارة الخلفي للجرار .

ومن خلال الشكل (7-8) نلاحظ ان الشفرات ذوات الاشكال J-C-L قد حققت افضل تفتيت مع زيادة السرعة الدورانية للمضرب واقل استهلاك للوقود مقارنة بالشفرات ذات شكل V مشرشر التي لم يتحقق عندها تفتيت جيد مع زيادة سرعة الضرب واستهلاك كبير للوقود.

8- الدراسة الإحصائية للنتائج

استخدم في التحليل الاحصائي برنامج excel2016، لمقارنة الفروقات بين المتوسطات بحساب قيمة أقل فرق معنوي عند LSD عند مستوى دلالة 5%، كما درست العلاقة الارتباطية وقوتها بين بعض الصفات المدروسة.

5-1- تأثير سرعة دوران مضرب تفتيت السماد وشكل الشفرة في أداء موزعات السماد العضوي

لتقييم أداء موزعات السماد ولمقارنة جودة تفتيت كتل السماد واستهلاك الوقود عند السرع المدروسة لمضرب تفتيت السماد، أُجريت تجربة باستخدام تصميم العشوائية الكاملة بواسطة برنامج excel2016، ولأجل المقارنات البعدية وتحليل تباين التجربة أُجري اختبار LSD وفق العلاقة الآتية (12).

$$LSD = T_{dfe} 0.05 \sqrt{\frac{2MSe}{m}}$$

أولاً-تأثير سرعة دوران مضرب تفتيت السماد وشكل الشفرة في استهلاك الوقود

جدول (4) حساب الفروقات المعنوية

5% LSD	ti	\bar{t}_i	$\overline{sh1}$	$\overline{sh2}$	$\overline{sh3}$	$\overline{sh4}$
		\bar{x}	8.90	10.14	10.67	11.8
0.15	$\overline{sh4}$	11.8	2.9 .	1.66	1.13	0
	$\overline{sh3}$	10.67	1.27	0.53	0	1.13
	$\overline{sh2}$	10.14	1.24	0	0.53	1.66
	$\overline{sh1}$	8.90	0	1.24	1.27	2.9

يلاحظ من الجدول (4) أن استهلاك الوقود قد تأثر معنوياً مع تغير شكل الشفرة وزيادة سرعة دوران مضرب تفتيت السماد حيث تفوقت الشفرات على شكل حرف C بمعنوية عالية جداً على باقي الشفرات مسجلة أقل قيمة لاستهلاك الوقود تليها الشفرات على شكل حرف L ثم الشفرات على شكل حرف J وأخيراً الشفرات على شكل حرف V مشرشر ومنها نستنتج ان الشفرات على شكل حرف C هي الأفضل لتحقيق أقل استهلاك للوقود .

حيث $sh1$: ترمز الى النوع الأول من الشفرات (الشفرات على شكل حرف C)

$sh2$: ترمز الى النوع الثاني من الشفرات(الشفرات على شكل حرف L)

$sh3$: ترمز الى النوع الثالث من الشفرات (الشفرات على شكل حرف J)

$sh4$: ترمز إلى النوع الرابع من الشفرات (الشفرات على شكل V مشرشر)

أولاً-تأثير سرعة دوران مضرب تفتيت السماد وشكل الشفرة في تفتيت كتل السماد

جدول (5) حساب الفروقات المعنوية

5% LSD	ti	\bar{t}_i	$\overline{Sh3}$	$\overline{Sh1}$	$\overline{sh2}$	$\overline{sh4}$
		\bar{x}	3.4	3.71	4.05	6
0.16	$\overline{Sh4}$	6	2.6	2.29	1.95	0
	$\overline{Sh2}$	4.05	0.65	0.34	0	1.95
	$\overline{Sh1}$	3.71	0.31	0	0.43	2.29
	$\overline{sh3}$	3.4	0	0.31	0.65	2.6

يلاحظ من الجدول (5) أن جودة تفتيت كتل السماد قد تأثرت معنوياً مع تغيير شكل الشفرة وزيادة سرعة دوران مضرب تفتيت السماد حيث تفوقت الشفرات على شكل حرف J بمعنوية عالية جداً على باقي الشفرات مسجلة أقل درجة تفتيت تليها الشفرات على شكل حرف C ثم الشفرات على شكل حرف L بينما حققت الشفرات على شكل حرف V مشرشر أقل درجة تفتيت لكتل السماد مقارنة بباقي الشفرات ومنه نبين التأثير الإيجابي لمساحة القطع حيث كلما زادت مساحة القطع لكتل السماد تحقق عندها أفضل قيمة تفتيت.

حيث $sh1$: ترمز إلى النوع الأول من الشفرات (الشفرات على شكل حرف C)

$sh2$: ترمز إلى النوع الثاني من الشفرات (الشفرات على شكل حرف L)

$sh3$: ترمز إلى النوع الثالث من الشفرات (الشفرات على شكل حرف J)

$sh4$: ترمز إلى النوع الرابع من الشفرات (الشفرات على شكل V مشرشر)

9-الاستنتاجات

1. تحقق عند الشفرات التي على شكل حرف J أفضل تفتيت للسماد تليها الشفرات على شكل حرف C ثم الشفرات على شكل حرف L وأخيراً الشفرات على شكل حرف V مشرشر.
2. تحقق عند الشفرات التي على شكل حرف C أقل استهلاك للوقود تليها الشفرات على شكل حرف L ثم الشفرات على شكل حرف J وأخيراً الشفرات على شكل حرف V مشرشر.
3. ازداد استهلاك الوقود مع زيادة السرعة الدورانية لمضرب تفتيت السماد لجميع أنواع الشفرات

10-المقترحات:

1. دراسة تأثير سرعة جهاز التلقين في جودة تفتيت السماد.
2. دراسة تأثير مواصفات السماد العضوي في جودة تفتيتها.
3. تصميم عناصر (شفرات) بأشكال مختلفة غير معدنية دوارة لتخفيف الوزن وزيادة فعالية النثر.

- المراجع References

- 1- المشهداني، عبد الستار. (2009). من محاضرات الدورة التدريبية الأولى للموسم الزراعي (2009-2010) للمهندسين والمرشدين الزراعيين.
- 2- Shiva Bhambota.,Gurusahib Singh Manes., Apoor Prakash and Anoop Dixit. 2014. Effect of blade shape and rotor speed of rotavator on pulverization and mixing quality of soil. Agricultural Engineering Today. 38(4): 25-30
- 3-مهيوب، علي. (2013). آلات البذر والزراعة، جامعة تشرين، كلية الهندسة التقنية.
- 4-Singh, R.C and Singh, C. D. 2013. Development and Performance Testing of a Tractor Trailer-cum-Farmyard Manure Spreader. Agricultural Mechanization in Asia, Africa and Latin America. 37(2): 1-6.
- 5-Vasilica, S; David, L and Popa, I. 2019. STUDY OF The PARTICLE MOTION ON A MANURE FERTILIZER SPREADER. ISSN.
- 6- nkrGurUdatT mAddu,2015.performance evaluation of farmyard manure pulverize cum spreader. Acharya n. g range agricultural University.
- 7-Swanad Sudhirpachpore,2018.Design&fabrication Of Organic fertilizer Manufacturing Machine. International Research Journal of Engineering and Technology.
- 8-Vasilica, S; PETRU, C and LuCRETIA, P. 2019. EXPERIMENTAL RESULTS ON THE SOLID ORGANIC FERTILIZER MACHINE MG5. Bucharest, Romania in China.8:352-360.
- 9-Sumer, S.K, 2018. THE EFFECT OF PTO OPTIONS ON OPERATIONAL CHARACTERISTICS OF DISC FERTILIZER SPREADER. CanakkaleOnsekiz Mart University, Turkey.

10-Le, T.T. 2017. CENTRIFUGAL FERTILIZER SPREADER: CONTROL OF WORKING WIDTH AND FERTILIZATION QUALITY. European Society of Agricultural Engineers, pp.97-104 ref.22.

11- نصور، محمد، 2015. اختبار ومعايرة الآلات الزراعية، منشورات جامعة تشرين، كلية الهندسة التقنية.

12- اسماعيل، ريم، (2021). مقرر الإحصاء وتصميم التجارب. جامعة طرطوس، كلية الهندسة التقنية.

معالجة مشكلة الملفات الصغيرة في هادوب

طالب الدراسات العليا: عمار العلي كلية الهندسة المعلوماتية - جامعة: البعث

إشراف الدكتور: محسن عبود

الملخص

إن نظام ملفات هادوب الموزع (HDFS) هو تحقيق مفتوح المصدر لأباتشي مخصص لتخزين الملفات الكبيرة، ولكن عند الحاجة لتخزين عدد كبير من الملفات الصغيرة، يواجه HDFS بعض المشكلات حيث تتم إدارة الملفات جميعها في HDFS من قبل عقدة واحدة تسمى السيد (NameNode) (Master)، التي تتعامل مع البيانات الوصفية لعدد كبير من العقد تسمى العبيد (Slaves) (DataNodes). تخزن عقدة NameNode جميع البيانات الوصفية في ذاكرة الوصول العشوائي الخاصة بها. لذلك، عند التعامل مع عدد كبير من الملفات الصغيرة، وبما أن كل ملف صغير يستهلك كتلة على نحو فردي مما يؤدي إلى زيادة متطلبات الذاكرة لعقدة NameNode وغالباً ما تصبح عقدة NameNode عنق زجاجة ل HDFS لأنها قد تنفذ من الذاكرة.

في هذه الدراسة تمت معالجة مشكلة الملفات الصغيرة في هادوب عن طريق تحسين (Hadoop Perfect File) (HPF)، وهو نظام أرشفة جديد قائم على الفهرسة لتحقيق أداء أفضل، وتم تحسين (Hadoop Perfect File) (HPF) عن طريق الاستفادة من ذاكرة الوصول العشوائي (RAM) لعقدة (Client) في عمليات التخزين المؤقت وال جلب المسبق للملفات الصغيرة بدلاً من الاعتماد دائماً على الأقراص للقراءة، وتم اقتراح خوارزمية للمحافظة على الملفات المطلوبة مؤخراً في ذاكرة العميل مما خفض من زمن الوصول للملفات وحسن الأداء.

