مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (٩) العدد (٣) ٢٠٢٥
Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (9) No. (3) 2025

منهجية محسنة لجدولة المهام في منصات الحوسبة السحابية

م. حسن سلامه*

(تاريخ الإيداع ٢٠٢٤/٦/٤ . قُبِل للنشر في ٢٠٢٥/٢/١٦)

□ ملخّص □

الحوسبة السحابية هي نموذج لتوفير وصول سهل ومرن إلى الموارد الحاسوبية والخدمات عبر الإنترنت ، إذ تؤثر جدولة المهام في الحوسبة السحابية بشكل أساسي ومباشر على استخدام الموارد والتكلفة التشغيلية للنظام السحابي.

هناك العديد من الدراسات التي تناولت مشكلة جدولة المهام في الحوسبة السحابية منها اعتمدت على اقتراح منهجيات جديدة، ومنها استخدم الدمج بين الخوارزميات بحيث يستفاد من مزايا كل خوارزمية بهدف تحسين الأداء. حديثاً استخدمت الخوارزميات الذكية لتحسين كفاءة تنفيذ المهام في الحوسبة السحابية منها الخوارزميات ذاتية الإرشاد metaheuristic algorithms التي أثبتت فعاليتها في إيجاد حلول جدولة مثلى مثل خوارزمية النمل Ant Colony Optimization والنحل Bee Algorithm وخوارزمية بحث الغراب Crow Search Algorithm.

ضمن هذا البحث سيتم تطبيق أحدث خوارزميات ذاتية الإرشاد لجدولة المهام في الحوسبة السحابية، وهي خوارزمية تحسين الحوت Whale Optimization Algorithm، إذ يستخدم نموذج تحسين متعدد الأهداف بهدف تحسين أداء النظام السحابي ، بالإضافة إلى ذلك يقدم هذا البحث تحسيناً لخوارزمية الحوت لتعزيز قدرة البحث عن الحلول المثلى لعملية الجدولة. تبين النتائج التي توصل إليها هذا البحث تحسناً واضحاً في تقليل تكلفة الحمل على النظام، وتقليل زمن التنفيذ، وتقليل السعر، بالتالي تخفيض التكلفة الإجمالية على النظام السحابي بالكامل.

يمكن تطبيق هذا البحث في الأنظمة والمنصات السحابية من أجل تحسين الأداء وتقليل الكلفة الإجمالية في هذه الأنظمة.

الكلمات المفتاحية: الحوسبة السحابية ، جدولة المهام ، الخوارزميات الذكية ، الآلات الافتراضية ، المنصات السحابية.

*حاصل على شهادة ماجستير دراسات عليا في هندسة تكنولوجيا المعلومات- كلية هندسة تكنولوجيا المعلومات والاتصالات – قسم هندسة تكنولوجيا المعلومات - جامعة طرطوس - طرطوس – سورية.

مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (٩) العدد (٣) ٢٠٢٥
Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (9) No. (3) 2025

An Enhanced Methodology for Task Scheduling in Cloud Computing Frameworks

***Eng.Hasan Salamah**

(Received 4/6/2024 . Accepted 16/2/2025)

□ ABSTRACT □

Cloud computing is a model that provides easy and flexible access to computing resources and services over the Internet. Task scheduling in cloud computing significantly and directly affects resource utilization and the operational costs of the cloud system.

There are many studies that have addressed the issue of task scheduling in cloud computing, some of which proposed new methodologies, while others combined algorithms to leverage the advantages of each algorithm to improve performance. Recently, intelligent algorithms have been used to enhance the efficiency of task execution in cloud computing, including metaheuristic algorithms that have proven effective in finding optimal scheduling solutions, such as Ant Colony Optimization, Bee Algorithm, and Crow Search Algorithm. In this research, the latest metaheuristic algorithms for task scheduling in cloud computing will be applied, specifically the Whale Optimization Algorithm (WOA). This algorithm utilizes a multi-objective optimization model aimed at improving the performance of the cloud system. Additionally, this research proposes an enhancement to the Whale Algorithm to improve the search capability for optimal scheduling solutions. The results obtained from this research show a clear improvement in reducing the load cost on the system, minimizing time cost, and consequently lowering the total cost of the entire cloud system.

This research can be applied in cloud systems and platforms to enhance performance and reduce overall costs in these systems.

Key Words: Cloud Computing , Task Scheduling , Intelligent algorithms, Virtual Machines, Cloud platforms.

*Holds a Master's degree in Advanced Studies in Information Technology Engineering- Faculty of Information and Communication Technology Engineering- Department of Information Technology Engineering -University of Tartous – Tartous – Syria.

١. المقدمة والدراسات المرجعية:

مع النمو الكبير في التطبيقات والبيانات والوصول إلى الأنترنت ، أصبحت الحوسبة السحابية أكثر اهتماماً وانتشاراً كونها تقدم العديد من المزايا، مثل المرونة وقابلية التوسع، فضلاً عن تسهيل عملية تقديم الخدمات للمستخدمين، حيث لا يحتاجون إلى امتلاك التكنولوجيا الأساسية، بل يمكنهم الوصول إلى موارد الحوسبة السحابية بطريقة الدفع مقابل الاستخدام Pay-Per-Use ومن ثم تحرير هذه الموارد عند الانتهاء منها[1].

يتم التركيز حديثاً على كيفية إرسال وتوزيع مهام المستخدمين إلى مجموعة الموارد الموجودة في نظام الحوسبة السحابية وهذا ما يعرف بمنهجية جدولة المهام، حيث تمثل العامل الحاسم في جودة عمل هذه الأنظمة، بحيث تعكس بشكل مباشر تحسن الأداء فضلاً عن زيادة الأرباح وعائد الاستثمار لمزودي الخدمة السحابية والمستخدمين أيضاً [2]. قدم الباحثون العديد من الدراسات في محاولات لحل مشكلات الجدولة في الحوسبة السحابية، حيث اقترح باحثون في [3] منهجية جديدة لجدولة المهام في الحوسبة السحابية تسمى SRDQ تجمع بين خوارزمية SJF وخوارزمية الجدولة RR، مع الأخذ بعين الاعتبار قيمة Quantum متغيرة بشكل ديناميكي حيث تعتمد الخوارزمية المقترحة على مفتاحين أساسيين: الأول تحديد قيمة Quantum بشكل ديناميكي لموازنة زمن الانتظار بين المهام ذات الطول القصير والمهام الكبيرة، الثاني: تقسيم رتل الجاهزية Ready Queue إلى رتلين جزئيين (Q1 للمهام القصيرة، Q2 للمهام الكبيرة) حيث يتم تعيين المهام من الرتلين Q1,Q2 إلى الموارد بشكل متبادل مهمتين من Q1 ومهمة واحدة من Q2 والهدف من هذه الدراسة موازنة زمن الانتظار بين المهام ،أثبتت نتائج الدراسة إلى تفوق المنهجية المقترحة في تقليل زمن الانتظار وزمن الاستجابة بالمقارنة مع الخوارزميات SJF,RR,TsPBRR، ولكن سببت مشكلة Starvation بشكل جزئي للمهام الطويلة.

