Journal of
Marine Science and Engineering

MDPI

Article

Optimization of Maritime Communication Workflow Execution with a Task-Oriented Scheduling Framework in Cloud Computing

Zulfiqar Ahmad [1], Tayfun Acarer [2] and Wooseong Kim [3,*]

[1] Department of Computer Science and Information Technology, Hazara University, Mansehra 21300, Pakistan; zulfiqarahmad@hu.edu.pk

[2] Maritime Transportation and Management Vocational School of Higher Education, Piri Reis University, Tuzla, İstanbul 34940, Türkiye; tacarer@hotmail.com

[3] Department of Computer Engineering, Gachon University, Seongnam-si 13120, Republic of Korea

* Correspondence: wooseong@gachon.ac.kr

Abstract: To ensure safe, effective, and efficient marine operations, the optimization of maritime communication workflows with a task-oriented scheduling framework is of the utmost importance. Navigation, vessel traffic management, emergency response, and cargo operations are all made possible by maritime communication, which necessitates seamless information sharing between ships, ports, coast guards, and regulatory bodies. However, traditional communication methods face challenges in adapting to the dynamic and distributed nature of maritime activities. This study suggests a novel approach for overcoming these difficulties that combines task-oriented scheduling and resource-aware cloud environments to enhance marine communication operations. Utilizing cloud computing offers a scalable, adaptable infrastructure that can manage various computational and communication needs. Even during busy times, effective data processing, improved decision making, and improved communication are made possible by utilizing the cloud. The intelligent allocation and prioritization of communication activities using a task-oriented scheduling framework ensures that urgent messages receive prompt attention while maximizing resource utilization. The proposed approach attempts to improve marine communication workflows' task prioritization, scalability, and resource optimization. In order to show the effectiveness of the proposed approach, simulations were performed in CloudSim. The performance evaluation parameters, i.e., throughput, latency, execution cost, and energy consumption, have been evaluated. Simulation results reflect the efficacy and practical usability of the framework in various maritime communication configurations. By making marine communication methods more durable, dependable, and adaptable to the changing needs of the maritime industry, this study advances maritime communication techniques. The findings of this research have the potential to revolutionize maritime communication, leading to safer, more efficient, and more resilient maritime operations on a large scale.

Keywords: data collection and processing; task execution; cloud computing; latency; vessel traffic management; maritime Strategy; maritime business management; maritime communication; optimization; scheduling; workflows

Citation: Ahmad, Z.; Acarer, T.; Kim, W. Optimization of Maritime Communication Workflow Execution with a Task-Oriented Scheduling Framework in Cloud Computing. *J. Mar. Sci. Eng.* **2023**, *11*, 2133. https://doi.org/10.3390/jmse11112133

Academic Editors: Hongpeng Zhang, Xingming Zhang and Lin Zeng

Received: 8 October 2023
Revised: 3 November 2023
Accepted: 7 November 2023
Published: 8 November 2023

1. Introduction

Marine transportation by ship provides a very cheap and effective solution for supply chains, especially at an international scale. While it provides so many advantages, ship accidents may have drastic consequences; thus, a crucial problem arises [1,2]. The purpose of the International Maritime Organization (IMO) is to reduce the number of ship accidents, which result in both economic and human costs [3]. Considering the extreme consequences of ship accidents, maritime navigation emerges as a key problem. It is extremely important for maritime navigation that sea vehicles provide easy, reliable, and consistent communication with each other and with the relevant institutions on land. In the meantime, it is of great importance that the private meetings of the seafarers in the sea vessels are carried out in a healthy manner in terms of the management and administration of the sea vessel [4]. Consequently, marine communications are crucial for maritime navigation. Today, marine communication is carried out on two different platforms, namely on satellite and terrestrial communication systems. The devices used in these systems, which are different from each other in terms of technical infrastructure, also have different features in terms of usage and communication possibilities [5].

The interchange of sensitive information between ships, ports, coast guards, shipping corporations, and regulatory bodies is a significant component of contemporary shipping and maritime operations [1]. For controlling traffic in congested waterways, responding to crises, and coordinating cargo operations, effective and prompt communication is paramount [2]. Traditional communication workflows, however, find it difficult to keep up with the changing demands of the business due to the complexity and volume of maritime activities [6]. The combination of cloud computing and task-oriented scheduling frameworks offers a possible solution to these problems and enhances the performance of marine communication. The diversified and distributed nature of marine communication operations may be handled by the flexible and scalable infrastructure provided by cloud computing [7–10]. Task-oriented scheduling frameworks allow for intelligent task allocation and optimization, guaranteeing that communication activities are carried out effectively in accordance with their priorities, dependencies, and computing needs [11,12]. The ability to seamlessly coordinate a variety of maritime activities is made possible through maritime communication, which is the lifeblood of the shipping industry. Vessel traffic management systems, which share location, course, and speed information between vessels to prevent collisions and navigate securely through congested trade lanes, are one practical application of marine communication. In order to organize search and rescue activities quickly, emergency response operations significantly rely on excellent communication between vessels, coast guards, and rescue centers. The efficient handling of cargo and timely berthing and departure ensured by effective communication between ships and ports improves port efficiency overall and shortens turnaround times [13–15].