الكلمات المفتاحية: الملفات الصغيرة، ملف هادوب المثالي، نظام ملفات هادوب الموزع، الجلب المسبق، النظم الموزعة.

Small Files Problem Handling In Hadoop

Abstract

The Distributed Hadoop File System (HDFS) is an open source implementation of Apache dedicated to storing large files, but when you need to store a large number of small files, HDFS faces some problems as the files are all managed in HDFS by a single node called Master (NameNode), and DataNodes that deal with the metadata of a large number of nodes are called Slaves. The NameNode stores all metadata in its RAM. Therefore, when dealing with a large number of small files, since each small file individually consumes a block, this leads to an increase in the memory requirements of the NameNode, and the NameNode often becomes a bottleneck for HDFS. Because it may run out of memory.

In this study, the problem of small files in Hadoop was addressed by optimizing HPF (Hadoop Perfect File), a new indexing-based archiving system for better performance, is improved by utilizing the client node's RAM for caching and prefetching of small files instead of always relying on disks to read, and an algorithm was proposed to keep the recently requested files in the client's memory, which reduced file access time and improved performance.

KEYWORDS: Small Files, HPF, HDFS, prefetch, Distributed Systems .

1. المقدمة:

لقد شهدت السنوات القليلة الماضية من تَقَاة الإنترنت وعالم الحاسوب الكثير من النمو والشعبية في مجال الحوسبة السحابية. ونتيجة لذلك، فإن التطبيقات السحابية قد ولدت بيانات ضخمة. هادوب، نظام موزع مفتوح المصدر مصمم من أباتشي، ويستخدم بكفاءة لمعالجة كمّية هائلة من البيانات بأسلوب متوازٍ وموزع ضمن العناقيد الكبيرة، ويعمل وفقاً لمبادئ الحوسبة المتوازية. يستخدم هادوب معمارية السيد والعبيد، ويوفر قابلية التوسع والموثوقية إلى حد كبير، ويساعد المستخدم في نشر تطبيقات وتنفيذ عمليات على مجموعة من العقد. مع أن هادوب مستخدم على نطاق واسع من قبل العديد من الشركات مثل Yahoo و IBM و Facebook و Amazon وما إلى ذلك، لا يزال لديه القليل من المشكلات. واحدة من هذه المشكلات هي عدم قدرة هادوب على التعامل مع الملفات الصغيرة [2].

الشبكات الاجتماعية والتجارة الإلكترونية والمكتبات الرقمية والرعاية الصحية والأرصاد الجوية وصور الأقمار الصناعية ليست سوى أمثلة قليلة على التطبيقات التي تنتج كمّية كبيرة من البيانات، ولكن في شكل ملفات صغيرة.

على سبيل المثال، تولد التطبيقات المنتشرة على الخادم العديد من ملفات السجل (Log) في الساعة أو يومياً. بغض النظر عن حجم موقع الويب، يمكن أن يعطي تحليل ملفات السجل إجابات مباشرة للمشكلات التي تتم مواجهتها على مواقع الويب. يعد تحليل السجل مفيداً لإجراء عمليات تحسين محركات البحث، ومراقبة موقع الويب. غالباً ما يكون حجم ملفات البيانات هذا صغيراً، ويتراوح من بعض كيلوبايت إلى عدة ميجابايت، ولكنها مهمة جداً لتحليل البيانات.

تولد الملفات الصغيرة الضخمة الكثير من البيانات الوصفية في نظام ملفات هادوب الموزع HDFS، وبما أن عقدة NameNode تحتفظ بجميع بياناتها الوصفية في الذاكرة، فقد يتسبب ذلك في نفاذ ذاكرتها وإلحاق الضرر بأدائها. بالإضافة للمشكلات الأخرى التي تسببها الملفات الصغيرة [3].

نظراً إلى عدم وجود نظام ملفات موزع يعمل جيداً مع كميات هائلة من الملفات الصغيرة لذلك دراسة مشكلة الملفات الصغيرة في هادوب اتجاه بحثي له أهمية عملية مباشرة.

2. الهدف من البحث:

يهدف البحث إلى التعامل مع مشكلة الملفات الصغيرة في HDFS، وبما أن HDFS مصمم على نمط «كتابة مرة واحدة، قراءة مرات عدّة». لذلك تحسين سرعة عملية قراءة الملفات أكثر أهمية من تحسين سرعة عملية كتابة الملفات [4].

في هذا البحث تم العمل على بناء نظام أرشفة قائم على الفهرسة يأخذ في الاعتبار مشكلة نفاذ ذاكرة NameNode عن طريق تقليل عبء البيانات الوصفية في ذاكرتها ومشكلة عنق الزجاجة لعقدة NameNode مع ضمان الوصول السريع إلى الملفات الصغيرة [3] عن طريق توفير آلية الجلب المسبق والتخزين المؤقت للملفات الصغيرة عند العميل، مما يقلل عمليات الدخل والخرج الإضافية على القرص، ويقلل طلبات البيانات الوصفية على عقدة NameNode، ويقلل الحسابات الناتجة عن معالجة ملفات الفهرسة في أثناء الوصول، ويؤدي إلى تحسين سرعة عملية القراءة وتحقيق أفضل الأداء عند قراءة الطلبات وتخفيف الحمل على عقدة NameNode.

3. مواد وطرائق البحث:

تم نشر هادوب على أجهزة افتراضية للاستفادة من جميع المزايا الافتراضية، والتي يمكن أن تجعل من إدارة العنقود أسهل، كما أنّ الأجهزة الافتراضية سريعة النشر ويمكن نقلها دون مخاطر، استخدمنا في بحثنا برنامج VMware وسيحتوي العنقود في هادوب على ثلاثة عقد تمتلك كل منها المواصفات التالية:

Operating System: Ubuntu 20.04 – Memory: 4GB –

Total Processor cores : 2 –Hard Disk: 25 GB SSD.

1. العقدة master ستكون العقدة الرئيسيّة (192.168.75.128).
2. العقدة node1 عقدة البيانات الأولى (192.168.75.129).
3. العقدة node2 عقدة البيانات الثانية (192.168.75.130).

مواصفات جهاز الحاسوب المستخدم:

- Processor : Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz
1.80 GHz
- Operating System: Windows10 Pro
- Memory: 20GB DDR4, 2400MHz.
- Graphic: AMD Radeon™ R5 M330 , with 2GB, VRAM.
- Storage: HDD 1TB + SSD Hard Drive 240 GB.

4. البيانات الضخمة:

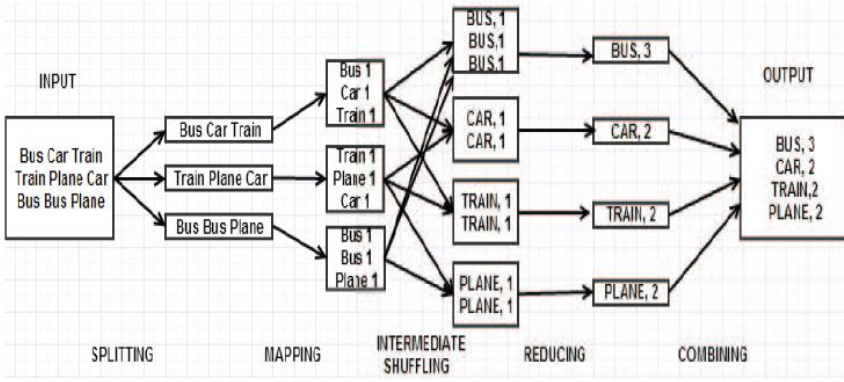
يشير مصطلح البيانات الضخمة إلى النمو الكبير في حجم البيانات التي تتضمن أشكالاً غير متجانسة: بيانات منظمة، بيانات غير منظمة وبيانات شبه منظمة. البيانات الضخمة لها طبيعة معقدة التي تتطلب موارد كبيرة وأساليب جديدة وتقنيات قوية وخوارزميات متقدمة وعلى نحو أكثر دقة، البيانات الضخمة تحتاج إلى المعالجة والتحليل والأمان والتخزين وتوفير إمكانية الوصول الدقيق إليها. نتيجة لمشاريع البيانات الضخمة المختلفة في جميع أنحاء العالم، تم إنشاء العديد من نماذج البيانات الضخمة وأطر العمل والتكنولوجيا الجديدة لتوفير المزيد من سعة التخزين والمعالجة المتوازية والتحليل بالزمن الحقيقي من مختلف المصادر غير المتجانسة. إضافة إلى ذلك، تم تطوير حلول جديدة لضمان خصوصية البيانات وأمنها. بالمقارنة مع التقنيات التقليدية، توفر هذه الحلول المزيد من المرونة، قابلية التوسع والأداء [6] [7].

5. تخزين ومعالجة البيانات الضخمة:

أباتشي هادوب هي منصة بيانات ضخمة معروفة ولديها مجتمع داعم مهم. تم تصميمها لتجنب الأداء المنخفض والتعقيد الذي يتم مواجهته عند معالجة وتحليل وتخزين البيانات الضخمة باستخدام التقنيات التقليدية. تعتمد قوة منصة هادوب على مكونين فرعيين رئيسيين: نظام ملفات هادوب الموزع (HDFS) وإطار المقابلة والاختزال (MapReduce).

1.5 معالجة البيانات الضخمة [7]:

يُعرف MapReduce بأنه نموذج برمجي لمعالجة وتوليد مجموعات كبيرة من البيانات. هناك مرحلتان في MapReduce، مرحلة المقابلة ومرحلة الاختزال. يقوم النظام بتقسيم بيانات الدخل إلى أجزاء متعددة، يتم إسناد كل جزء لمهمة مقابلة التي يمكنها معالجة البيانات بالتوازي. تقرأ كل مهمة مقابلة المدخلات كمجموعة من أزواج (المفتاح، القيمة)، وتنتج مجموعة محولة من أزواج (المفتاح، القيمة) كخرج. يقوم الإطار بخلط وترتيب مخرجات مهام المقابلة، وإرسال الأزواج الوسيطة (المفتاح، القيمة) لمهمة الاختزال، التي تقوم بتجميع النتائج للحصول على النتيجة النهائية كما هو موضح في الشكل (1).