في العام ٢٠١٧ قام الباحثان PeiYun Zhangi و MengChu Zhou بإجراء دراسة [4] بعنوان Dynamic Cloud Task Scheduling Based on a Two-Stage Strategy وكان الهدف من هذه الدراسة تحسين أداء جدولة المهام وتقليل معدل الفشل في عملية الجدولة، حيث تم اقتراح استراتيجية من مرحلتين: المرحلة الأولى يتم استخدام مصنف Bayes لتصنيف المهام بناء على بيانات جدولة المهام التاريخية، وفقاً لذلك يتم إنشاء عدد من الVM من أنواع مختلفة، المرحلة الثانية: تتم مطابقة المهام مع الأجهزة الافتراضية المتاحة، أظهرت النتائج التجريبية لهذه الدراسة أنه يمكن تحسين أداء جدولة المهام السحابية بشكل فعال وتحقيق موازنة الحمل. في العام 2019 قام مجموعة من الباحثين في[5] بإجراء دراسة أكاديمية بعنوان An Enhanced Task Scheduling in Cloud Computing (HSLJF) Based on Hybrid Approach حيث ركزت هذه الدراسة على النظر في الاختلاف في كفاءة الموارد، والتي يمكن أن تؤثر على نتيجة الجدولة النهائية، وكان الهدف من هذه الدراسة تعظيم استخدام الموارد، أشارت النتائج التجريبية لهذه الدراسة إلى تفوق خوارزمية HSLJF في تقليل الMakespan وتقليل زمن الاستجابة مع زيادة استخدام الموارد والإنتاجية بالمقارنة مع الخوارزميات التقليدية SJF,LJF. أيضاً في العام ٢٠١٩ قام الباحثان K. R. Prasanna Kumar وK. Kousalya بالاستفادة من الخوارزميات المستوحاة من الطبيعة لحل مشكلة الجدولة في الحوسبة السحابية من خلال تطبيق خوارزمية بحث

الغراب[6] المستوحاة من سلوكيات جمع الطعام للغراب، حيث يستمر الغراب في مراقبة الغربان الآخرين للعثور على مصدر غذاء أفضل من مصدر الغذاء الحالي بهذه الطريقة تجد خوارزمية بحث الغراب جهازاً افتراضياً مناسباً للمهمة،، قدمت هذه الدراسة تحسين واضح في قيم الMakespan (إجمالي زمن تنفيذ المهام). في العام ٢٠٢٢ قدم مجموعة من الباحثين دراسة [7] بعنوان Task Scheduling in Cloud Computing Using PSO Algorithm حيث تم تقديم طريقة تعتمد على خوارزمية ذكاء الأسراب particle swarm optimization(PSO) لتعيين المهام للأجهزة الافتراضية، أثبتت النتائج التجريبية تفوق النهج المقترح المقدم في هذه الدراسة على الخوارزميات الإرشادية من حيث زمن التنفيذ الكلي.

إن مسألة الجدولة معقدة ودرجة تعقيدها NP–Complete بالتالي لا يمكن حلها بزمن خطي وستعاني خوارزميات الجدولة التقليدية من الفشل مع زيادة حجم المشكلة وأبعادها، لذلك هناك حاجة للتحسين من أجل إيجاد منهجية فعالة لجدولة المهام تحقق تقليل كلفة النظام الإجمالية مع زيادة عدد المهام[8].

تعد خوارزمية تحسين الحوت واحدة من أحدث الخوارزميات المستوحاة من الطبيعة من سلوكيات الصيد والافتراس عند الحوت Humpback Whales، إذ يمكن أن يقدم تطبيق هذه الخوارزمية فائدة نطاق بحث عالمي جيد مما يجعلها خياراً جيداً للمشاكل التي تتطلب استكشافاً واسعاً في الفضاء البحثي واستغلال للحلول المحتملة، إذ يضيف هذا البحث تحسيناً لقدرة البحث عن الحل الأمثل لعملية الجدولة من خلال استخدام معامل ليس خطياً بدلاً من المعامل الخطي، وتطبيق بعض استراتيجيات التكييف الذاتي لحجم المجتمع الخاصة بالخوارزميات ذاتية الإرشاد.

٢. مشكلة البحث:

في السنوات الأخيرة، أدى التغير الكبير في تكنولوجيا الشبكات والأنترنت والتوجه المتزايد نحو اعتماد الأنظمة والمنصات السحابية ودخولها كافة القطاعات خصوصاً مع تزايد التطبيقات والخدمات السحابية، بالتالي زيادة عدد مستخدمي هذه الأنظمة والمهام التي يطلبونها، وهنا تبرز المشكلة الرئيسية خصوصاً أن خوارزميات الجدولة التقليدية والمستخدمة في هذه الأنظمة لم تعد تفي بالغرض، لذا ظهرت الحاجة إلى إيجاد منهجيات أفضل وأكثر فعالية لجدولة المهام في الحوسبة السحابية من ناحية القدرة على العمل مع زيادة أبعاد وحجم المشكلة والمتمثل بزيادة عدد المهام الواردة إلى النظام وتنوعها مع وجود نفس الموارد، بحيث يتم جدولة هذه المهام ومعالجتها مع الحفاظ على حالة النظام السحابي بالكامل، ومراعاة اتفاقية جودة الخدمة بين المستخدم ومزود الخدمة السحابية.

لذلك سيقدم البحث منهجية محسنة لجدولة المهام في الحوسبة السحابية بهدف تحسين أداء هذه الأنظمة.

٣. أهمية البحث وأهدافه:

تلعب جدولة المهام دوراً أساسياً في تحسين معايير جودة الخدمة، مثل تقديم الخدمة للزبون بأقل وقت وتقليل التكلفة التشغيلية للنظام وتحسين كفاءة استخدام الموارد. يهدف هذا البحث إلى تحسين منهجية لجدولة المهام في الحوسبة السحابية بهدف تحسين جودة الخدمة فيها، من خلال المحافظة على حالة

النظام مستقرة دون تدهور الموارد مع زيادة عدد المهام، وتحقيق رضا الزبون من خلال تقديم الخدمات بأسرع وقت وأقل التكاليف.

٤. طرائق البحث ومواده:

تم تنفيذ هذا البحث باستخدام برنامج MATLAB والذي يعد بيئة برمجية هامة وفعالة تستخدم في العديد من المجالات الهندسية والعلوم والرياضيات وتحليل البيانات.

تم كتابة وتنفيذ الأسطر البرمجية والخوارزميات الخاصة بهذا البحث ضمن محرر أكواد m file في برنامج ماتلاب إذ يستخدم m file للتحليل وتطوير وتحسين الخوارزميات ونمذجة المحاكاة ويتيح تنفيذ هذه البرمجيات وتشغيل عملية المحاكاة وعرض الرسوم البيانية الخاصة بها بشكل واضح والمقارنة بين أداء الخوارزميات المختلفة مما يجعله أداة بحثية هامة جداً.

تم استكشاف تطبيق خوارزمية تحسين الحوت Whale Optimization Algorithm في المنصات السحابية، إضافة إلى التحسينات المقترحة لتعزيز قدرة البحث عن الحلول المثلى لعملية الجدولة النهائية، حيث تم إجراء سيناريوهي تقييم: السيناريو الأول تم تجربة وتقييم أداء المنهجية المقترحة في بيئة سحابية نموذجية(عدد مهام وارد صغير)، والسيناريو الثاني تجربة أداء المنهجية المقترحة مع زيادة عدد المهام(زيادة عبء النظام).

٥. جدولة المهام في الحوسبة السحابية:

١,٥.مفهوم جدولة المهام في الحوسبة السحابية

أصبحت جدولة المهام في الحوسبة السحابية من أهم المجالات البحثية التي يوليها الباحثون اهتماماً في السنوات الأخيرة كونها تعد من المشكلات الرئيسية التي تعاني منها أنظمة الحوسبة السحابية، إذ اقترح الباحثون خوارزميات جدولة مختلفة لتحسين وحل هذه المشكلة .