In many industries, cloud computing has become a game-changer, and marine communication is no exception. The scalable and adaptable infrastructure provided by cloud-based solutions can handle the rising data volume and computing demands of maritime communication operations [16]. Maritime stakeholders can analyze massive amounts of data, improve decision-making processes, and guarantee smooth communication even during peak times by outsourcing computing and storage demands to the cloud. Cloud computing makes it possible for dynamic provisioning and cost-effective resource utilization, which makes it the perfect platform for streamlining maritime communication procedures [13,17,18]. Although cloud computing has many advantages, streamlining marine communication operations presents particular difficulties. The maritime sector operates in a variety of remote locations with patchy or nonexistent network access. For processes that are safety-critical, ensuring real-time communication and data synchronization becomes essential. To protect important maritime data and communication channels, strong security measures are also necessary due to the constantly changing threat scenario. To manage the dynamic nature of maritime operations, where communication requirements differ dramatically depending on elements such as weather conditions, traffic volume, and port activities, scalability and adaptability are essential [13,14,17,19–23].

This research proposes the creation and application of an optimization framework that makes use of cloud computing and uses a task-oriented scheduling strategy in order to overcome the aforementioned difficulties and enhance marine communication workflows. The task-oriented scheduling framework strategically ranks communication tasks, guaranteeing that urgent messages receive prompt attention and that non-urgent tasks are scheduled effectively to reduce resource waste. The proposed framework intends to enhance resource optimization, scalability, and communication efficiency through the use of cloud computing, resulting in safer, more effective, and resilient marine operations.

The usefulness of the proposed framework is verified through simulations, revealing its potential to change maritime communication workflows for the good of the marine industry. In the field of maritime operations, where efficient communication is essential for safety, coordination, and operational effectiveness, this research is of enormous value. The study deals with complex difficulties preventing efficient information exchange between ships, ports, and stakeholders. The research efficiently combines task-oriented scheduling and cloud computing. A paradigm shift might result from improved resource allocation, scalability, and prioritized communication. This modification might improve teamwork, expedite processes, and increase marine safety. Beyond changing marine communication, the findings could have significant effects on regulatory agencies, ports, and ships themselves. Through this research, navigation could achieve previously unheard-of efficacy and efficiency.

1.1. Main Contributions

Main contributions of this paper are briefed as follows:

The maritime communication workflow execution framework is designed for efficient management and execution of maritime workflows in cloud computing.

- A task-oriented smart controller component is defined in the proposed framework to define the nature of tasks before execution. This includes high-performance, latency-aware prioritized tasks, computational tasks, and data-oriented tasks.
- A resource-aware smart controller component is defined in the proposed framework to categorize the nature of resources for effective provisioning of tasks. This includes high-performance latency-aware resources, computation-aware resources, and data-aware resources.
- The maritime communication workflow scheduler component is defined for optimal scheduling of tasks for the resources. The scheduling strategy is represented by Maritime communication workflow optimization (MCWO) scheduling, which is further explained through pseudocode.
- The maritime communication workflow engine component is defined for the execution of tasks.

1.2. Organization of the Paper

The rest of the paper is organized as follows. Section 2 presents the literature. Section 3 provides the system's design and model. Section 4 provides the pseudocode-based solution for the proposed model. Section 5 presents the performance evaluation method, whereas Section 6 provides numerical results and discussions. Section 7 concludes the article.

2. Related Work

Cloud computing and maritime communication have attracted a lot of attention in recent years. Cloud environments have the ability to completely transform the maritime industry by improving productivity, safety, and resource management. The usage and potential of cloud computing in marine communication are examined [18]. It examines various cloud-based options for emergency response, cargo operations, and vessel traffic management. The survey [18] identifies the difficulties associated with data security and network connectivity in the marine domain and underlines the advantages of cloud computing, including scalability, cost-effectiveness, and data processing capabilities.

A well-known platform for corporations, IT companies, and mobile computing apps is cloud computing. Resources such as software, CPU, memory, and I/O devices are no longer purchased, they are instead used and taxed based on actual utilization. However, because of the rapidly rising cloud utilization, efficient resource allocation in this dynamic cloud environment becomes a difficult task. Although a number of solutions have been created to improve resource allocation effectiveness, inefficiencies in job scheduling and power usage still exist, particularly when the system is overloaded. Ref. [24] offers an energy-efficient task-scheduling algorithm and an approach to optimal power minimization in order to address this issue and boost the effectiveness of dynamic resource allocation. The suggested system achieves improved resource allocation in terms of job completion and reaction time by utilizing prediction mechanisms and a dynamic resource table update method.

Today, a key strategy for corporate operations across a variety of industries is cloud computing. Scientific enterprises are drawn to it because of its special qualities including on-demand capabilities, measurable service, virtualization, and quick adaptability. However, when the quantity of users and jobs grows, determining the best job scheduling strategy becomes difficult. Numerous cloud scheduling strategies in use today concentrate solely on one work type, either computation- or data-intensive. Unfortunately, such methods might not be appropriate in all contexts and could waste resources. To overcome these issues, [25] suggests the cost-based job scheduling (CJS) algorithm. During the job allocation process, the CJS algorithm takes into account data, processing power, and network factors. The simulation outcomes show that the CJS algorithm successfully decreases the response time of submitted jobs, which may include both data-intensive and computation-intensive tasks. This makes it a promising method for effective and balanced job scheduling in cloud environments.

Recent times have seen an increased requirement for quick and effective data processing due to the ever-expanding communication networks and the continual accumulation of large amounts of data. However, carrying out such broad computing tasks necessitates the use of significant processing and storage resources, which raises the price of hardware. The task-scheduling problem, which aims to optimize resource allocation for optimal utilization and shortened reaction time, presents an important difficulty in this setting. By utilizing cloud and fog computing, the research in [26] seeks to develop a smart home energy management system that reduces hardware costs. A virtual machine matching technique that considers latency is suggested in order to schedule tasks effectively. A simulated case study is used to test the success of the suggested approach, and the algorithm is applied while taking important considerations such as execution time, latency, available memory, and a cost function into account. The simulation's results reveal the suggested algorithm's superior performance and highlight its potential to address job scheduling.