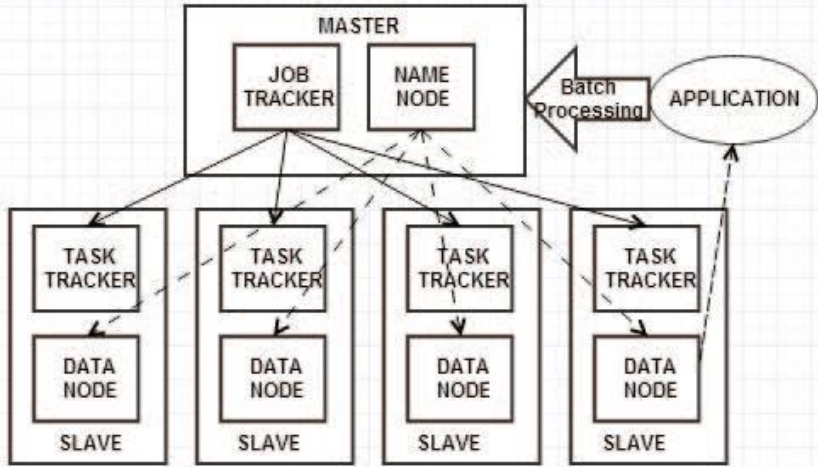
الشكل (1): آلية عمل خوارزمية المقابلة والاختزال [7].

2.5 تخزين البيانات الضخمة [6]:

يعرف HDFS بأنه نظام ملفات لتخزين البيانات يدعم ما يصل إلى مئات العقد في العنقود، ويوفر قدرة تخزين فعالة من حيث التكلفة والموثوقية. يمكنه التعامل مع كل من البيانات المنظمة وغير المنظمة واحتواء أحجام ضخمة من البيانات (أي إن الملفات المخزنة يمكن أن تكون أكبر من تيرابايت).

الميزة الرئيسية لـ HDFS هي قابليته للنقل عبر منصات الأجهزة والبرامج غير المتجانسة. إضافة إلى ذلك، يساعد HDFS على تقليل ازدحام الشبكة وزيادة أداء النظام عن طريق نقل الحسابات بالقرب من تخزين البيانات. كما يضمن تكرار البيانات للتسامح

مع الأخطاء. هذه الميزات تفسر اعتماده على نطاق واسع. HDFS يعتمد على بنية السيد والعبيد بحيث يوزع البيانات الكبيرة عبر عقد المجموعة. في الواقع، تحتوي المجموعة على عقدة (Namenode) واحدة تدير عمليات نظام الملفات والعديد من عقد (DataNodes) التي تدير وتنسيق تخزين البيانات على العقد الحاسوبية الفردية كما هو موضح في الشكل (2).

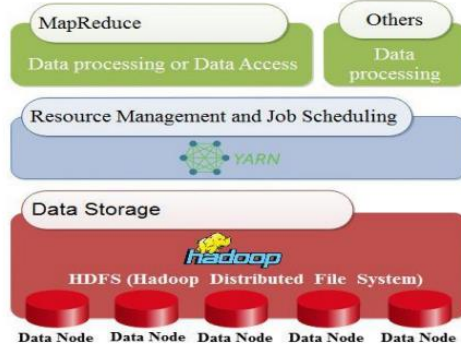


الشكل (2): بنية السيد والعبيد في هادوب [7].

6. تعريف هادوب [13] [6] :

إنَّ أبانتشي هادوب هو إطار مفتوح المصدر من Apache Software Foundation مكتوب بلغة الجافا يعتمد على نحو أساسي على الحوسبة الموزعة ومفاهيم المعالجة المتوازية، ويحتوي على مجموعة من المكونات - HDFS للتخزين و MapReduce لمعالجة البيانات و YARN لإدارة الموارد في العنقود كما هو موضح في الشكل (3). إحدى الميزات الرئيسية ل هادوب هي قدرته على تخزين ومعالجة البيانات الضخمة بسرعة، بفضل امتلاكه العناقيد المتوازية ونظام الملفات الموزع. في الواقع، على عكس التقنيات التقليدية، يركز هادوب على نقل الكود البرمجي إلى مكان البيانات بدلاً من النقل المستمر للبيانات أي أصبحت العمليات الحسابية تتم على البيانات الموجودة على نفس الجهاز لأنَّ نقل البيانات سوف يستغرق وقتاً أطول من إجراء العمليات الحسابية

على البيانات. ومن الميزات الأخرى هي قدرته على تشغيل البرامج مع ضمان التسامح مع الأخطاء، التي عادة ما تتم مواجهتها في البيئة الموزعة. لضمان ذلك، فإنه يمنع فقدان البيانات عن طريق تكرار البيانات على الخوادم. بالإضافة إلى ذلك، يمكن للمستخدمين إضافة وحدات نموذجية فوق هادوب حسب الحاجة وفقاً لأهدافهم بالإضافة إلى متطلبات تطبيقاتهم (على سبيل المثال، القدرة والأداء والموثوقية وقابلية التوسع والأمان). ينتشر مستخدمو هادوب في جميع أنحاء العالم، ومعظمهم من الشركات الكبيرة بما في ذلك eBay و Facebook و Yahoo.

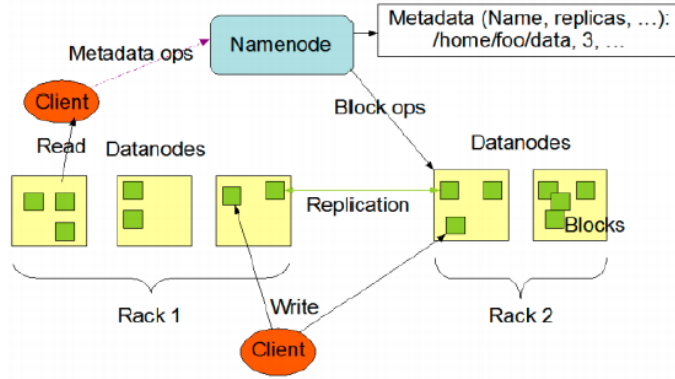


الشكل (3): بنية هادوب [14].

7. نظام ملفات هادوب الموزع HDFS:

هو نظام تخزين مفتوح المصدر متسامح مع الأعطال تم تصميمه للتعامل مع الملفات الكبيرة التي لا يستطيع نظام الملفات التقليدي التعامل معها [15]. يوزع ملفات كبيرة عبر مجموعة تتكون من العديد من عقد البيانات مع التخزين المحلي [13] ويفصل البيانات الوصفية لنظام الملفات عن بيانات التطبيق [11].

إن عقد (NameNode) و (DataNode) هي مكونات برمجية لهادوب يمكن تنفيذها على أجهزة سلعية. تم إنشاء نظام ملفات هادوب الموزع بلغة جافا، ويمكنه تشغيل عقد (NameNode) و (DataNode) على أي جهاز يدعم الجافا [16] والشكل (4) يوضح بنية نظام ملفات هادوب الموزع.



الشكل (4): بنية نظام ملفات هادوب الموزع [16].

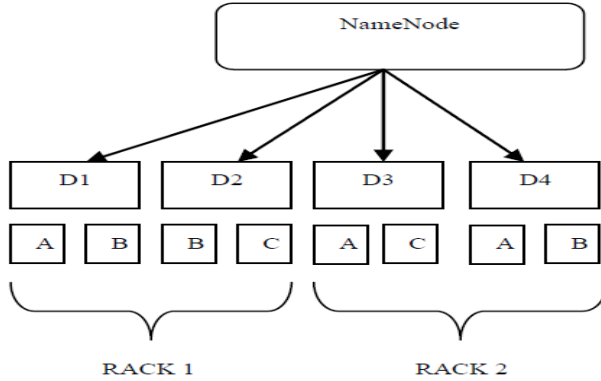
1.7 العقدة الرئيسية **NodeName** [13]:

تقوم عقدة (NameNode) بتقسيم الملف الأصلي إلى مجموعة من الكتل، حجم كل كتلة 64 ميجابايت افتراضياً حيث يتم تكرار كل كتلة بيانات على عقد بيانات مختلفة ثلاث مرات بناءً على قواعد محددة مسبقاً للحفاظ على الإتاحة العالية بحيث يتم تخزين النسخة الثانية ضمن نفس Rack بينما يتم تخزين النسخة الثالثة ضمن Rack آخر كما هو موضح في الشكل (5) باعتبار أن الأحرف A, B, C تمثل كتل بيانات والحرف D متبوع برقم يمثل عقدة البيانات. Rack ليس سوى مجموعة من عقد البيانات. تحتفظ عقدة (NameNode) أيضاً بالبيانات الوصفية لهذه النسخ، وتسجل في ملف EditLog باستمرار كل تغيير يحدث للبيانات الوصفية لنظام الملفات بالإضافة إلى ذلك تسجل عقدة (NameNode) في ملف FsImage معلومات عن كتل البيانات الخاصة بكل ملف ومكان وجود هذه الكتل ومساحات التخزين، وتدير فضاء أسماء النظام الموزع namespace الذي يسجل إنشاء وحذف وتعديل الملفات من قبل المستخدمين، وتدير عمليات نظام الملفات مثل فتح الملفات والمجلدات وإغلاقها وإعادة تسميتها [15].

تتحكم العقدة الرئيسية بوصول العملاء إلى الملفات فعندما يرغب عملاء HDFS في قراءة ملف، فإنهم يتصلون أولاً بالعقدة الرئيسية لتحديد كتل البيانات في الملف ثم يتلقون المعلومات الموجودة في هذه الكتل من عقد البيانات الأقرب إليهم. وفي أثناء كتابة البيانات يطلب من العميل اختيار مجموعة من ثلاث عقد بيانات

لاستضافة النسخ المتماثلة من الكتلة، ثم ينقل العميل البيانات إلى عقد البيانات [16]. باستخدام الإشارات المرسلة دورياً من عقد البيانات تعرف عقدة (NameNode) دائماً أي عقد البيانات لا تزال تعمل على نحو صحيح. إذا كانت الإشارة (heartbeat) مفقودة، فإن عقدة (NameNode) تتعرف إلى فشل عقدة البيانات، وتزيل عقدة البيانات الفاشلة من عنقود هادوب، وتحاول توزيع حمل البيانات بالتساوي عبر عقد البيانات الحالية.