يقصد بجدولة المهام إسناد المهمة الواردة لنظام الحوسبة السحابية إلى المورد الأكثر مناسبة لتنفيذ هذه المهمة، إذ تلعب خوارزمية الجدولة الدور الأساسي الذي يحدد الجهاز الافتراضي Virtual Machine (VM) المناسب لمهمة ما. هناك العديد من الخوارزميات التي تم اقتراحها لحل مشكلات الجدولة صنفت في مجموعات ثلاث وهي الخوارزميات التقليدية[9] مثل خوارزمية Shortest Job First(SJF),Longest Job First(LJF),Round Robin(RR)، والخوارزميات الإرشادية[10] مثل خوارزمية MIN-MIN,MAX-MIN، الخوارزميات ذاتية الإرشاد تم تطبيق الخوارزميات المستوحاة من الطبيعة حديثاً والتي تعمل بشكل أفضل من الخوارزميات التقليدية، مثل خوارزمية النحل والنمل والطيور والخوارزمية الجينية.

٢,٥. نموذج جدولة المهام متعدد الأهداف في الحوسبة السحابية Multi-Objective Task Scheduling Model:

يمكن أن يكون لنظام الحوسبة السحابية متطلبات تنفيذ تخدم النظام من جهة والزبون من جهة أخري، لذا سيستخدم هذا البحث نموذج متعدد الأهداف لجدولة المهام.

لدينا مجموعة من عقد الحوسبة (N)عددها m عقدة كما يأتي: m(Virtual Machines) {N1,N2,N3,…,Nm} ولدينا مجموعة من المهام الواردة(T)عددها n مهمة كما يأتي: n { T1,T2,T3,…,Tn} ، حيث n>m، بالتالي تمثل عملية الجدولة الناتجة بالمصفوفة A_{nm} الموضحة بالعلاقة (١):

$$A_{nm} = \begin{bmatrix} a_{11} & a_{12} \cdots & a_{1m} \\ a_{21} & a_{22} \cdots & a_{2m} \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & a_{nm} \end{bmatrix} \quad \text{العلاقة (١)}$$

حيث a_{ij}: متغير قرار، عندما a_{ij}=1 هذا يعني أن المهمة i يتم تنفيذها على العقدة j [11].

لتوصيف قدرة المعالجة العامة واستهلاك الموارد لنظام الحوسبة السحابية يتم تمثيل كل عقدة باستخدام ثلاث سمات: القدرة على المعالجة processing capability– القدرة على الحمل load capability – عرض الحزمة للمورد resource bandwidth . يمكن نمذجة نظام الحوسبة الأساسي بثلاث متجهات: متجه قدرة المعالجة E_n ومتجه القدرة على الحمل S_n ومتجه عرض الحزمة للمورد C_n وبالمثل يتم استخدام ثلاث متجهات للمهام C_tE_t , S_t .,

من أجل مجموعة من المهام الواردة إلى نظام الحوسبة السحابية لابد لهذا النظام أن يعالج هذه المهام بطريقة عالية الكفاءة من حيث الأداء واستخدام الموارد وتقليل التكلفة أيضاً، يتم تمثيل دالة تكلفة الوقت $f1$ ودالة تكلفة الحمل $f2$ ويتم تمثيل تكلفة الموارد من خلال قياس استهلاك الطاقة والتكلفة الاقتصادية بالتالي سنمثل دالة تكلفة السعر $f3$ وفق العلاقات (٢) و (٣) و (٤) التالية [12]:

$f1 = \sum_{i=1}^{n} \sum_{j=1}^{m} aij\ E_{t,i} / E_{n,j}$ العلاقة (٢)

$f2 = \sum_{i=1}^{n} \sum_{j=1}^{m} aij\ S_{t,i} / S_{n,j}$ العلاقة (٣)

$f3 = \sum_{i=1}^{n} \sum_{j=1}^{m} aij \frac{E_{t,i}}{E_{n,j}} \times \frac{C_{t,i}}{C_{n,j}} \times P$ العلاقة (٤)

- حيث: $E_{t,i}$: قيمة E_t للمهمة i، $E_{n,j}$: قيمة E_n للآلة الافتراضية j، أيضاً بنفس الطريقة يتم التعبير عن البارامترات CS .,

بما أن قيم E_n و S_n و C_n بالإضافة إلى قيم E_t و S_t و C_t في مقاييس مختلفة حسب كل عقدة و كل مهمة ومن أجل جميع المهام والعقد سيسبب هذا الاختلاف انحرافاً في مسار البحث عن الحل الأمثل وسيؤثر هذا الانحراف على عملية التحسين النهائية لذلك سيتم عمل *Normalization* لنحصل على قيم الدوال الثلاث التالية المبينة بالعلاقات (٥) (٦) (٧) على الترتيب[13]:

$f1 = \frac{1}{N} \sum_{i=1}^{n} \sum_{j=1}^{m} a_{i,j} \frac{E_{t,i}/E_{n,j}}{\max_{\forall i,j} \{E_{t,i}/E_{n,j}\}}$ العلاقة (٥)

$f2 = \frac{1}{N} \sum_{i=1}^{n} \sum_{j=1}^{m} a_{i,j} \frac{S_{t,i}/S_{n,j}}{\max_{\forall i,j} \{S_{t,i}/S_{n,j}\}}$ العلاقة (٦)

$f3 = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{m} a_{i,j} \frac{(P\ E_{t,i} C_{t,j})\ /\ (E_{n,j}\ C_{n,j}\ )}{\max_{\forall i,j} \{(P\ E_{t,i} C_{t,j})\ /\ (E_{n,j}\ C_{n,j}\ )\}}$ العلاقة (٧)

الهدف هو إيجاد طريقة لجدولة المهام الواردة تحقق تقليل قيم الدوال الثلاث السابقة مع الأخذ بعين الاعتبار الاختلاف في متطلبات الأنظمة السحابية حيث يمكن أن يكون لنظام سحابي متطلبات تنفيذ مهام بحيث يعطي أهمية لتقليل كلفة الوقت $f1$ بشكل أكبر من تكلفة الموارد $f3$ أو بالعكس، لذلك سيتم إضافة قيم الأوزان weights(wi) للدوال الثلاث للتحكم في درجة التحسين لكل دالة وفق المطلوب من قبل النظام السحابي

وبالتالي تحقيق هدف عملية التحسين النهائية. بناء على ما سبق يتم تمثيل هدف التحسين النهائي بالعلاقة (٨)التالية:

$$F_{opt} = \min\{w1f1 + w2f2 + w3f3\} \quad \text{العلاقة (٨)}$$

في هذا البحث الحالة عامة، لذلك سيتم أخذ قيم الأوزان بشكل متساوي لجميع الدوال (w1=w2=w3=1/3) وبالتالي يتم تمثيل هدف عملية التحسين في مشكلة جدولة المهام في الحوسبة السحابية في هذا البحث وفق العلاقة (٩):

$$F_{opt} = \min\{\tfrac{1}{3}(f1 + f2 + f3)\} \quad \text{العلاقة (٩)}$$

٦. المنهجية المقترحة:

يقدم هذا البحث استكشاف تطبيق أحدث الخوارزميات المستوحاة من الطبيعة وهي خوارزمية تحسين الحوت Whale Optimization Algorithm في جدولة المهام في الحوسبة السحابية مع تقديم اقتراح تحسين لهذه الخوارزمية لتعزيز قدرة البحث فيها والوصول إلى الحل الأمثل لعملية الجدولة .