Numerous industries have seen a significant increase in the use of cloud computing, which gives users the freedom to use computing resources as required. Effective task scheduling is one of the major obstacles in cloud computing, and it is essential to decreasing wait times and maximizing resource use. The study in [27] introduces the Enhanced Marine Predator Algorithm (EMPA), a scheduling efficiency technique that has been modified to meet this problem. The approach starts by building a task scheduling model that takes resource utilization and makespan into account. The goal of the algorithm is to find the best outcome. Each individual in the algorithm represents a potential work scheduling solution. This is accomplished by integrating the golden sine method, nonlinear inertia weight coefficient, and an operator from the Whale Optimization Algorithm (WOA) into the Marine Predator Algorithm. The experimental results emphasize the advantages of the EMPA algorithm in handling work scheduling issues in cloud computing environments by showing that it beats the other algorithms in terms of makespan, degree of imbalance, and resource consumption.

The idea of mobile edge computing (MEC) as an important technology in the following generation of networks is expanded upon in [17]. The exponential growth of vessel user data as a result of the rapid development of marine networks has made it necessary to perform resource-intensive tasks such as multimedia apps and high-definition video playback. As a result of the fusion of cloud computing with mobile Internet technology, the maritime mobile cloud network has arisen in response. The article suggests the implementation of computation offloading technology into maritime mobile cloud networks to address the issues with rising energy consumption and bandwidth demands in vessel terminals and networks. The study's main objective is to reduce energy consumption and execution delays through the offloading of computed tasks for vessel terminals. The suggested multivessel computational offloading algorithm seeks to decrease execution delay and device energy consumption. It is based on an enhanced Hungarian algorithm. The usefulness of the suggested plan is proved through simulations, proving its potential influence on boosting the functionality of marine mobile cloud networks.

A new paradigm of computing called fog computing brings cloud computing's capabilities out to the network's edge. It is intended to alleviate the shortcomings of cloud computing and is especially created for applications demanding minimal latency. However, the dynamic, varied, and distributed resources present problems for the fog computing system. In order to improve resource usage and user satisfaction, effective scheduling and resource allocation are essential. The resource-aware scheduler RACE (Resource Aware Cost-Efficient Scheduler) is suggested in [28] to distribute incoming application modules to fog devices in a way that maximizes fog layer resource utilization. The scheduler also tries to cut down on application execution times, bandwidth utilization, and financial costs associated with utilizing Cloud resources. The results indicate the usefulness of the suggested RACE strategy in maximizing resource allocation and enhancing overall system performance in the setting of fog computing.

3. System Design and Model

This research work proposes an approach for optimization of maritime communication workflow execution with a task-oriented scheduling framework in cloud computing. The framework consists of several components with particular functionalities. Figure 1 shows the maritime communications workflow execution framework in which maritime communication workflows are scheduled and managed through cloud computing resources.

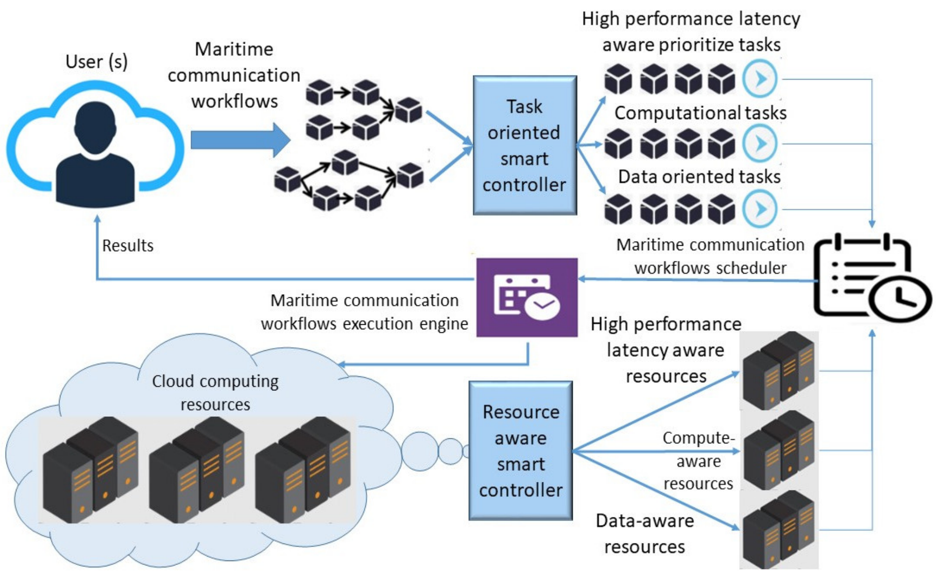

Figure 1. A framework for maritime communication workflows execution.

Communication workflows are submitted by one or more users to the task-oriented smart controller. Task-oriented smart controllers analyze the communication workflows and generate tasks with three categories, i.e., high-performance latency-aware prioritized tasks, computational tasks and data-oriented tasks. All three types of tasks are submitted to the maritime communication workflows scheduler. Cloud computing resources are analyzed by resource aware smart controller and categorized them into three categories, i.e., high-performance latency-aware resource, computation-aware resources and data-aware resources. All these three types of resources are also submitted to the maritime communication workflows scheduler. The maritime communication workflows scheduler schedules the tasks to cloud resources and sends them to the execution engine for execution. The execution engine executes the tasks and after successful execution the results are generated and returned to the user.