علاوة على ذلك، تضمن عقدة (NameNode) الاحتفاظ دائماً بعدد محدد للنسخ المتماثلة لضمان توافرها بشكل كبير ويسمى عدد النسخ من الملف بعامل النسخ المتماثل replication factor.



الشكل (5): تكرار كتل البيانات على عقد بيانات مختلفة [15].

2.7 عقد البيانات DataNodes [10]:

عقدة البيانات هي نوع من عقد العبيد (Slave Node) في هادوب، التي تستخدم لحفظ البيانات، وهناك منتبع المهام في عقدة البيانات الذي يستخدم لتتبع الوظيفة الجارية على عقدة البيانات والوظائف التي تأتي من عقدة الاسم (NameNode).

تقوم عقدة البيانات بتخزين الكتل والنسخ المتماثلة واسترجاعها. في أثناء بدء التشغيل، تقوم كل عقدة بيانات بالتواصل وإجراء مصافحة مع عقدة الاسم. تتحقق عقدة البيانات من معرف فضاء الأسماء الدقيق، وإذا لم يتم العثور عليه، فإن عقدة بيانات تتوقف عن العمل تلقائياً. يمكن لعقدة بيانات جديدة الانضمام إلى المجموعة بمجرد التسجيل لدى

عقدة الاسم وتسلم معرف فضاء الأسماء. تحتفظ كل عقدة بيانات بتقارير عن الكتل الموجودة لديها. ترسل كل عقدة بيانات تقرير الكتلة الخاص بها إلى عقدة الاسم كل ساعة بحيث يكون لعقدة الاسم دائماً معلومات محدثة عن مكان وجود النسخ المتماثلة في العنقود.

في أثناء التشغيل العادي لنظام ملفات هادوب الموزع ترسل عقدة البيانات أيضاً إشارات نبضات القلب (heartbeat) إلى عقدة الاسم كل عشر دقائق بحيث تعرف عقدة الاسم أي عقد البيانات تعمل بشكل صحيح. إذا لم تتلق عقدة الاسم أي إشارة من عقدة البيانات بعد عشر دقائق، فإن عقدة الاسم تفترض أن عقدة البيانات قد فقدت وتبدأ في إنشاء النسخ المتماثلة من كتل عقدة البيانات المفقودة على عقد البيانات الأخرى.

8. الدراسات السابقة:

في الورقة البحثية [8]، تم اقتراح آلية تستند إلى (HAR) وهو سياسة لأرشفة الملفات تجمع الملفات الصغيرة في ملف أرشيف واحد (.har) [1] [2] [3]، تسمى الآلية المقترحة (NHAR) New Hadoop Archive، التي أعادت تصميم آلية الفهرسة واعتماد فهرسة ذات مستوى واحد مقارنة بالفهرسة ذات المستويين في HAR حيث تقوم التقنية المقترحة بإنشاء جدول تقطيع لمعلومات الفهرسة وتوزيع هذه المعلومات على ملفات الفهرسة، وللوصول لملف ما يتم تنفيذ تابع المقابلة التالي

HashCode (Mod) M حيث HashCode هو ناتج تطبيق تابع التقطيع على اسم الملف و M هو عدد ملفات الفهرسة وإن خرج تابع المقابلة هو رقم ملف الفهرسة الذي يتضمن البيانات الوصفية الخاصة بهذا الملف. إضافة إلى ذلك، تم توسيع نطاق القدرات للسماح بإضافة ملفات جديدة إلى ملفات أرشيف موجود مسبقاً دون إعادة بناء ملف أرشيف جديد وهذا يتضمن ثلاث خطوات أساسية:

1- أرشفة الملفات الجديدة

2- دمج ملفات الفهرسة

3- نقل ملف part الجديد

إن إضافة الكثير من الملفات الصغيرة إلى ملفات أرشيف موجود مسبقاً تؤدي إلى إنشاء

الكثير من ملفات * - part وزيادة استهلاك الذاكرة لتخزين البيانات الوصفية لهذه الملفات لذلك يتم دمج ملفات * - part ليصبح كل part بحجم كتلة HDFS. نظراً لأن HAR و NHAR يستخدمان خوارزمية MapReduce لإنشاء ملف HAR/NHAR، فإن وقت إنشاء الأرشيف متشابه في HAR/NHAR، وبالمقارنة مع HAR، فإن NHAR يستهلك المزيد من الذاكرة ويرجع ذلك إلى استراتيجية الفهرسة في NHAR التي تحتوي على أربعة ملفات فهرسة بدلاً من ملفين فهرسة في HAR. ينتج عن هذا كلفة إضافية إلى العقدة الرئيسية. يمكن لهذا النهج أن يحسن كفاءة الوصول للملفات الصغيرة بشكل كبير حيث يتفوق على HAR بنسبة تصل إلى 85.47% بسبب اعتماده على فهرسة ذات مستوى واحد.

أما في الورقة البحثية [2]، تم تصميم نظام قائم على جداول التقطيع، Hadoop Archive Plus (HAR+) باستخدام sha256 كمفتاح، وهو تعديل على HAR الحالي حيث يستخدم HAR+ فهرسة ذات مستوى واحد مقارنة بالفهرسة ذات المستويين في HAR، وتم تصميم HAR+ لتوفير المزيد من الموثوقية وقابلية التوسع للبيانات الوصفية حيث بدل من استخدام عقدة NameNode واحدة لتخزين البيانات الوصفية، يستخدم HAR+ العديد من عقد NameNode.

إن تابع المقابلة المستخدم هو $M \pmod{M}$ حيث HashCode هو ناتج تطبيق تابع التقطيع على اسم الملف و M هو عدد عقد NameNode وإن خرج تابع المقابلة هو رقم عقدة NameNode. هذا يساعد العميل على الذهاب مباشرة إلى عقدة NameNode المطلوبة في أثناء القراءة التي تحتوي على جداول التقطيع في ذاكرتها والتي بدورها تتضمن البيانات الوصفية الفعلية للملف وهذا يقلل من وقت الوصول للملف.

لضمان الموثوقية والتسامح مع الأعطال تحتوي كل عقدة NameNode على ثلاثة جداول تقطيع. الأول هو الخاص بها واثنان آخران «مستسخان» وهما جدول التقطيع من عقد NameNode اليسرى واليمنى في الشبكة.

تظهر النتائج التجريبية أن HAR+ يقلل من حمل عقدة NameNode واحدة بكمية كبيرة وهذا يجعل العقود أكثر قابلية للتوسع وأكثر قوة وأقل عرضة للفشل على عكس

.HAR

إن HAR+ يظهر تحسن كبير عن HAR من حيث زمن الوصول أما بالنسبة لزمن إنشاء الأرشيف فهو متشابه تقريباً لكنتا الطريقتين. ولكن HAR قد تفوق قليلاً على HAR+، السبب الوحيد في أن HAR+ يستغرق وقتاً أطول قليلاً هو حقيقة أن HAR يستخدم عقدة NameNode واحدة في حين HAR+ يستخدم العديد من عقد NameNode لتخزين البيانات الوصفية.

ويتناول البحث [4] حلاً يعتمد على أعمال Dong وآخرون في الدراسة [12]، وهو نظام ملفات هادوب الموزع الموسع (EHDFS) الذي لديه أربع تقنيات تؤدي دوراً مهماً في تحسين الكفاءة التي يتعامل بها HDFS مع الملفات الصغيرة. إنها دمج الملفات ومقابلة الملفات والجلب المسبق واستخراج الملفات. في هذا النهج، يتم دمج مجموعة من الملفات المترابطة، كما يحددها العميل، في ملف واحد كبير لتقليل عدد الملفات وتقليل البيانات الوصفية للملفات والبيانات الوصفية للكتل التي تحتفظ بها العقدة الرئيسية وتحافظ على البيانات الوصفية للملف المدمج فقط، كما يتم إنشاء آلية فهرسة للوصول إلى الملفات الفردية من الملف المدمج المقابل حيث يتم وضع فهرس في بداية كل كتلة، يحتوي هذا الفهرس على مدخل لكل ملف صغير يمثل جزءاً من هذه الكتلة. كل مدخل هو زوج (إزاحة- طول) يفيد في استخراج محتويات الملف المطلوب من الكتلة. إضافة إلى ذلك، تحافظ العقدة الرئيسية على ترتيب الملفات في الكتلة. هذا يجعل من السهل جلب المسبق للبيانات الوصفية للملفات الصغيرة التالية في نفس الكتلة من العقدة الرئيسية وتخزينها في ذاكرة التخزين المؤقتة للعميل خلال الوصول إلى الملفات في الملف المدمج، هذا يضمن تخفيض عدد الطلبات المرسل إلى العقدة الرئيسية بشكل كبير، وبالتالي تحسين أداء عملية القراءة.

تعمل تقنية مقابلة الملفات عندما يريد المستخدم قراءة ملف صغير من الملف المدمج. حيث يجب على المستخدم أن يرسل طلب إلى العقدة الرئيسية يحدد فيه اسم الملف

الدمج واسم الملف الصغير للحصول على موقع الملف الصغير المطلوب. تحافظ العقدة الرئيسية على بنية معطيات تسمى MapConstituent File لكل ملف مدمج تحتوي على مقابلة بين اسم الملف الصغير ورقم الكتلة المنطقية للملف المدمج الذي يحتوي الملف الصغير ورقم المدخل في الفهرس المخزن في بداية الكتلة والمقابل للملف الصغير وبالتالي تجنب البحث الخطي.