١,٦. خوارزمية تحسين الحوت Whale Optimization Algorithm:

تعتمد خوارزمية تحسين الحوت على خصائص وسلوكيات الصيد لدى الحيتان Humpback Whales حيث تعد من أحدث الخوارزميات المستوحاة من الطبيعة والتي تستخدم لحل مشاكل التحسين المعقدة. يعد الحوت Humpback Whales حلاً ممكناً في مشكلة التحسين ويسمى Search Agent ، تستخدم خوارزمية الحيتان عدد من وكلاء البحث من أجل تحديد الحل الأمثلي حيث تبدأ عملية البحث بمجموعة من الحلول العشوائية يتم تحديث الحل المرشح بالاعتماد على قواعد التحسين الخاصة بالخوارزمية حتى تحقق شرط النهاية[14]. تستخدم خوارزمية الحيتان ثلاث مراحل أساسية :

١) تطويق الفريسة: في البداية لا تدرك الحيتان الموقع الأمثل بالنسبة للفريسة عندما تكون الفريسة محاطة بالتالي بعد ذلك ستقترب الحيتان وتقوم بتحديث مواقعها بشكل تدريجي يتم تمثيل هذا السلوك بالعلاقتين (١٠) و(١١) على الترتيب:

$$\vec{\mathrm{D}} = |\mathrm{C} \times \vec{X}^*(\mathrm{t}) - \vec{X}(\mathrm{t})| \quad \text{العلاقة (١٠)}$$

$$\vec{\mathrm{X}}(\mathrm{t}+1) = \vec{X}^*(\mathrm{t}) - \mathrm{A} \times \vec{\mathrm{D}} \quad \text{العلاقة (١١)}$$

حيث: $\vec{\mathrm{D}}$: متجه المسافة من وكيل البحث Search Agent إلى الفريسة المستهدفة، $\vec{X}^*$: حل مثالي محلي، $\vec{X}$: متجه الموقع للحوت، t رقم التكرار الحالي، A , C معاملات خاصة بالخوارزمية ويتم تعريف طريقة حسابهم بالعلاقات (١٢) (١٣):

$$C = 2 \times r \quad \text{العلاقة (١٢)}$$

$$A = 2a \times r - a \quad \text{العلاقة (١٣)}$$

حيث: r رقم عشوائي بين 0 و 1، قيمة a قيمة متناقصة من 2 إلى 0 بشكل خطي بناء على رقم التكرار t بالنسبة إلى عدد التكرارات الأعظمية t_{max} كما هو موضح بالعلاقة (١٤):

$$a = 2 - \frac{2t}{t_{max}} \quad \text{العلاقة (١٤)}$$

٢) هجوم شبكة الفقاعات(مرحلة الاستغلال): تطلق الحيتان عدد كبير جداً من الفقاعات حول الفريسة المستهدفة وفق سلوكين: سلوك تقليص التطويق وسلوك التقليص اللولبي، تم نمذجة سلوك تقليص التطويق بالاعتماد على علاقتي تطويق الفريسة في المرحلة الأولى العلاقتان (١٠،١١) ، حيث يمكن ملاحظة أنه يتم تطبيق هذا السلوك عندما |A|<1 وبالتالي وفق

هذا السلوك تتقلص الدائرة المطوقة حول الفريسة تقلصاً تدريجياً، أما سلوك التقليص اللولبي يحسب فيه كل حوت المسافة بينه وبين الحوت المثالي الحالي ومن ثم تتحرك الحيتان في مسار حلزوني الشكل، تم نمذجة سلوك التقليص اللولبي [15] وفق العلاقة (١٥) الآتية:

$$\vec{\mathrm{X}}(\mathrm{t}+1) = \vec{\mathrm{D}}' \times e^{lb} \times \cos(2\pi \mathrm{l}) + \overrightarrow{X^*}(\mathrm{t}) \qquad \text{العلاقة (١٥)}$$

حيث: $\vec{\mathrm{D}}' = |X^*(\mathrm{t}) - \vec{\mathrm{X}}(\mathrm{t})|$: تمثل المسافة من الحوت إلى أفضل حوت حالي، b:ثابت، l: رقم عشوائي بين 1,1-

لمحاكاة السلوكين معاً في التطبيقات العملية للخوارزمية يتم إعطاء قيمة احتمالية p تأخذ قيمة عشوائية بين *0* و *1* تفترض أن احتمال قيام الحوت بتحديث موقعه بناء على سلوك تقليص التطويق وسلوك التقليص اللولبي هو 0.5.

٣) البحث عن الفريسة(مرحلة الاستكشاف): تسلك الحيتان هذا السلوك في محاولة لتوسيع نطاق البحث عن الفريسة والحصول على كمية غذاء أكبر [15]، وبالتالي ضمان الوصول إلى حل مثالي عالمي وليس محلي ضمن فضاء البحث للخوارزمية (عندما |A|>1) ،ويتم وفق هذا السلوك استبدال موقع الحوت Search Agent المثالي الحالي بموقع Search Agent تم اختياره عشوائياً وفق العلاقة (١٦):

$$\vec{\mathrm{X}}(\mathrm{t}+1) = \vec{X}_{rand} - \mathrm{A} \times |\mathrm{C} \times \vec{X}_{rand} - \vec{\mathrm{X}}(\mathrm{t})| \qquad \text{العلاقة (١٦)}$$

حيث: $\vec{X}_{rand}$: متجه الموقع لوكيل البحث الذي تم اختياره عشوائياً. يمكن إسقاط خوارزمية الحيتان على مشكلة جدولة المهام في الحوسبة السحابية وفق الجدول (١):

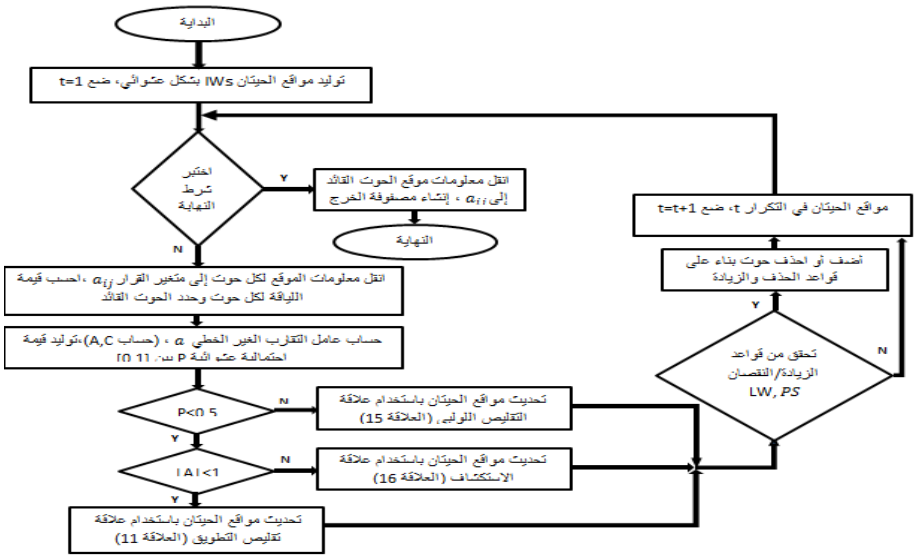
الجدول (١) مقابلة بارامترات خوارزمية الحيتان بمشكلة جدولة المهام في الحوسبة السحابية

الافتراس عند الحيتان	جدولة المهام
الأفراد (الحيتان)	المهام السحابية
عمليات البحث عن الغذاء(الفريسة)	عمليات البحث عن الحل الأمثل
موقع الحوت	حل قيم A_{nm} للدالة F_{opt}
موقع الحوت Leader whale	الحل المثالي الحالي A_{nm} للدالة F_{opt}
قيمة اللياقة للحوت fitness of whale	قيمة F_{opt}

استناداً إلى ما ورد (في الفقرتين ٢,٥ و ١,٦) فإنه:

- يمكن استخدام خوارزمية الحيتان للحصول على حل محسن لمشكلة جدولة المهام السحابية.
- في التكرار t في خوارزمية الحيتان عندما تقوم جميع الحيتان بتحديث مواقعها يمكن نقل معلومات موقع كل حوت إلى حل A_{nm} للمهام المعطاة.
- من أجل القيم الموجودة في المصفوفة A_{nm} يمكننا حساب قيمة اللياقة F_{opt} للحوت وتحديد الحوت القائد Leader Whale ذو القيمة الأصغر لتابع اللياقة.
- يتم استخدام معلومات موقع الحوت القائد لتحديث مواقع الحيتان الأخرى في التكرار التالي من الخوارزمية(عندما يتطلب ذلك).
- يتم تكرار المراحل والعمليات السابقة حتى الوصول إلى التكرار النهائي.
- يتم نقل معلومات موقع الحوت القائد النهائي إلى مصفوفة A_{nm} وسيتم استخدام هذا الحل لإنشاء خطة تنفيذ المهام المحسنة في نظام الحوسبة السحابية.