The paper proposes a novel strategy that combines task-oriented scheduling and resource-aware cloud environments in order to improve marine communication operations. Despite the vital role that maritime communication plays in upholding the safety, effectiveness, and efficiency of marine operations, traditional communication methods find it difficult to keep up with the continuously evolving demands of the industry. This research, however, suggests a possible solution to these problems by combining task-oriented scheduling and cloud computing. The use of cloud computing provides a scalable, adaptable infrastructure that can efficiently manage the diverse and distributed nature of marine communication operations. By employing cloud resources, stakeholders in the marine industry may manage enormous amounts of data, enhance decision-making, and ensure seamless communication even during hectic times. In addition to intelligently prioritizing communication activities, the proposed task-oriented scheduling framework also ensures that important messages are handled immediately while optimizing resource consumption.

The main components of the proposed maritime communication workflows execution framework are as follows.

3.1. Task-Oriented Smart Controller

Task-oriented smart controllers obtain maritime communication workflows and generate tasks with three categories. These categories are high-performance latency-aware prioritized tasks, computational tasks and data-oriented tasks. Equation (1) shows the generation of tasks from communication workflows.

$$\mathrm{MCW} = \sum_{i=1}^{n} \left(\mathrm{Priority}(\mathrm{T_i}) + \mathrm{Computational}(\mathrm{T_i}) + \mathrm{Data}(\mathrm{T_i})\right) \quad (1)$$

where MCW represents maritime communication workflows, Priority ($\mathrm{T_i}$) represents high-performance latency-aware prioritized tasks, Computational ($\mathrm{T_i}$) represents computational tasks, and Data ($\mathrm{T_i}$) represents data-oriented tasks.

Computational tasks are those tasks that require high computational power for processing and data storage is not very important in such cases. The data-oriented tasks are those tasks which require high storage power for execution and high computational power is not important in such cases. High-performance latency-aware prioritized tasks is the special category which covers both computational and data-oriented tasks; however, these tasks have priorities based on criticality, urgency and impact on maritime operations. Equation (2) shows the generation of high-performance latency-aware prioritized tasks.

$$\mathrm{Priority}(\mathrm{T_i}) = \sum_{i=1}^{n} \left(\mathrm{Criticality}(\mathrm{T_i}) + \mathrm{Urgency}(\mathrm{T_i}) + \mathrm{Impact}(\mathrm{T_i})\right) \quad (2)$$

where Criticality ($\mathrm{T_i}$) is the importance of the task, Urgency ($\mathrm{T_i}$) is the time-sensitivity of the task, and Impact ($\mathrm{T_i}$) is the potential effect on maritime operations.

The tasks are generated by the task-oriented smart controller component of the proposed framework. It receives maritime communication workflows as input from one or more users or locations. Maritime communication workflows are specialized combinations of tasks with several resource requirements in terms of memory, storage, and bandwidth. These workflows include vessel-to-shore communication, safety and navigational communication, emergency response, cargo operations, search and rescue, environmental monitoring, and port operations. The task-oriented smart controller, after receiving the maritime communication workflows, split them into three categories: priority tasks, computational tasks, and data-oriented tasks.

The decision regarding categorization is made by considering the nature of the tasks and making a few assumptions. If the task is critical, needs to be executed urgently, and is assumed to have an impact on the safety of vessels, the environment, or human life, then it will be categorized as a priority task. If the task involves complex calculation, data processing, or mathematical modeling, it will be categorized as a computational task.

Computational tasks include forecasting or collision avoidance algorithms. Similarly, if the task involves the collection, transmission, or storage of data, it will be categorized as a data-oriented task. Data-oriented tasks include sensing data from navigational systems or cargo tracking systems.

3.2. Resource Aware Smart Controller

Resource aware smart controllers obtain resources from cloud computing environment and make three categories. These categories are high-performance latency-aware resource, computation-aware resources, and data-aware resources. Equation (3) shows the categorization of resources after obtaining from cloud environment.

$$\mathrm{CCR} = \sum_{i=1}^{n} (\mathrm{High\ Performance}(\mathrm{R_i}) + \mathrm{Computational}(\mathrm{R_i}) + \mathrm{Data}(\mathrm{R_i})) \tag{3}$$

where CCR represents cloud computing resources, High Performance ($\mathrm{R_i}$) represents high-performance resources, Computational ($\mathrm{R_i}$) represents computation-aware resources, and Data ($\mathrm{R_i}$) represents data-aware resources.

3.3. Maritime Communication Workflows Scheduler

The maritime communication workflow scheduler obtains tasks from task-oriented smart controller, obtains resources from resource aware smart controller and schedules them. Communication scheduler works in three steps. In the first step, it schedules high-performance latency-aware prioritized tasks and assigns them to high-performance resources. In the second step, it maps data-oriented tasks to the data-aware resources and computational-oriented tasks to computation-aware resources. The third and final step is further divided into two phases. In the first phase of the final step, it finds and schedules the most fitted data-aware resource to each data-oriented task by finding minimum data transfer time as reflected in Equation (4).

$$\mathrm{DRA}(\mathrm{T_i}) = \mathrm{Minumum\_Transfer\_Time}(\mathrm{Data}(\mathrm{R_i})) \tag{4}$$

where DRA ($\mathrm{T_i}$) represents the data-aware resource allocation for each task and Minimum_Transfer_Time (Data ($\mathrm{R_i}$)) represents the minimum transfer time of a task to the specific resource.