مقارنة بالورقة البحثية التي اقترحها دونغ وآخرون فإن استخدام الذاكرة في EHDfs أعلى بسبب وجود Constituent FileMap لكل ملف مدمج وهذه الزيادة في استخدام الذاكرة يمكن تبريرها بحقيقة أن EHDfs ليس محدداً في المجال ويمكن تطبيقه على أي مجموعة عشوائية من الملفات الصغيرة. كما أنه لا يحتفظ برقم الكتلة المنطقية كجزء من اسم الملف كما اقترح دونغ وآخرون وهذا يسمح للحل أن يكون مستقلاً ويمكن الوصول للملف المدمج دون الاعتماد على اسم الملف.

تشير النتائج التجريبية إلى أنه بالنسبة لـ 100000 ملف صغير فإن EHDfs قادر على تقليل استخدام الذاكرة بنسبة 16%. وتقليل وقت الكتابة بنسبة 82% وتخفيض وقت القراءة بنسبة 28% (مع الجلب المسبق) مقارنة بـ HDFS الأصلي. وكذلك تحسين كفاءة تخزين عدد كبير من الملفات الصغيرة والوصول إليها.

أما الدراسة [9] اقترحت نظاماً لمعالجة الملفات الصغيرة يعتمد على آلية الفهرسة بالاعتماد على الدراستين [4] [12]. قبل تحميل الملف على عنقود نظام ملفات هادوب الموزع، يتم قياس حجم الملف، إذا كان الملف صغيراً، فسيتم فهرسته ومعالجته حيث سيتم إنشاء ملف فهرسة مؤقت لحفظ معلومات الفهرسة (الإزاحة والحجم) للملف الصغير ويتم حساب حجم الملف الصغير المراد دمجه وحجم الملف المدمج الحالي وتقران النتيجة بحجم كتلة بيانات HDFS، إذا كانت أصغر من كتلة بيانات HDFS فيضيف النظام محتويات الملف الصغير المراد دمجه إلى محتويات الملف المدمج، ويضيف

محتويات ملف الفهرسة المؤقت إلى محتويات ملف الفهرسة المدمج. وإذا كانت النتيجة أكبر من كتلة بيانات HDFS يتم حشو الملف المدمج الحالي بملف فارغ، بحيث يصبح بحجم كتلة بيانات كاملة، ويتم إرساله إلى عميل HDFS لتخزينه وينشئ النظام ملفاً مدمجاً جديداً وملف فهرسة مدمج جديد، ويكتب محتويات الملف الصغير المراد دمجها للملف المدمج الجديد ويكتب معلوماته لملف الفهرسة المدمج الجديد.

أخيراً يتم إرسال الملف إلى نظام ملفات هادوب الموزع لتخزينه. أما عند قراءة ملف صغير يرسل عميل HDFS طلباً للعقدة الرئيسية التي تعيد عنوان عقدة البيانات المقابلة ومعلومات كتلة البيانات ذات الصلة إلى عميل HDFS وبما أن هذا النظام يقدم استراتيجية التخزين المؤقت الموزع إضافة إلى آلية الجلب المسبق ذات المستويين من أجل تحسين سرعة قراءة الملفات الصغيرة، يستفسر عميل HDFS عما إذا كان هناك ذاكرة تخزين مؤقتة لعقدة البيانات وفي حال وجودها يتم إرجاع النتيجة إلى العميل وإلا يقرأ عميل HDFS ملف الفهرسة المدمج من الملف المدمج المقابل في عقدة البيانات. يمكنه الحصول على معلومات قيمة لملف الفهرسة المدمج عن طريق معرف الملف الصغير كمفتاح.

نتيجة لتحليل القيمة، يحصل العميل على حجم وإزاحة الملف الصغير، الذي يمكنه من استخراج محتوى الملف الصغير من الملف المدمج وتحمله إلى ذاكرة التخزين المؤقتة لعقدة البيانات.

تظهر النتائج التجريبية أنه مقارنة بنظام HDFS و HAR، فإن هذا النظام يمكنه أن يحسن كفاءة استهلاك موارد الذاكرة للعقدة الرئيسية ويمكنه أن يحسن سرعة القراءة أيضاً. قام الباحثون بالاستفادة من الدراستين [8] و [12] واقترح الدراسة [3] التي تتضمن ملف أرشيف جديد يعتمد على الفهرسة يسمى (HPF) Hadoop Perfect File مصمم

للعناقيد ذات الوصول العالي والمتكرر. نظام الفهرسة يستخدم اثنين من توابع التقطيع. إن تابع التقطيع الأول مسؤول عن توزيع سجلات البيانات الوصفية عبر ملفات الفهرسة التي تخزن كملفات HDFS عادية بحيث يتم تخزين كل ملف فهرسة على عقدة بيانات واحدة ويشغل بحد أقصى كتلة HDFS واحدة لتجنب كلفة الشبكة الإضافية الناتجة عن التنقل بين مواقع عشوائية لملف فهرسة تتوزع كتله عبر عقد بيانات مختلفة. بينما يحفظ تابع التقطيع الثاني موقع سجل البيانات الوصفية لملف صغير داخل ملف الفهرسة مما يوفر الوصول المباشر والسريع إلى البيانات الوصفية من ملف الفهرسة الذي يحتوي على المعلومات بدلاً من تحميل ملف الفهرسة بأكمله في الذاكرة والبحث عن البيانات الوصفية فيه وبالتالي، فإنه يقلل من المعالجة الإضافية وعمليات الدخل والخرج المطلوبة ويحسن كفاءة الوصول لملفات الأرشيف.

يتضمن إنشاء ملف أرشيف HPF أربع خطوات أساسية:

1. خطوة الدمج: يتم تحميل محتوى الملف إلى ذاكرة العميل لتنفيذ عملية الضغط باستخدام خوارزمية LZ4 إذا تم تمكين الضغط، ثم يتم نقل المحتوى من جهاز العميل إلى HDFS لإضافته إلى ملف *part- أكبر ويتم جمع المعلومات اللازمة لاسترداد محتوى كل ملف، والتي تشير إليها باسم سجل البيانات الوصفية ويتضمن السجل (قيمة التجزئة لاسم الملف (file name hash) - رقم part - الإزاحة - الحجم) ويتم إضافته لملف فهرسة مؤقت temporaryIndex_ خلال عملية الدمج لضمان الاسترداد في حالة الفشل ويتم حذف الملف المؤقت عند اكتمال إنشاء ملف أرشيف HPF. يتم تنفيذ الدمج باستخدام مسالك متعددة، يعمل كل منها على ملف *part- منفصل.

2. إنشاء Buckets : عند إنشاء ملف أرشيف HPF جديد يكون هناك Bucket واحد فقط (يمثل ملف فهرسة عند تخزينه على HDFS) وعندما يصل Bucket إلى سعته القصوى، يتم استخدام آلية تقسيم لزيادة عدد Buckets بشكل ديناميكي. عند اكتمال

عملية الدمج، يتم توزيع سجلات البيانات الوصفية على Buckets باستخدام تابع تقطيع موسع بالاعتماد على البتات القليلة الأخيرة من قيمة التجزئة لاسم الملف (file name hash) والتي تسمى global depth بالإضافة إلى مؤشرات على Buckets.

3. حفظ الترتيب: يتم استخدام تابع تقطيع أمثلي (OPMPHF) لكل Bucket وهذا التابع لا يحتوي على فجوات ولا يحتوي على تصادمات ويحفظ الترتيب الذي ستكتب به سجلات البيانات الوصفية في ملف الفهرسة المقابل.

4. الكتابة إلى ملف الفهرسة: أخيراً، تتم كتابة سجلات البيانات الوصفية و OPMPHF في ملف الفهرسة المقابل مع OPMPHF أولاً، تليها سجلات البيانات الوصفية.

5. من أجل جعل كل سجل بيانات وصفية داخل ملف الفهرسة له حجم ثابت، يتم نقل أسماء الملفات التي لا يمكن أن يكون لها حجم ثابت إلى ملف _names.

النتيجة النهائية لإنشاء ملف أرشيف HPF هي مجلد يحتوي على ملفات فهرسة *index- و ملفات *part- وملف _names واحد ، علاوة على ذلك، يسمح HPF للعميل بإضافة المزيد من الملفات الصغيرة بعد إنشاء ملف الأرشيف.

يعد نقل البيانات عبر الشبكة وكتابة كتل البيانات إلى القرص أكثر العمليات استهلاكاً للوقت في HDFS لذلك يدعم HPF افتراضياً سياسة Lazy Persist حيث تتم كتابة البيانات إلى ذاكرة الوصول العشوائي لعقدة البيانات التي بدورها تقوم بكتابتها بشكل غير متزامن إلى القرص الصلب وهذا يحفظ العميل من انتظار كتابة البيانات إلى القرص الصلب.

تظهر النتائج التجريبية أن النظام المقترح يتفوق على أنظمة الملفات الأخرى مثل LHF و HAR و File-Map و HDFS الأصلي من حيث سرعة الوصول للملفات الصغيرة ووقت إنشاء الملف واستهلاك ذاكرة العقدة الرئيسية.

الجديد في هذا البحث هو معالجة مشكلة الملفات الصغيرة في هادوب عن طريق تحسين

إدارة البيانات في نظام الأرشفة (Hadoop perfect file) HPF باستخدام آلية الجلب المسبق والتخزين المؤقت للملفات الصغيرة عند العميل. حيث إن أغلب الدراسات السابقة كانت تعتمد على تحسين إدارة البيانات الوصفية عن طريق الدمج والفهرسة لزيادة فعالية التخزين على حساب كفاءة الوصول، وهذا ما سوف يتم تقديمه في هذا البحث إذ سوف يتم العمل على زيادة فعالية الوصول وتقليل زمن الاستجابة عند العميل.

9. آلية الجلب المسبق المقترحة :

عند قراءة ملف صغير من *part- تقوم آلية الجلب المسبق بقراءة الملفات المرتبطة معه والموجودة بعده ضمن نفس *part- وتخزينها ضمن ذاكرة العميل.