٢,٦. تحسين قدرة البحث في خوارزمية الحيتان:



الشكل (١) المخطط التدفقي للمنهجية المقترحة

وفقاً لخوارزمية الحيتان السابقة التي تم تناولها في الفقرة (١,٦) ومراحل العمل من خلالها يتم اختيار Search Agent إما بشكل عشوائي (|A|>1)حيث يتم تحديث مواقع الحيتان وفق العلاقة (١٦) أو يتم اختيار أفضل حل (|A|<1) وفق العلاقة (١١) ، حيث يتم استخدام المعامل (a) للتأكد من موقع Search Agent المحدث أنه يقع في المجال $[-a, a]$ وفق العلاقة (١٤) ، نلاحظ أن قيمة (a) تقل خطياً مع زيادة عدد التكرارات هذا الانخفاض الخطي سيسبب المشاكل التالية[16]:

١) تتأثر دقة البحث بالاستكشاف إذا ظهرت القيمة المثلى العالمية في تكرار أولي (قيمة |A|>1).

٢) يمكن أن يكون التقارب بطيئاً بمجرد الاقتراب من القيمة المثلى.

- يقترح هذا البحث استخدام معامل ليس خطياً بدلاً من المعامل الخطي المستخدم من أجل حل المشكلات السابقة والمعبر عنه وفق العلاقة (١٧):

$$a = (1 - \frac{t}{\gamma t_{max}})(1 + \frac{1}{1-\gamma\frac{t}{t_{max}}}) \qquad \text{العلاقة (١٧)}$$

(γ بارامتر أكبر من الصفر)، من خلال اعتماد هذا المعامل سيتم التحكم في عمليات الاستكشاف والاستغلال في خوارزمية الحيتان أي ستزيد قيمة (a) بشكل حاد ثم تنخفض بسرعة إلى قيمة صغيرة وذلك في التكرارات الأولية، ثم تزداد قيمته بمقدار قليل. على الرغم من أن هذا المعامل يزيد من التعقيد الحسابي بمقدار طفيف مقارنة بالمعامل الخطي أثناء مراحل عمل الخوارزمية إلا أنه يؤثر إيجابياً بشكل كبير على فعالية البحث حيث يحقق بحث عالمي فعال في التكرارات الأولية وضمان الوصول إلى قيمة مثالية بسرعة.

حجم المجتمع متغير: سيقوم هذا البحث بتطبيق تعديل على خوارزمية الحيتان من خلال تطبيق بعض استراتيجيات التكييف الذاتي لحجم المجتمع الخاصة بالخوارزميات ذاتية الإرشاد إما بزيادة حجم السكان أو تقليله بناء على حالة أداء الخوارزمية [17] والموضحة بالقواعد التالية:

- القاعدة (١): إذا تم تحديث Leader Whale باستمرار في جيلين و ps > PS_{min} سيتم تنفيذ عامل الحذف لحذف عدد n_{dec} من الحيتان.
- القاعدة (٢): إذا لم يتم تحديث Leader Whale باستمرار في جيل واحد و ps = PS_{max} سيتم تنفيذ عامل الحذف لحذف عدد n_{dec} من الحيتان.
- القاعدة (٣): إذا لم يتم تحديث Leader Whale باستمرار في جيل واحد و ps < PS_{max} سيتم تنفيذ عامل الزيادة لزيادة عدد n_{inc} من الحيتان.

حيث: ps الحجم الحالي للمجتمع، PS_{min} الحد الأدنى لحجم المجتمع، PS_{max} الحد الأعظمي لحجم المجتمع، n_{inc} عدد الأفراد المراد إضافتها، n_{dec} عدد الأفراد المراد حذفها.

معامل الزيادة: يتم تحديد عدد الأفراد المراد إضافتها للمجتمع وفق العلاقة (١٨)، ثم يتم تقسيم المجتمع الحالي إلى n_{inc} فئة باستخدام نهج التجميع العام general clustering approach والموضح في الخوارزمية (١) واختيار الفرد الأمثل في كل فئة لتشكيل مجموعة (S) ستحوي أفضل الأفراد في كل فئة. يتم اختيار أفراد $(x1)$ و$(x2)$ من المجموعة (S) لتوليد أفراد جديدة [18] وفق العلاقة (١٩):

$$n_{inc} = \text{ps} \times (PS_{max} - \text{PS})^2 \times \text{PS}_{max}^{-2} \quad \text{العلاقة (١٨)}$$

$$x_{new} = \alpha^{0.5} x1 + (1 - \alpha^{0.5}) x2 \quad : \alpha \in (0,1) \quad \text{العلاقة (١٩)}$$

Algorithm 1 Whale Clustering

1: Generate a reference point R within the search range

2: Among the current whales P, select a whale X' with the position closest to R

3: In $P \backslash \{X'\}$, find out the points closest to $M-1$ and X' to form a sub-cluster

4: Delete M individual whales in P

5: Repeat 2-4 until the whales have been divided into $N_P \backslash M$ classes.

الخوارزمية (١) الخاصة بنهج التجميع العام

من خلال معامل الزيادة السابق يتم إضافة أفراد جديدة للمجتمع حول القيمة المثلى ، هذا سيحسن من دقة الحل الأمثل الحالي ويعزز قدرة التطوير العامة لخوارزمية الحيتان.

معامل الحذف: مع زيادة عدد التكرارات قدد تظهر بعض الأفراد الزائدة بالتالي يحدد معامل الحذف عدد الأفراد المراد حذفها من المجتمع وفق العلاقة (٢٠)، بعد ذلك يتم تقسيم المجتمع الحالي إلى n_{dec} فئة باستخدام نهج التجميع العام general clustering approach ، ثم يتم حذف أسوء الأفراد في كل فئة[18] .

$$n_{dec} = PS^2 \times (PS_{max} - \text{PS}) \times \text{PS}_{max}^{-2} \quad \text{العلاقة (٢٠)}$$

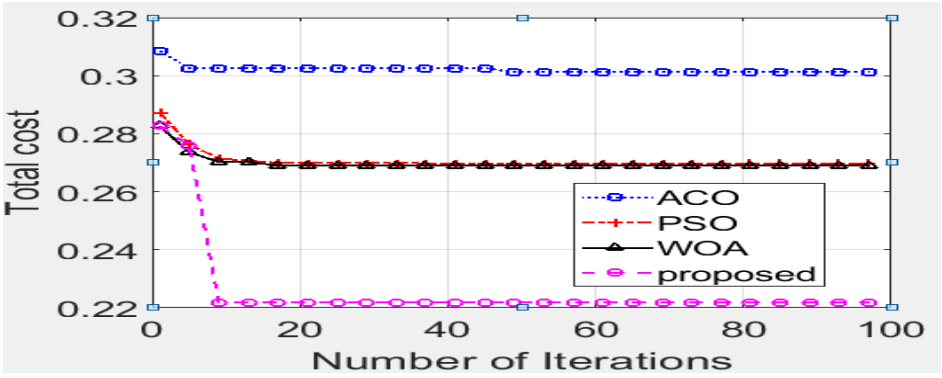
من خلال معامل الحذف سيتم حذف الأفراد بالتساوي من كل فئة وبالتالي سيحافظ استخدام هذا المعامل على تنوع المجتمع مما يعكس على زيادة فعالية البحث عن الحل الأمثل لعملية الجدولة. بالاعتماد على كل ما سبق يمكن تلخيص مراحل عمل المنهجية المقترحة لهذا البحث بالمخطط التدفقي والموضح في الشكل(١).