In the second phase of the final step, it finds and schedules the most fitted computation-aware resource to each computational-oriented task by finding the minimum computation time as reflected in Equation (5).

$$\mathrm{CRA}(\mathrm{T_i}) = \mathrm{Minumum\_Compute\_Time}(\mathrm{Compute}(\mathrm{R_i})) \tag{5}$$

where CRA ($\mathrm{T_i}$) represents the computation-aware resource allocation for each task and Minimum_Compute_Time (Compute ($\mathrm{R_i}$)) represents the minimum computational time of a task to the specific resource.

So far, as the overlapping of tasks in categorization is concerned, it is assumed that high-performance resources can not only handle urgent tasks but can also handle both computational and data-oriented tasks. If a task is both urgent and computational, then it will fall into the urgent nature of tasks. Similarly, the computational resources also have enough storage power, although not as much as the data-oriented resources; thus, if a task is both computationally and data-intensive, it will be checked to see whether the computational resources can meet the storage requirement; otherwise, it will be assigned to the high-performance resource. That is why scheduling in cloud computing is more important than any other functionality.

In order to decide to what extent a task is considered computational or data-intensive, the decision is taken by considering the resource requirements of the task. When we create a task in CloudSim, we mention the resource requirement; thus, for the real-world implantation of the proposed approach in the maritime industry, it will be assumed that the resource requirements of the tasks are known in advance.

3.4. Problem Formulation

Marine communication workflow scenarios include GPS location services, route planning, weather forecasts, and navigation assistance. GPS location services contain GPS data retrieval, data preprocessing, location updates, and geo-fencing. Route planning contains path calculation, route optimization, and data integration from various sources, including AIS (Automatic Identification System), weather data, and navigational charts. Weather forecasts contain data collection, numerical modeling, and data analysis. Navigation assistance includes real-time monitoring, alert generation, and communication tasks. We express the marine communication workflow by Equation (6).

$$\mathrm{MCW} = \sum_{i=1}^{n} (\mathrm{GPSW_i} + \mathrm{RPW_i} + \mathrm{WFW_i} + \mathrm{NAVW_i}) \tag{6}$$

where GPSW represents the GPS services workflow, RPW represents the route planning workflow, WFW represents the weather forecast workflow, and NAVW represents navigation assistance workflow. A cloud-based framework is proposed to schedule and manage the tasks of maritime communication workflows.

By using Equation (6), we aim to maximize throughput, while minimizing latency, execution cost, and energy consumption. The proposed framework is used to organize and schedule tasks effectively in GPS location services by giving priority to important location updates, making data preprocessing pipelines faster and more accurate, and allocating resources based on how often and how important tasks are. In route planning, the proposed framework can be used for path calculation, route optimization, and data integration from various sources by taking AIS as a computation-intensive task since it includes a lot of processing, or as a data-oriented task due to its large storage or prioritizing routes based on vessel schedules and safety requirements. In weather forecasts, tasks can be scheduled and managed by parallelizing numerical modeling processes, optimizing data storage and retrieval, and ensuring that the most critical forecasts receive priority. In navigation assistance, an effective solution to schedule and manage tasks is to streamline real-time monitoring and ensure low-latency communication channels for navigation guidance.

4. Maritime Communication Workflows Engine

Maritime communication workflow engine obtains scheduled maritime communication tasks to the required resources and executes them. If any task failed to execute, it again obtains the information from workflow scheduler and re-execute it. After successful execution results are compiled, generated, and returned to the user. Resources are released after execution and added again to the available pool of cloud computing.

Algorithm 1 shows the optimized process of scheduling communication tasks to required resources and it represents Maritime Communication Workflows Optimization (MCWO) Scheduling.

The "MCWO Scheduling" method is designed to enhance marine communication workflows in a cloud computing environment through task-oriented scheduling. It begins by classifying the tasks according to their criticality, urgency, impact, computational intensity, and data intensity before applying the MCW formula to determine their total importance. Resources are classified according to their criticality, urgency, impact, computational awareness, and data awareness, with the CCR formula used to determine their suitability. After scheduling tasks to resources, the method determines the minimal computation time and transfer time for computational and data-oriented tasks, respectively, and prioritizes high-performance resources for priority tasks. The task execution engine manages it, produces results, and, if necessary, re-executes unsuccessful tasks. The method attempts to improve the overall efficiency and efficacy of maritime communication processes in the cloud computing environment by optimizing resource allocation, task prioritizing, and task execution.

Algorithm 1: MCWO Scheduling

Input: α: maritime communication workflows
Output: β: results
Procedure: MCWO (α)
Task Collection and Categorization

Tasks = (T_1, T_2, T_3, …, T_N)
$MCW = \sum_{i=1}^{n} (\text{Priority}(T_i) + \text{Computational}(T_i) + \text{Data}(T_i))$
for (each task T_1 to T_N) **do**
Categorize_Tasks( )
Q1 = Priority_Tasks(task.criticality, task.urgency, task.impact)

$$\text{Priority}(T_i) = \sum_{i=1}^{n} (\text{Criticality}(T_i) + \text{Urgency}(T_i) + \text{Impact}(T_i))$$

Q2 = Computational_Tasks (task.compute_intensive)
Q3 = Data_Oriented_Tasks (task.data_intensive)
end for