قمنا في هذه التجربة بكتابة الملفين multimedia-0.txt و network-0.txt في

*Part على الترتيب باستخدام HPF، وأرسلنا طلب لقراءة الملف

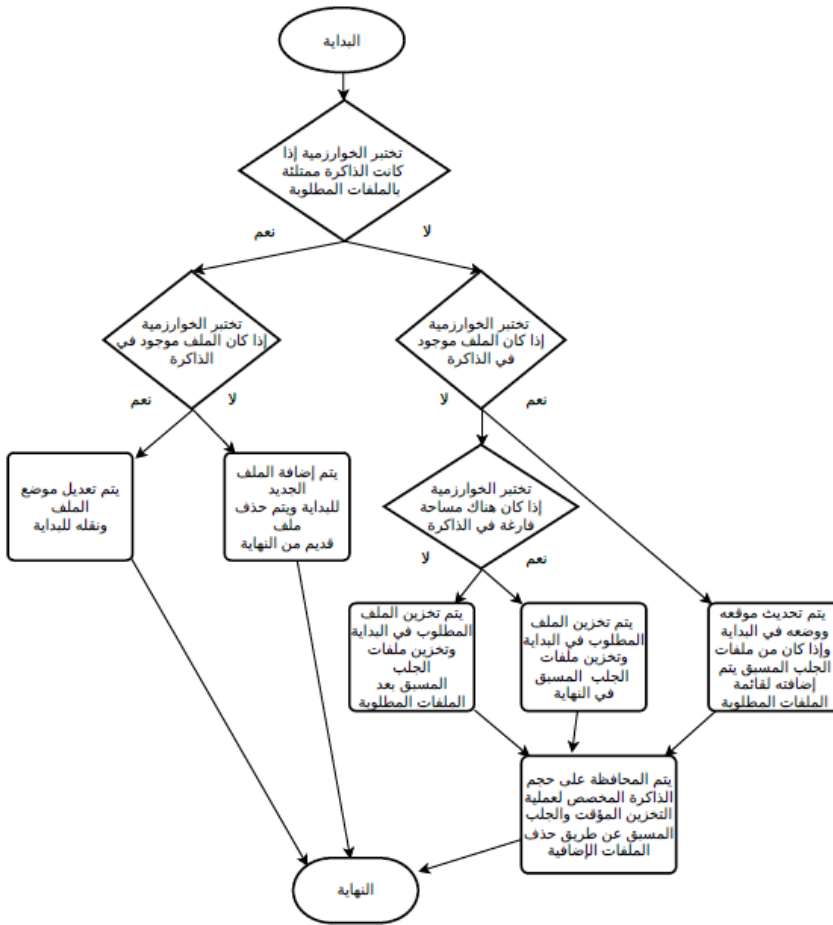
multimedia-0.txt فتم تنفيذ الشرط if وهذا يعني أن الملف تمت قراءته من القرص الصلب، وعند إرسال طلب لقراءة الملف multimedia-0.txt مرة أخرى تم تنفيذ جزء else وهذا يعني أن الملف تمت قراءته من الذاكرة بفضل آلية التخزين المؤقت، وعند إرسال طلب لقراءة الملف network-0.txt تم تنفيذ جزء else أيضاً وهذا يعني أن الملف تمت قراءته من الذاكرة بفصل آلية الجلب المسبق التي تم تنفيذها عند قراءة الملف multimedia-0.txt لأول مرة كما هو موضح في الشكل (6).

```
dataset : 2
Random Access...
Starting the archive files Access test...
->HPF : Access Files
...in if....
prefetching completed
the name is multimedia-0.txt
the content of the file is aviavi
Starting the archive files Access test...
->HPF : Access Files
...in else....
the name is multimedia-0.txt
the content of the file is aviavi
->HPF : Access Files
...in else....
the name is network-0.txt
the content of the file is tcpudp
```

الشكل (6): نتائج تطبيق آلية الجلب المسبق.

1.9 خوارزمية الإضافة والاستبدال المقترحة

إن جزء الذاكرة المخصص لعمليتي التخزين المؤقت وال جلب المسبق محدود بعدد معين من الملفات الصغيرة. تواجه عمليتي التخزين المؤقت وال جلب المسبق مشكلة عند الوصول للعدد الأعظمي من الملفات المسموح تخزينها في هذا الجزء من ذاكرة العميل، لذلك تم تطوير خوارزمية إضافة واستبدال تحافظ على الملفات المطلوبة مؤخراً في البداية كما في الشكل (7).



الشكل (7): آلية عمل خوارزمية الإضافة والاستبدال المقترحة.

قمنا بتجربة حيث تم كتابة (20) ملف باستخدام HPF وتم ضبط حجم جزء ذاكرة العميل المخصص لعمليتي التخزين المؤقت وال جلب المسبق إلى (5) ملفات وتم إرسال طلب لقراءة (5) ملفات ويوضح الشكل (8) نتيجة التجربة.

1- قراءة الملف network-2.txt من القرص الصلب وتنفيذ عملية جلب مسبق لأربع ملفات صغيرة من نفس part وتخزينها في ذاكرة العميل.

2- قراءة الملف multimedia-6.txt من الذاكرة بسبب آلية الجلب المسبق وتعديل مكانه للمحافظة على الملفات المطلوبة مؤخراً في البداية.

3- قراءة الملف multimedia-5.txt من القرص الصلب وإضافته للبداية في الذاكرة وتنفيذ عملية جلب مسبق لملفين فقط من نفس part وتخزينهم في ذاكرة العميل.

4- قراءة الملف network-2.txt من الذاكرة وتعديل مكانه فقط.

5- قراءة الملف network-6.txt من القرص الصلب وإضافته للبداية في الذاكرة وتنفيذ آلية الجلب المسبق لملف واحد فقط من نفس part وتخزينه في ذاكرة العميل.

إن التعقيد الزمني لهذه الخوارزمية كبير بسبب عدد عمليات الجلب المسبق التي يتم تنفيذها لذلك تم تعديل الخوارزمية لتقوم بال جلب المسبق لمرة واحدة فقط مع المحافظة على التخزين المؤقت للملفات المطلوبة مؤخراً.

```

....in if....
prefetching completed
the name is network-2.txt
the content of the file is tcp-2

=====the names is=====
[network-2.txt, multimedia-4.txt, network-4.txt, multimedia-6.txt, network-5.txt]
....in else....
the name is multimedia-6.txt
the content of the file is mp3-6

=====the names is=====
[multimedia-6.txt, network-2.txt, multimedia-4.txt, network-4.txt, network-5.txt]
....in if....
prefetching completed
the name is multimedia-5.txt
the content of the file is mp3-5

=====the names is=====
[multimedia-5.txt, multimedia-6.txt, network-2.txt, network-3.txt, multimedia-7.txt]
....in else....
the name is network-2.txt
the content of the file is tcp-2

=====the names is=====
[network-2.txt, multimedia-5.txt, multimedia-6.txt, network-3.txt, multimedia-7.txt]
....in if....
prefetching completed
the name is network-6.txt
the content of the file is tcp-6

=====the names is=====
[network-6.txt, network-2.txt, multimedia-5.txt, multimedia-6.txt, multimedia-3.txt]

```

الشكل (8): نتائج تطبيق خوارزمية الإضافة والاستبدال المقترحة.

2.9 مشكلة طلب قراءة ملف موجود في نهاية الجزء :

تواجه عملية الجلب المسبق مشكلة عند طلب قراءة ملف موجود في نهاية الجزء، بسبب عدم وجود ملفات إضافية للقيام بقراءتها وتخزينها بشكل مسبق في الذاكرة، وقد تم حل هذه المشكلة عن طريق السماح لآلية الجلب المسبق بقراءة الملفات حتى تصل إلى نهاية الجزء ثم نقل مؤشر القراءة إلى بداية الجزء ومتابعة عملها حتى تصل للملف المطلوب.

عند إرسال طلب لقراءة الملف network-3.txt فتتمت قراءة الملف network-3.txt وقراءة الملفات network-1.txt, network-9.txt وتخزينها مسبقاً في ذاكرة العميل وإكمال عملية الجلب المسبق وقراءة الملفات الموجودة في بداية الجزء حتى الوصول للملف network-3.txt المطلوب كما هو موضح في الشكل (9):

```

Random Access...
Starting the archive files Access test...
->HPF : Access Files
prefetching completed
the name is network-3.txt
=====the names is=====
[network-3.txt, network-1.txt, network-9.txt, network-7.txt, network-2.txt, network-4.txt, network-5.txt, network-6.txt, network-8.txt, network-8.txt]

```

الشكل (9): طريقة عمل آلية الجلب المسبق المعدلة.

10. النتائج ومناقشتها:

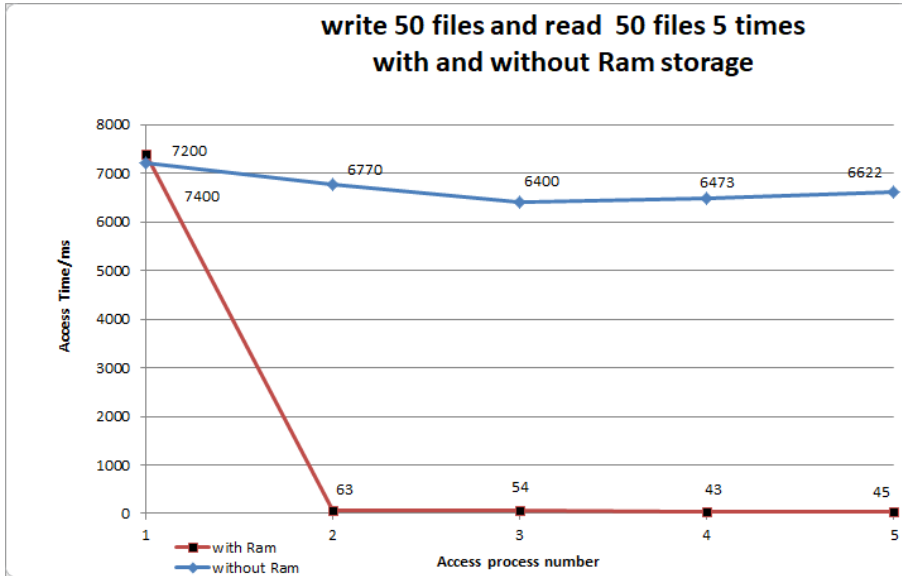
تعرض الجداول (1) و(2) زمن قراءة (25) ملفاً و(50) ملفاً (5) مرات مع استخدام التخزين المؤقت ودون استخدام التخزين المؤقت. تظهر الأشكال (10) و (11) التحسن الكبير في زمن الوصول للملفات الصغيرة عند استخدام التخزين المؤقت للملفات الصغيرة المطلوبة مؤخراً في الذاكرة لأن القراءة من الذاكرة أسرع من القراءة من القرص الصلب.