٧. النتائج والمناقشة:

تم تنفيذ المنهجية المقترحة على برنامج ماتلاب ومقارنتها مع أشهر الخوارزميات الذكية مثل خوارزمية النمل وخوارزمية الأسراب بالإضافة إلى تطبيق خوارزمية الحيتان التي تناولها هذا البحث وملاحظة التحسين المقترح فيها، حيث تم تقييم أداء المنهجية المقترحة في مشكلة جدولة المهام في الحوسبة السحابية من خلال عمليات المحاكاة والمقارنة مع الخوارزميات الأخرى من حيث كلفة الوقت ، كلفة الحمل ، السعر ، التكلفة الإجمالية F_{opt}. يمكن الإشارة إلى أنه تم قياس تكلفة وقت التنفيذ للمهام بدلاً من Makespan بهدف التركيز على قدرة معالجة المهام من جهة النظام.

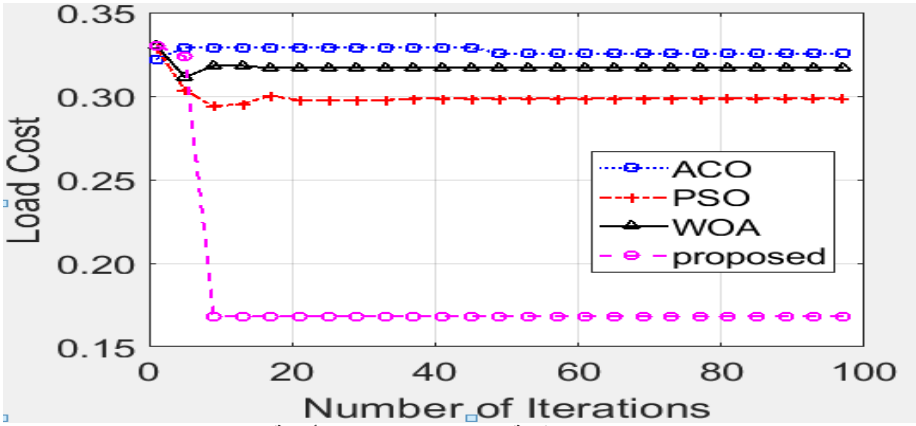
١,٧ السيناريو الأول: تجربة المنهجية المقترحة مع عدد مهام واردة صغير:

يعد هذا السيناريو نموذجياً في الأنظمة السحابية حيث سيتم تحديد المهام الواردة إلى النظام بعدد ١٠٠ مهمة، يوضح الشكل (٢) قيمة التكلفة الإجمالية لكل خوارزمية، إذ يمكن ملاحظة انخفاض قيمتها مع زيادة عدد التكرارات بالنسبة للخوارزميات الأربعة، وهذا يشير إلى الأداء الفعال لجميع الخوارزميات والقدرة العالية للخوارزميات ذاتية الإرشاد على التعامل مع مشكلات الجدولة. تكون التكلفة الإجمالية لكل من PSO,WOA أقل من ACO وهذا يؤكد أن لديهما قدرة أكبر بالبحث عن خطط الجدولة المثلى. يمكن أيضاً ملاحظة أن أداء المنهجية المقترحة هو الأفضل إذ تقلل التكلفة الإجمالية بأكثر من ٢٠%، حيث مع زيادة عدد التكرارات يمكنها الوصول إلى الحل الأمثل لعملية الجدولة والذي يحقق هدف دالة التحسين النهائية بطريقة أسرع.



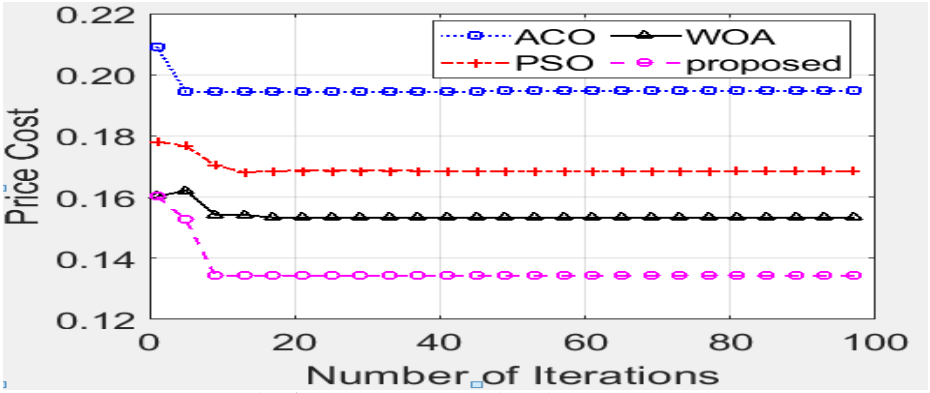
الشكل (٢) قيمة F_{opt} (التكلفة الإجمالية للنظام) في حالة ١٠٠ مهمة

يوضح الشكل (٣) تكلفة الحمل (المعرف وفق العلاقة ٣) على النظام للخوارزميات الأربعة بشكل تفصيلي، إذ يمكن ملاحظة أن الخوارزميات ACO,PSO,WOA تقلل الحمل على النظام السحابي، بالمقابل فإن المنهجية المقترحة يمكنها أن تقلل من تكلفة الحمل على النظام بشكل أكبر عند العدد نفسه من المهام، لذا تحقق سرعة تقارب أفضل من المنهجيات الأخرى في الوصول إلى الحل الأمثل بطريقة أكثر دقة.



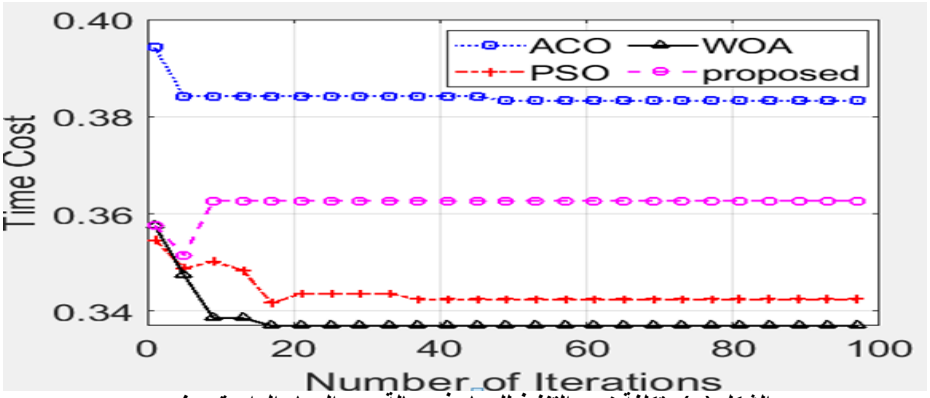
الشكل (٣) قيمة الحمل للخوارزميات الأربعة

يوضح الشكل (٤) تكلفة السعر، إذ يمكن الملاحظة أن الخوارزميات الأربعة تعمل بشكل فعال مع زيادة عدد التكرارات، ولكن يبقى أداء المنهجية المقترحة أكثر فعالية من خلال تقليل السعر بنسبة أكبر وهذا يثبت كفاءة المنهجية المقترحة في تقليل استهلاك الموارد المستخدمة حسب الدالة $f3$ الممثلة بالعلاقة (٧) والتي تعبر عن السعر.