Resource Categorization

Resources = (R_1, R_2, R_3, …, R_N)

$$CCR = \sum_{i=1}^{n} (\text{High Performance}(R_i) + \text{Computational}(R_i) + \text{Data}(R_i))$$

for (each resource R_1 to R_N) **do**
Categorize_Resources ( )
Q4 = High_Performance_Resources (Resource.criticality, Resource.urgency, Resource.impact)
Q5 = Computational_Resources (resource.compute_aware)
Q6 = Data_Aware_Resources (resource.data_aware)
end for

Task Scheduling

Schedule_Task ( )
while (Priority_Tasks) **do**
    Priority_Task (R_i) $\leftarrow$ High_Performance_Resources (R_i)
end while
for (Computational_Task T_i) **do**
$CRA(T_i) = \text{Minumum\_Compute\_Time}(\text{Compute}(R_i))$
end for
for (each Data_Oriented_Task T_i) **do**
$DRA(T_i) = \text{Minumum\_Transfer\_Time}(\text{Data}(R_i))$
end for

Task Execution and Result Generation

Execution_Engine ( )
while (WorkflowEngine( )) **do**
    if (execution_failed (T_i))
    Re-execute_task (T_i)
    else
    Generate_results( )
        $\beta \leftarrow$ results
    end if else
end while
Return β

5. Performance Evaluation Method

The performance evaluation method involves the creation of simulated cloud environment to evaluate the proposed framework for optimizing maritime communication workflows. The method includes the following components.

5.1. CloudSim Simulation Toolkit

The cloud computing simulator, i.e., CloudSim [29–31] is used to evaluate the proposed MCWO scheduling. CloudSim provides extensive libraries for creating cloud data centers, virtual machines (VMs), and cloudlets (communication tasks). We created the cloud data center with multiple VMs and defined its characteristics, such as processing capacity, memory, storage, and bandwidth. We prepared a dataset of communication tasks representing various maritime communication scenarios. Each task includes attributes such as criticality, urgency, impact, and communication load. This dataset will serve as the input for the simulation. We implemented the mathematical model for task prioritization based on the attributes of each communication task. We implemented algorithm 1 to allocate cloud resources to each communication task based on three predefined categories. Table 1 shows the details of resource configurations.

Table 1. Resource configurations.

No. of VMs	**Memory (MB)**	**BW (Mbps)**	**VM**	**Arch**
1000 VMs	1024–8192	4000–10,000	Xen	X86
OS	**Cost per VM ($/Hr)**	**Memory Cost ($/s)**	**Storage Cost ($/s)**	**Data Transfer Cost ($/s)**
Linux	3.0	0.05	0.1	0.1

There is no standard for categorizing the tasks into criticality, urgency, or impact; however, it is assumed the system knows the transmission is carried out regarding the safety of vessels, the environment, human life, forecasting, collision avoidance, navigational systems, and cargo-tracking systems. If the task has a high impact on the safety of vessels, the environment, or human life, then it will be categorized as a priority task. If the task involves complex calculation, data processing, or mathematical modeling, it will be categorized as a computational task. Computational tasks include forecasting or collision avoidance algorithms. Similarly, if the task involves the collection, transmission, or storage of data, it will be categorized as a data-oriented task. Data-oriented tasks include sensing data from navigational systems or cargo tracking systems.

Table 2 shows the characteristics of VM configuration. For the creation of heterogeneous VMs, these configurations vary from VM to VM in terms of memory, bandwidth, cost, and processing power. The configurations reflect that:

- High-performance VMs are the resources for which every aspect is important and are required to deliver exceptionally fast and efficient performance. In the present case, each high-performance VM has 40,000 MB of image size, which indicates a long-running task that requires sustained high processing power. Similarly, four CPU cores show parallel processing capability and enable simultaneous execution of multiple tasks. The processing power of 4000 MIPS, 8 GB of RAM, and 10,000 Mbps of high bandwidth is compatible with high-performance cloudlets.
- The computation-intensive VMs are the resources optimized for tasks and workloads that require substantial computational power and processing capabilities. In the present case, each computation-intensive VM has a 5000 MB image size length, which indicates a relatively short task with a high computational power requirement as compared with data-intensive or traditional resources. Similarly, four CPU cores show parallelism and simultaneous execution of computation-intensive cloudlets. The processing power of 3000 MIPS, 4 GB of RAM, and 4000 Mbps of bandwidth is compatible with computation-intensive cloudlets.

- The data-intensive VMs are the resources optimized for tasks and workloads that involve large volumes of data. In the present case, each data-intensive VM has 40,000 MB of image size, which shows relatively large storage. Similarly, two CPU cores show moderate parallelism. The processing power of 1000 MIPS, 2 GB of RAM, and 6000 Mbps of bandwidth is compatible with data-intensive cloudlets.

Table 2. Resource configuration in terms of high-performance resources, computation-aware resources and data-aware resources.

VM Category	Image Size (MB)	CPU Cores	VM MIPS	VM RAM (MB)	VM BW (Mbps)
High performance	40,000	4	4000	8192	10,000
Computation-aware	5000	4	3000	4096	4000
Data-aware	40,000	2	1000	2048	6000

Table 3 shows the details of cloudlets configuration in terms of high-performance cloudlets, computationally intensive cloudlets, and data-intensive cloudlets.

Table 3. Cloudlets configuration in terms of high-performance cloudlets, computationally intensive cloudlets, and data intensive cloudlets.

Cloudlet Category	Length	CPU Core Requirements	Input File Size	Ouput File Size
High performance	40,000	4	10,000	10,000
Computationally intensive	5000	4	6000	6000
Data intensive	40,000	2	4000	4000

5.2. Performance Evaluation Parameters

In order to evaluate the effectiveness of the proposed framework, the following performance evaluation parameters are used.

