UNIVERSITY OF MOSUL
COLLEGE OF COMPUTER SCIENCES
AND MATHEMATICS

# Improving Parallel Schedulers in Heterogeneous Distributed Computing for Containerized Big Data

**Balqees Talal Hasan Agha**

Ph.D. Thesis in

Computer Sciences

Supervised By

**Prof. Dr. Dhuha Basheer Abdullah Albazaz**

**2022 A.D** **1443 A.H**

# ABSTRACT

In today's digital world, large amounts of data are generated which cannot be processed or managed using traditional computing systems. This has resulted in the emergence of a number of high-performance and distributed computing frameworks. Apache Spark is a computing framework which is distributed in its nature for big data processing. Spark uses a cluster of machines to process data in parallel in order to complete its task. Spark must manage the cluster of machines while working in a distributed fashion in order to produce the desired results. A cluster manager assists Spark with this management and arranges the resources needed to run the applications. Kubernetes is a new cluster manager that allows Spark to run in containerized environment. However, when making scheduling decisions, the Kubernetes default scheduler does not take into account the heterogeneous nature of computing nodes when scheduling and deploying different workloads in a distributed computing environment, nor does it take into account real-time usage of computing nodes.

In this dissertation a new system has been proposed and implemented which is named as Hetero-Kubernetes. Hetero-Kubernetes categorizes computational nodes in a cluster into two sub-clusters, each managed by a separate special scheduler, allowing the system to distribute and parallelize the scheduling process between the two schedulers, Kube-Spark and Kube-EDIoT which are programmed using Go Language.

Kube-spark manages the first sub-cluster which consists of traditional computers to schedule big data components on it using FCFS scheduling algorithm whereas Kube-EDIoT manages the second sub-

cluster which consists of Raspberry pi devices to schedule IoT applications on using a new proposed HRRMLQ scheduling algorithm.

Hetero-kubernetes integrates two critical components to makes more efficient scheduling decisions : Real-time computing resource monitoring and scheduling. Kube-Spark and Kube-EDIoT rely on a new proposed resource management algorithm which is named as " capacity-aware scheduling algorithm". Experiments show that the Hetero-Kubernetes system outperforms the default Kubernetes system because the Hetero-Kubernetes schedulers always choose the node with the lowest resource usage as well as Kube-EDIoT scheduler can schedule high priority pods first while also reducing the starvation problem that multilevel queues may experience. Furthermore, Hetero-Kubernetes approves its ability to schedule pods on nodes based on real-time resource usage metrics, which in turn load balance among cluster nodes.

جامعة الموصل
كلية علوم الحاسوب والرياضيات

# تحسين الجدولة المتوازية في الحسابات الموزعة غير المتجانسة لحاويات البيانات الضخمة

بلقيس طلال حسن أغا

أطروحة دكتوراه

علوم الحاسوب

بإشراف

الأستاذ الدكتورة ضحى بشير عبدالله البزاز

1443هـ 2022م

# الخلاصة

في العالم الرقمي اليوم ، يتم إنشاء كميات كبيرة من البيانات، والتي لا يمكن معالجتها أو إدارتها باستخدام أنظمة الحوسبة التقليدية. وقد أدى ذلك إلى ظهور عدد من أطر الحوسبة الموزعة التي تتسم بكونها عالية الأداء. Apache Spark هو إطار عمل حوسبة موزع لمعالجة البيانات الضخمة. يستخدم Spark مجموعة من عقد الحوسبة لمعالجة البيانات بشكل متوازٍ من أجل إكمال مهامه المطلوبة. يجب على Spark إدارة مجموعة عقد الحوسبة أثناء العمل بطريقة موزعة من أجل تحقيق النتائج المرجوة. يساعد مدير الحوسبة العنقودية Spark في هذه الإدارة لتشغيل التطبيقات. Kubernetes هو مدير حوسبة عنقودية جديد يسمح لـ Spark بالعمل في بيئة الحاويات. ومع ذلك ، عند اتخاذ قرارات الجدولة ، لا يأخذ مجدول Kubernetes الافتراضي في الاعتبار الطبيعة غير المتجانسة لعقد الحوسبة عند جدولة ونشر أحمال عمل مختلفة في بيئة حوسبة موزعة ، ولا يأخذ في الاعتبار الاستخدام في الوقت الفعلي لعقد الحوسبة.

في هذه الاطروحة تم اقتراح وتنفيذ نظام جديد يسمى Hetero-Kubernetes. يصنف Hetero-Kubernetes عقد الحوسبة في النظام الموزع إلى مجموعتين فرعيتين ، تدار كل منهما بواسطة مجدول خاص منفصل ، مما يسمح للنظام بتوزيع وموازاة عملية الجدولة بين المجدولين ، Kube-Spark و Kube-EDIoT واللذين تمت برمجتها باستخدام Go لغة.

يدير Kube-spark المجموعة الفرعية الأولى التي تتكون من أجهزة كمبيوتر تقليدية لجدولة مكونات البيانات الضخمة عليها باستخدام خوارزمية جدولة FCFS بينما يدير Kube-EDIoT المجموعة الفرعية الثانية التي تتكون من أجهزة Raspberry pi لجدولة تطبيقات إنترنت الأشياء عليها باستخدام خوارزمية مقترحة جديده سميت HRRMLQ .

يدمج Hetero-kubernetes مكونان مهمان لاتخاذ قرارات جدولة أكثر كفاءة: مراقبة موارد الحوسبة في الوقت الفعلي والجدولة. يعتمد كل من Kube-Spark و Kube-EDIoT على خوارزمية إدارة الموارد المقترحة الجديدة والتي تسمى "خوارزمية الجدولة المدركة للقدرات". تُظهر التجارب أن نظام Hetero-Kubernetes يتفوق على نظام Kubernetes الافتراضي لأن مجدولي Hetero-Kubernetes يختارون دائمًا العقدة ذات أقل استخدام للموارد بالإضافة إلى Kube-EDIoT يمكنه جدولة البودات ذات الأولوية العالية أولاً مع تقليل مشكلة الجوع التي قد تواجهها قوائم الانتظار متعددة المستويات. علاوة على ذلك ، اثبت