الشكل (٤) قيمة تكلفة السعر للخوارزميات الأربعة

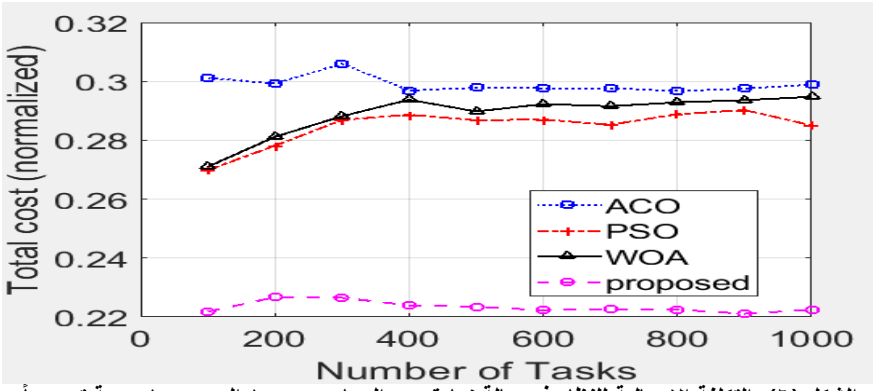
يوضح الشكل(٥) تكلفة زمن التنفيذ. يمكن ملاحظة أن النتائج مختلفة نوعاً ما عن نتائج تكلفة الحمل والسعر السابقة، إذ تنخفض قيم التكلفة لكل من ACO,PSO,WOA وهذا يعني قدرة هذه الخوارزميات على تقليل طاقة وحدة المعالجة المركزية للعقد والتي هي العامل الأساسي في تقليل كلفة زمن تنفيذ المهام، يمكن ملاحظة المخطط البياني للمنهجية المقترحة بأنه ينخفض ومن ثم يزداد حتى يصل إلى حالة استقرار هذا يفسر الأهمية التي توليها المنهجية المقترحة لهدف التحسين النهائي بشكل أكبر على حساب هدف تقليل زمن التنفيذ بمفردها. على الرغم من أن المنهجية المقترحة لها قيم أعلى من PSO,WOA إلا أنها تتمتع بقدرة أعلى على البحث عن الحل الأمثل العالمي والذي يمثل التكلفة الإجمالية للنظام.



الشكل (٥) تكلفة زمن التنفيذ للمهام في حالة عدد المهام الواردة صغير

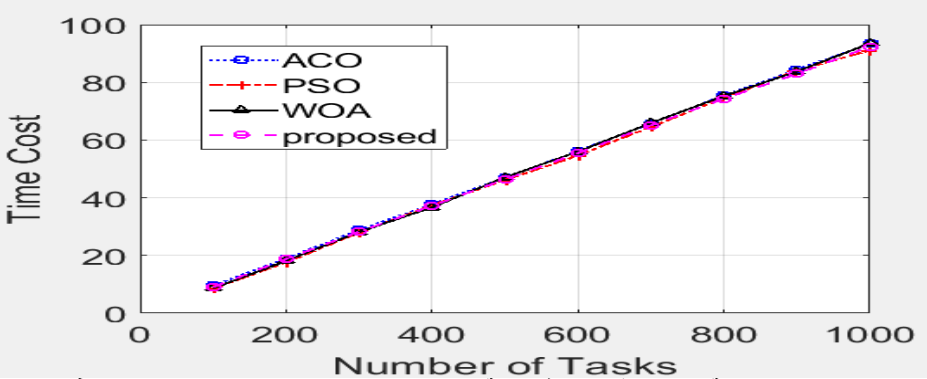
السيناريو الثاني: تجربة أداء المنهجية المقترحة مع زيادة عدد المهام (زيادة عبء النظام):

سيتم وفق هذا السيناريو تقييم أداء الخوارزميات ACO,PSO,WOA مع المنهجية المقترحة في حالة زيادة عدد المهام من ١٠٠ إلى ١٠٠٠ مهمة. يوضح الشكل(٦) مقارنة بين الخوارزميات الأربعة فيما يخص التكلفة الإجمالية للنظام، إذ يمكن ملاحظة أن PSO وWOA يقدمان أداء أفضل من ACO لكن تحقق المنهجية المقترحة الأداء الأفضل بنسبة تتجاوز ٢٢ % وهذا يؤكد قدرة البحث للمنهجية المقترحة في الوصول إلى خطط الجدولة المثالية في حالة زيادة عدد المهام بشكل أكثر كفاءة من الخوارزميات الأخرى.



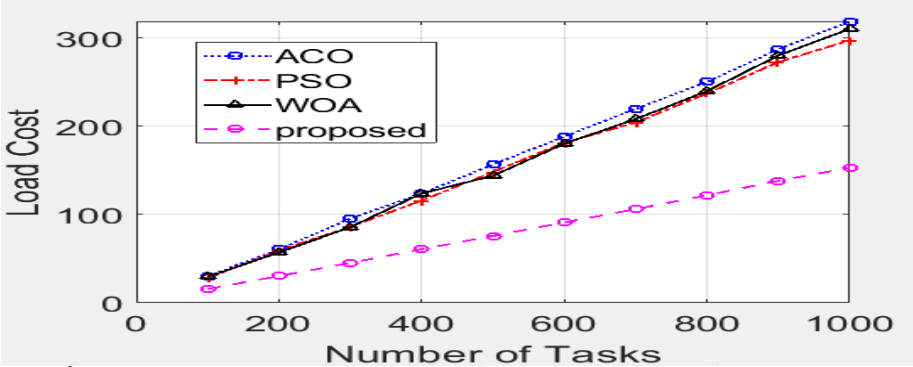
الشكل (٦) التكلفة الإجمالية للنظام في حالة زيادة عدد المهام من ١٠٠ إلى ١٠٠٠ مهمة تدريجياً

يوضح الشكل(٧) تكلفة زمن التنفيذ للمهام حيث يمكن ملاحظة تزايد جميع القيم للخوارزميات الأربعة مع زيادة عدد المهام وهذا أمر متوقع بسبب زيادة عبء النظام، بالإضافة إلى أن جميع القيم لكل الخوارزميات متطابقة تقريباً وهذا يؤكد أن جميع الخوارزميات تستهلك قدرة المعالجة للنظام الأساسي بطريقة متساوية. أيضاً يمكن الملاحظة بأنه عند زيادة عدد المهام بشكل كبير (من ٨٠٠ إلى ١٠٠٠ مهمة) فإن المنهجية المقترحة تقدم تحسناً في تقليل تكلفة زمن التنفيذ مقارنة مع الخوارزميات الثلاث الأخرى.