Throughput: Throughput is measured by the rate at which tasks or jobs are processed or completed in a cloud computing environment [32,33]. It shows the number of tasks executed per unit of time. It is expressed as tasks per second (TPS) for maritime communication workflows and can be calculated by Equation (7).

$$\text{Throughput} = \frac{\text{Total number of completed tasks}}{\text{Total execution time}} \quad (7)$$

Latency: Latency is the time delay between the initiation of a task and the receipt of the corresponding response or completion notification [33–36]. It is an important metric for real-time applications and communication systems. Latency is measured in milliseconds (ms) and can be calculated by Equation (8).

$$\text{Latency} = \text{Finish}_{\text{time}} - \text{Task\_intiation}_{\text{time}} \quad (8)$$

Execution cost: Execution cost represents the monetary cost incurred to execute a particular task in the cloud environment [34,37,38]. Execution cost is measured in dollars for maritime communication workflows and can be calculated by Equation (9).

$$\text{Execution cost} = \text{Total\_exeuction}_{\text{time}} \times \text{Cost\_per\_unit}_{\text{time}} \quad (9)$$

Energy consumption: Energy consumption is the amount of power consumed by the cloud resources during execution of tasks over a specific period of time [34,37,38]. Energy consumption is measured in killowatt-hour (kwh) for maritime communication workflows and can be calculated by Equation (10).

$$\text{Energy Consumption} = \text{Power}_{\text{killowatt}} \times \text{Time}_{\text{hours}} \quad (10)$$

6. Results and Discussions

We designed a scenario for the evaluation of the proposed MCWO scheduling with 500, 1000, 1500, and 2000 heterogeneous tasks. The tasks are created by variation in data storage and computational requirement. Several tasks have also been assigned priorities in order to fully implement the proposed framework. All the tasks are executed on constant number of cloud computing resources, i.e., 1000 heterogeneous VMs. Resources are also created by variation in storage capacity and computational capacity. Several resources are created with high computational and storage power, and they have been termed high-performance resources. The results of the proposed MCWO are compared with the most relevant and latest scheduling strategy cost-based job scheduling (CJS) algorithm [25]. During the job allocation process, the CJS algorithm takes into account data, processing power, and network factors. The CJS algorithm take into account both data-intensive and computation-intensive tasks. This makes it a promising method for effective and balanced job scheduling in cloud environments. CJS is a cost optimization scheduling approach that assigns resources to tasks based on cost optimization. Thus, CJS was configured with all three categories of cloudlets by considering the cost as mentioned in Table 1 and without categorizing the resources.

Throughput: The throughput of the proposed MCWO scheduling as compared with the existing scheduling techniques is shown in Figure 2. As reflected in Figure 2, the proposed MCWO scheduling performed better in terms of throughput.

Latency: The latency of the proposed MCWO scheduling as compared with the existing scheduling techniques is shown in Figure 3. As reflected in Figure 3, the proposed MCWO scheduling performed better in terms of latency.

Execution cost: The execution cost of the proposed MCWO scheduling as compared with the existing scheduling techniques is shown in Figure 4. As reflected in Figure 4, the proposed MCWO scheduling performed better in terms of execution cost.

Energy Consumption: The energy consumption of the proposed MCWO scheduling as compared with the existing scheduling techniques is shown in Figure 5. Figure 5 shows that the proposed MCWO scheduling performed better in terms of energy consumption significantly reduce the amount of energy.

Table 4 shows the evaluation results of the proposed MCWO scheduling compared with the existing CJS scheduling in terms of through, latency, execution cost, and energy consumption.

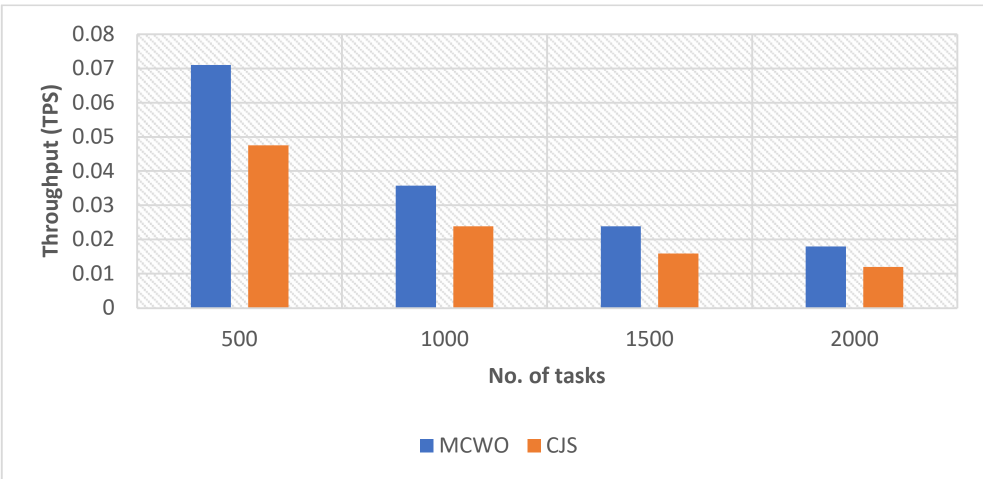

Figure 2. Throughput comparison of the proposed MCWO scheduling and existing CJS.