الجدول (1): زمن قراءة (ميلي ثانية) (25) ملف .

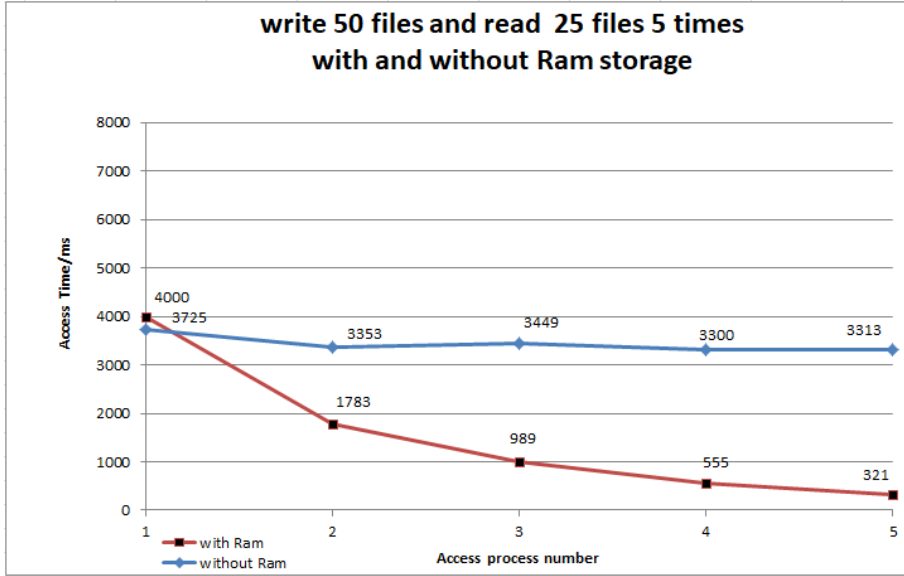
	1	2	3	4	5
with Ram	4000	1783	989	555	321
without Ram	3725	3353	3449	3300	3313

الجدول (2): زمن قراءة (ميلي ثانية) (50) ملف.

	1	2	3	4	5
with Ram	7400	63	54	43	45
without Ram	7200	6770	6400	6473	6622



الشكل (10): مخطط زمن قراءة (50) ملف.



الشكل (11): مخطط زمن قراءة (25) ملف.

في تجربة أخرى قمنا بكتابة (50) ملفاً باستخدام HPF، وقراءة (100) ملف بشكل منتظم (بنفس الترتيب الذي تمت كتابة الملفات فيه)، بحيث تمت قراءتهم بالشكل التالي (50 ملفاً - 25 ملفاً - 25 ملفاً)، وتمت دراسة عدد مرات القراءة من الذاكرة من أجل أحجام مختلفة لجزء ذاكرة العميل المخصص لعمليتي التخزين المؤقت وال جلب المسبق (صفر، 25 ملفاً، 50 ملفاً) ، وتسجيل النتائج في الجدول (3)، بالإضافة إلى دراسة أزيمة الوصول للملفات (ميلي ثانية) وتسجيل النتائج في الجدول (4) وهنا تم دراسة حالتين:

1- حساب زمن الوصول للملفات فقط دون حساب زمن عملية الجلب المسبق للملفات المترابطة ودون حساب زمن كتابة النتائج الإحصائية ضمن ملفات.

2- حساب زمن التجربة بالكامل (زمن الوصول للملفات وزمن عملية الجلب المسبق وزمن كتابة النتائج الإحصائية ضمن ملفات).

يعود السبب في دراسة الحالتين لأن الجلب المسبق يمكن أن يقلل زمن الوصول عند

الاستفادة بشكل كبير من الملفات التي تم تخزينها بشكل مسبق في الذاكرة، ويمكن أن يزيد زمن الوصول عند عدم الاستفادة من الملفات التي تم تخزينها بشكل مسبق في الذاكرة.

نفذنا كل تجربة خمس مرات وتم حساب المتوسط الحسابي للنتائج.

يوضح الشكل (12) أنه بإهمال زمن الجلب المسبق للملفات المترابطة فإن زمن الوصول يتناقص مع زيادة حجم جزء ذاكرة العميل المخصص لعمليتي التخزين المؤقت والجلب المسبق بسبب زيادة عدد مرات القراءة من الذاكرة.

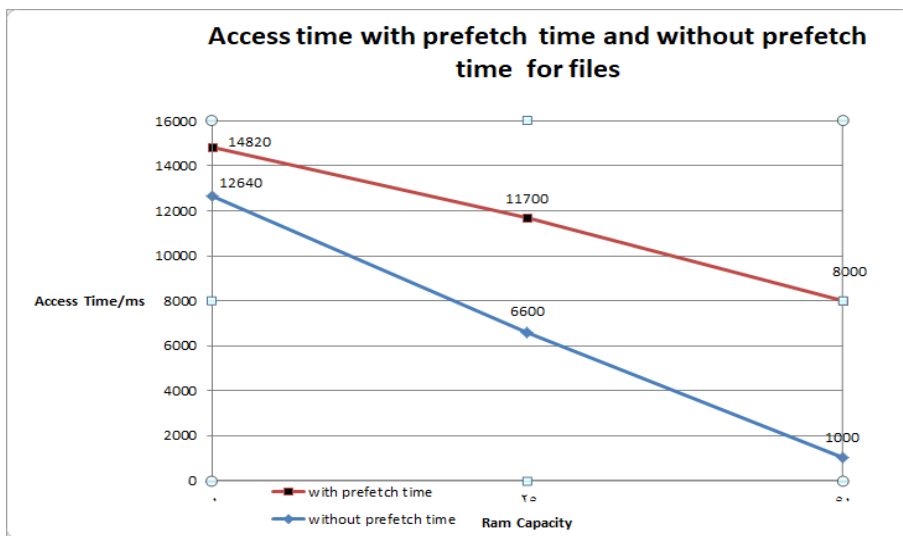
كما يوضح الشكل (12) أيضاً أنه مع الأخذ بعين الاعتبار زمن الجلب المسبق فإن زمن الوصول يتناقص مع زيادة حجم جزء ذاكرة العميل المخصص لعمليتي التخزين المؤقت والجلب المسبق لأن عدد الملفات التي تم جلبها مسبقاً من القرص الصلب أصغر من عدد الملفات التي تمت قراءتها من الذاكرة.

الجدول (3): عدد مرات القراءة من الذاكرة عند قراءة (100) ملف بشكل منتظم.

Ram Capacity	0	25	50
Number of hits	0	49	99

الجدول (4): زمن قراءة (ميلي ثانية) (100) ملف بشكل منتظم

Ram Capacity	0	25	50
with prefetch time	14820	11700	8000
without prefetch time	12640	6600	1000



الشكل (12): مخطط زمن قراءة (100) ملف بشكل منتظم.

ثم قمنا في تجربة أخرى بقراءة (100) ملف بشكل عشوائي وتسجيل النتائج في الجدول (5) والجدول (6).

يوضح الشكل (13) أنه بإهمال زمن الجلب المسبق للملفات المترابطة فإن زمن الوصول يتناقص مع زيادة حجم جزء ذاكرة العميل المخصص لعمليتي التخزين المؤقت والجلب المسبق بسبب زيادة عدد مرات القراءة من الذاكرة.

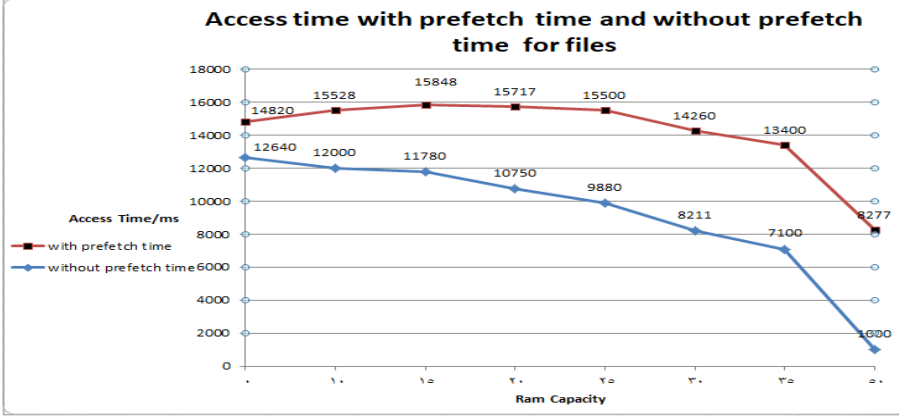
كما يوضح الشكل (13) أيضاً أنه مع الأخذ بعين الاعتبار زمن الجلب المسبق فعندما تكون قيمة Ram Capacity [25-20-15-10] نلاحظ زيادة في زمن الوصول للملفات الصغيرة، لأن عدد الملفات التي تم جلبها مسبقاً من القرص الصلب أكبر من عدد الملفات التي تمت قراءتها من الذاكرة.

الجدول (5): عدد مرات القراءة من الذاكرة عند قراءة (100) ملف بشكل عشوائي.

Ram Capacity	0	10	15	20	25	30	35	50
Number of hits	0	4	6	15	22	35	47	99

الجدول (6): زمن قراءة (ميلي ثانية) (100) ملف بشكل عشوائي.

Ram Capacity	0	10	15	20	25	30	35	50
with prefetch time	14820	15528	15848	15717	15500	14260	13400	8277
without prefetch time	12640	12000	11780	10750	9880	8211	7100	1000



الشكل (13): مخطط زمن قراءة (100) ملف بشكل عشوائي.