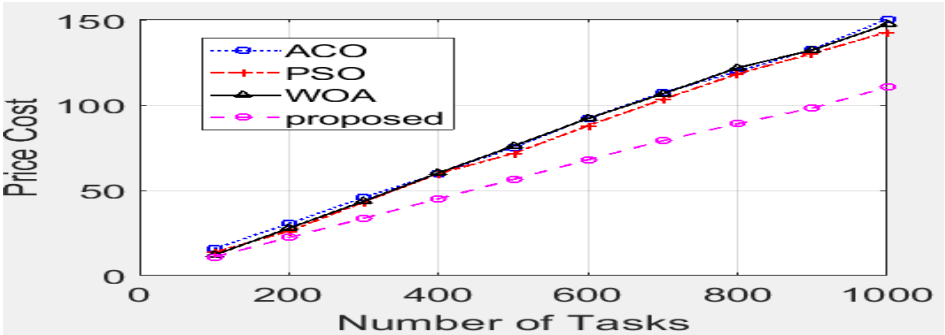
الشكل (٧) تكلفة زمن تنفيذ المهام في حالة زيادة عدد المهام من ١٠٠ إلى ١٠٠٠ تدريجياً

يوضح الشكل (٨) تكلفة الحمل على النظام حيث يمكن ملاحظة أن المنهجية المقترحة تقلل من تكلفة الحمل بشكل كبير، هذا يؤكد قدرة المنهجية المقترحة على استخدام ذاكرة النظام المتاحة بشكل فعال. بالإضافة إلى أنه مع زيادة حمل النظام تحقق المنهجية المقترحة قيماً أقل من الخوارزميات الثلاث ويزداد الفرق في هذه القيم مع زيادة الحمل هذا يؤكد فعالية المنهجية المقترحة في الوصول إلى الحل الأمثل لعملية الجدولة في حالة زيادة عبء النظام.



الشكل (٨) تكلفة الحمل على النظام في حالة زيادة عدد المهام من ١٠٠ إلى ١٠٠٠ تدريجياً

يبين الشكل (٩) تكلفة السعر لكل خوارزمية من الخوارزميات الأربعة حيث يمكن ملاحظة أنه مع زيادة عدد المهام تحقق المنهجية المقترحة أداء أفضل من الخوارزميات الأخرى من حيث تقليل التكلفة التي يدفعها المستخدم. يعكس ذلك كفاءة المنهجية المقترحة في تنفيذ جميع المهام، إذ تقلل من استخدام الموارد بما ينعكس على تقليل التكلفة، بالتالي تحقيق اتفاقية جودة الخدمة (تقديم الخدمة بأسرع وقت وأقل تكلفة) .



الشكل (٩) تكلفة السعر في حالة زيادة عدد المهام من ١٠٠ إلى ١٠٠٠ تدريجياً

٦. الاستنتاجات والتوصيات:

تم في هذا البحث تطبيق أحدث الخوارزميات الذكية وهي خوارزمية الحيتان مع استخدام نموذج متعدد الأهداف بهدف حل مشاكل الجدولة التي تعاني منها الأنظمة السحابية، ومن ثم تم تقديم تحسين مقترح لخوارزمية الحيتان من خلال استبدال معامل التقارب الخاص بالخوارزمية بمعامل تقارب غير خطي وتكييف حجم المجتمع. أثبتت النتائج أن المنهجية المقترحة تحقق أداء أفضل من الخوارزميات الذكية الأخرى فضلاً عن تحسين قدرة البحث عن الحلول المثلى في عملية الجدولة، بالتالي تقليل استخدام الموارد واستهلاك ذاكرة النظام والتكلفة الكبيرة التي يدفعها المستخدم مقابل الخدمات السحابية المطلوبة بنسبة تحسين تفوق ٢٠%.

في الأعمال المستقبلية لهذا البحث يمكن تحسين جدولة المهام في الأنظمة السحابية من خلال النظر في المهام المرتبطة مع بعضها والنظر في أولوية المهام فضلاً عن إدخال أدوات التعلم العميق وربطها مع المنهجية المقترحة لدينا بهدف زيادة دقة الوصول إلى الحل الأمثل لعملية الجدولة.

٧. قائمة المراجع :

[1] Miyuki sato creating next generation cloud computing based network services and the contribution of social cloud operation support system (OSS) to society . *IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*.٢٠١٥ .

[2] Vijindra and Sudhir Shenai Survey of Scheduling Issues in Cloud Computing . .٢٠١٥

[3] S., Sarhan, S & .Joundy, M Elmougy .A novel hybrid of Shortest job first and round Robin with dynamic variable quantum time task scheduling technique .*J Cloud Comp*.٢٠١٩ . ١٢،٦

[4] P. Zhang and M. Zhou . Dynamic Cloud Task Scheduling Based on a Two-Stage Strategy .*in IEEE Transactions on Automation Science and Engineering, vol. 15, no. 2, pp. 772-783*.٢٠١٨ .

[5] Dhari A., El-Booz S.A., Nasr A.A., Arpitha A., Mallappa S Alworafi M.A . An Enhanced Task Scheduling in Cloud Computing Based on Hybrid

Approach .*In: Nagabhushan P., Guru D., Shekar B., Kumar Y. (eds) Data Analytics and Learning. Lecture Notes in Networks and Systems, vol 43. Springer, Singapore*.٢٠٢٠ .

[6] K.R., Kousalya, K Prasanna Kumar . Amelioration of task scheduling in cloud computing using crow search algorithm .*Neural Comput & Applic*.٢٠٢٠ . ٥٩٠٧–٥٩٠١ ،٣٢

[7] M. Nagaratna, Lakshmi Harika Marrivada Sriperambuduri Vinay Kumar . Task Scheduling in Cloud Computing Using PSO Algorithm .*Smart Intelligent Computing and Applications, Volume 1*٢٠٢٢ ..

[8] M. S. Sanaj and P. M. J. Prathap An efficient approach to the map-reduce framework and genetic algorithm based whale optimization algorithm for task scheduling in cloud computing environment .*Materials Today: Proceedings*.٢٠٢١ .

[9] Kittusamy, Kousalya & K R, Prasanna Kumar. (2016). QoS based Task Rescheduling in Computational Grid Environment. Asian Journal of Research in Social Sciences and Humanities. 6. 1975. 10.5958/2249-7315.2016.00341.5.

[10] El-Gamal, Ahmed & Mostafa, Reham & Hikal, Noha. (2020). Load Balancing Enhanced Technique for Static Task Scheduling in Cloud Computing Environments. 10.1007/978-981-15-3075-3_28.

[11] L. Zuo, L. Shu, S. Dong, C. Zhu, and T. Hara, "A multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing," IEEE Access, vol. 3, pp. 2687–2699, 2015.

[12] Y. Hu, C. De Laat, Z. Zhao et al., "Multi-objective container deployment on heterogeneous clusters," in Proc. 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2019, pp. 592–599.

[13] H. Aziza and S. Krichen, "Bi-objective decision support system for taskscheduling based on genetic algorithm in cloud computing," Computing, vol. 100, no. 2, pp. 65–91, 2018.

[14] Andrew Lewis Seyedali Mirjalili . The Whale Optimization Algorithm,Advances in Engineering Software, pp. 51-67, 2016, 18.

[15] Analysis of logistic growth models . *Mathematical biosciences, vol. 179 no. 1*.٢٠٠٢ .

[16] Qu, S., Liu, H., Xu, Y. *et al.* Application of spiral enhanced whale optimization algorithm in solving optimization problems. *Sci Rep* **14**, 24534 (2024). https://doi.org/10.1038/s41598-024-74881-9.

[17] T.-Y. Sun, C.-C. Liuand S.-J. Tsai, S.-T. Hsieh .olving large scale global optimization using improved particle swarm optimizer .*in Proc.2008 IEEE Congress on Evolutionary Computation* ،٢٠٠٨ .pp. 1777-1784.

[18] A. Tsoularis and J. Wallace . Analysis of logistic growth models . *Mathematical biosciences, vol. 179 no. 1.2002. pp. 21-55.*