وقمنا أيضاً في تجربة أخرى بقراءة (150) ملفاً بشكل عشوائي وتسجيل النتائج في الجدول (7) والجدول (8).

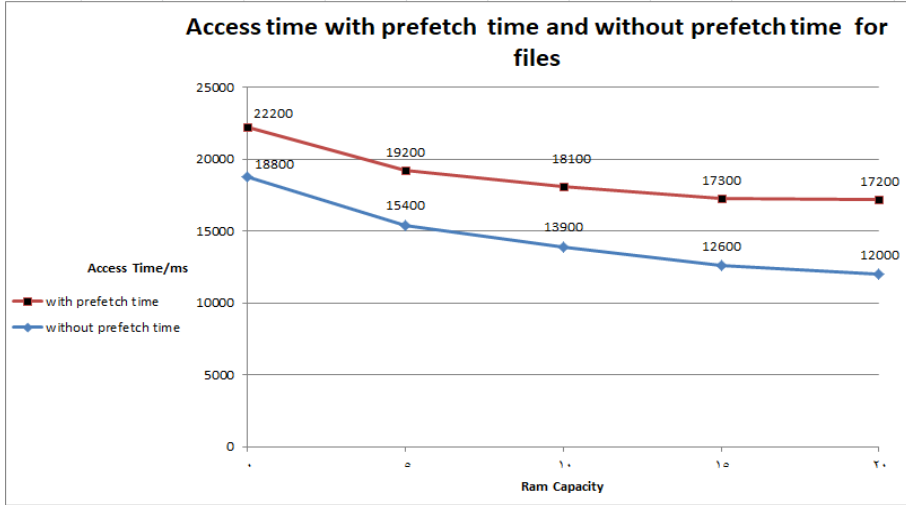
يوضح الشكل (14) تناقص أزمنة الوصول للملفات في كلتا الحالتين (مع جلب مسبق ودون جلب مسبق)، وهذا سببه أنه مع زيادة عدد الملفات المطلوب قراءتها تزداد نسبة الاستفادة من الملفات التي تم جلبها بشكل مسبق (زيادة عدد مرات القراءة من الذاكرة) حتى لو كان حجم جزء ذاكرة العميل المخصص لعمليات التخزين المؤقت والجلب المسبق صغيراً. حيث يمكن ملاحظة أنه في جميع التجارب عدد الملفات المقروءة من الذاكرة أكبر من عدد الملفات التي تم جلبها بشكل مسبق.

الجدول (7): عدد مرات القراءة من الذاكرة عند قراءة (150) ملف بشكل عشوائي.

Ram Capacity	0	5	10	15	20
Number of hits	0	14	29	44	49

الجدول (8): زمن قراءة (ميلي ثانية) (150) ملف بشكل عشوائي.

Ram Capacity	0	5	10	15	20
with prefetch time	22200	19200	18100	17300	17200
without prefetch time	18800	15400	13900	12600	12000



الشكل (14): مخطط زمن قراءة (150) ملف بشكل عشوائي.

11. الاستنتاجات والتوصيات:

قمنا بدراسة عملية وتجريبية لأزمة قراءة ملفات صغيرة مع حساب زمن الجلب المسبق للملفات المترابطة الموجودة ضمن نفس الجزء ودون حساب زمن الجلب المسبق للملفات المترابطة الموجودة ضمن نفس الجزء، وتنوعت التجارب من حيث عدد الملفات المقروءة وطرق قراءتها (بشكل منتظم/بشكل عشوائي) وحجم جزء ذاكرة العميل المخصص لعملية التخزين المؤقت والجلب المسبق مع ملاحظة أن الملفات تم توليدها بأحجام عشوائية وجميع أحجام الملفات أقل من حجم الكتلة الافتراضي (64 ميغابايت) في نظام ملفات هادوب الموزع .

وجدنا أنه بإهمال زمن الجلب المسبق للملفات المترابطة فإن زمن القراءة يتناقص بشكل دائم مع زيادة حجم جزء ذاكرة العميل المخصص لعمليتي التخزين المؤقت والجلب المسبق؛ بسبب تزايد أعداد الملفات التي تتم قراءتها من الذاكرة وتناقص أعداد الملفات التي تتم قراءتها من القرص الصلب، وإن القراءة من الذاكرة أسرع من القراءة من القرص الصلب.

أما عندما يتم حساب زمن الجلب المسبق للملفات الصغيرة فيجب أن نميز حالتين :

الحالة الأولى: إذا كان عدد الملفات التي تم جلبها بشكل مسبق أكبر من عدد الملفات التي تمت قراءتها من الذاكرة فإن زمن القراءة يزداد؛ بسبب قراءة ملفات من القرص الصلب وتخزينها بشكل مسبق في ذاكرة العميل دون الاستفادة منها وقراءتها من الذاكرة. الحالة الثانية: إذا كان عدد الملفات التي تم جلبها بشكل مسبق أقل من عدد الملفات التي تمت قراءتها من الذاكرة فإن زمن القراءة يتناقص؛ بسبب الاستفادة من الملفات التي تم جلبها بشكل مسبق وقراءتها من الذاكرة.

يوصي هذا البحث باستخدام تقنيات التخزين المؤقت والجلب المسبق للملفات الصغيرة في الحواسيب التي تتصف بذاكرة وصول عشوائي ذات حجم كبير، بسبب قدرتها على تخزين عدد كبير من الملفات الصغيرة في ذاكرتها وتحسين سرعة الوصول للملفات الصغيرة، كما يوصي باستخدام هذه التقنيات عند القيام بعملية تحليل ومعالجة لأعداد كبيرة من الملفات الصغيرة، بسبب الحاجة في بعض الأحيان لقراءة كل ملف صغير عدة مرات وبالتالي سوف يسرع استخدام التخزين المؤقت في الذاكرة عملية القراءة حتى لو كان حجم الذاكرة صغيراً. ويوصي هذا البحث بتخصيص جزء لكل نوع معين من الملفات الصغيرة من أجل تحسين أداء ونتائج آلية الجلب المسبق. أخيراً نقترح دراسة ارتباط الملفات مع بعضها قبل تخزينها في الجزء والاستفادة من خوارزميات الذكاء الصناعي في دراسة ارتباط الملفات.

12. قائمة المصطلحات

Apache Hadoop	أباتشي هادوب
HDFS	نظام ملفات هادوب الموزع
Map-Reduce	ماب ريدوس
HAR	أرشيف هادوب
Small Files	الملفات الصغيرة
Big Data	البيانات الضخمة
NameNode	العقدة الرئيسية
DataNode	عقدة البيانات
Map Task	مهمة مقابلة
Reduce Task	مهمة اختزال
Replication Factor	معامل النسخ المتماثل
part	ملف لتخزين المحتوى
HashCode	ترميز التقطيع
EHDFS	نظام ملفات هادوب الموزع الموسع
Bucket	ملف الفهرسة
Cluster	عنقود
OPMPHF	تابع تقطيع أمثلي خال من الفجوات والتصادمات
NHAR	أرشيف هادوب الجديد
Index	ملف فهرسة
Heartbeats	إشارات نبضات القلب

13. المراجع

- [1] <https://hadoop.apache.org/docs/r3.2.4/hadoop-archives/HadoopArchives.html>
- [2] Dev, Dipayan, and Ripon Patgiri. "Har+: Archive and metadata distribution! why not both?." 2015 International Conference on Computer Communication and Informatics (ICCCI). IEEE, 2015.
- [3] Zhai, Yanlong, et al. "Hadoop perfect file: A fast and memory-efficient metadata access archive file to face small files problem in hdfs." Journal of Parallel and Distributed Computing 156 (2021): 119–130.
- [4] Chandrasekar, S., et al. "A novel indexing scheme for efficient handling of small files in hadoop distributed file system." 2013 International Conference on Computer Communication and Informatics. IEEE, 2013.
- [5] Rathidevi, R., and R. Parameswari. "Performance Analysis of small files in HDFS using clustering small files based on centroid algorithm." 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC). IEEE, 2020.
- [6] Oussous, Ahmed, et al. "Big Data technologies: A survey." Journal of King Saud University-Computer and Information Sciences 30.4 (2018): 431–448.
- [7] Singh, Kamalpreet, and Ravinder Kaur. "Hadoop: addressing

challenges of big data." 2014 IEEE International Advance Computing Conference (IACC). IEEE, 2014.

[8] Vorapongkitipun, Chatuporn, and Natawut Nupairoj. "Improving performance of small-file accessing in Hadoop." 2014 11th international joint conference on computer science and software engineering (JCSSE).IEEE,2014.

[9] Cheng, Wenjuan, et al. "Optimizing small file storage process of the HDFS which based on the indexing mechanism." 2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA). IEEE, 2017.

[10] Chavan, Vibhavari, and Rajesh N. Phursule. "Survey paper on big data." Int. J. Comput. Sci. Inf. Technol 5.6 (2014): 7932–7939.

[11] Pan, Shengti. "The performance comparison of hadoop and spark." (2016).

[12] Dong, Bo, et al. "A novel approach to improving the efficiency of storing and accessing small files on hadoop: a case study by powerpoint files." 2010 IEEE International Conference on Services Computing. IEEE, 2010.

[13] Bhathal, Gurjit Singh, and Amardeep Singh. "Big data: Hadoop framework vulnerabilities, security issues and attacks." Array 1 (2019): 100002.

- [14] Alkatheri, Safaa, Samah Anwar Abbas, and Muazzam Ahmed Siddiqui. "A comparative study of big data frameworks." *International Journal of Computer Science and Information Security (IJCSIS)* 17.1 (2019): 66–73.
- [15] Kulkarni, Amogh Pramod, and Mahesh Khandewal. "Survey on Hadoop and Introduction to YARN." (2014).
- [16] Merceedi, Karwan Jameel, and Nareen Abdulla Sabry. "A Comprehensive Survey for Hadoop Distributed File System." *Asian Journal of Research in Computer Science* (2021).