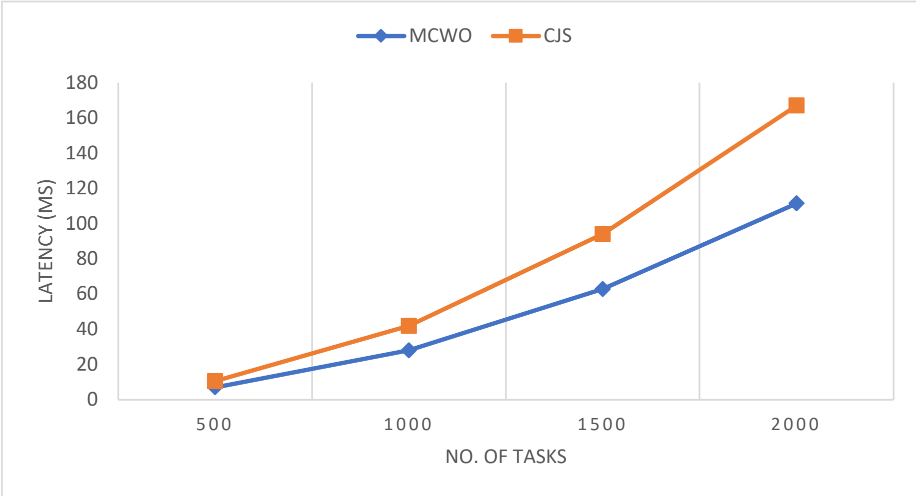

Figure 3. Latency comparison of the proposed MCWO scheduling and existing the CJS.

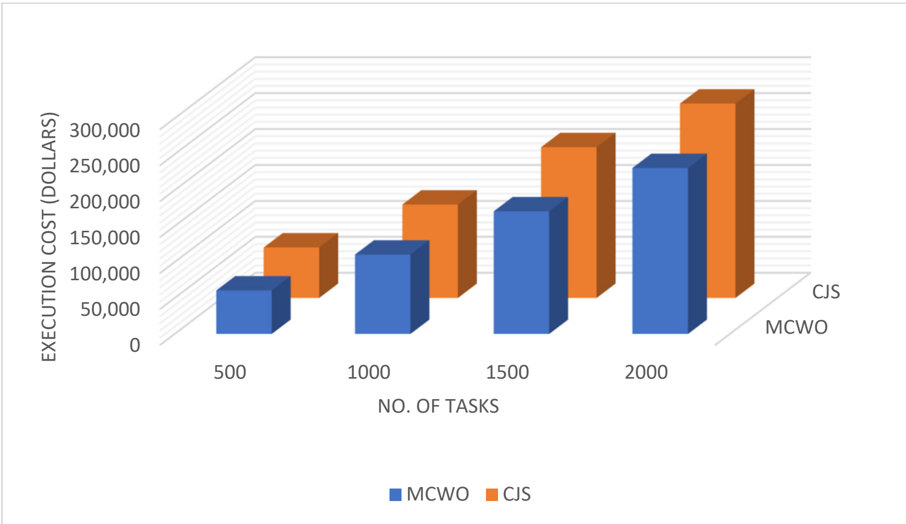

Figure 4. Execution cost comparison of the proposed MCWO scheduling and existing the CJS.

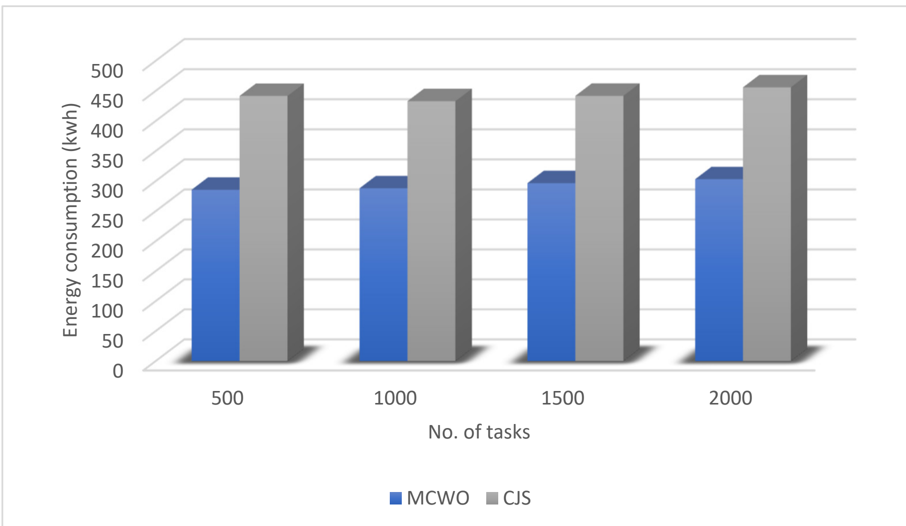

Figure 5. Energy consumption comparison of the proposed MCWO scheduling and the existing CJS.

Table 4. Evaluation results of the proposed MCWO scheduling and the existing CJS scheduling.

No. of Tasks	Throughput (TPS)		Latency (ms)		Execution Cost (Dollars)		Energy Consumption (kwh)	
	MCWO	**CJS**	**MCWO**	**CJS**	**MCWO**	**CJS**	**MCWO**	**CJS**
500	0.0709703	0.0475286	7.0452	10.51998	60,570	70,220	285.68133	441.71246
1000	0.0357478	0.023885	27.97374	41.86736	110,440	130,055	288.13755	432.95901
1500	0.023893	0.0159512	62.78002	94.03654	170,570	209,800	296.66246	441.8643
2000	0.0179417	0.0119737	111.4719	167.03235	230,990	270,440	303.20908	455.97739

The proposed MCWO scheduling mechanism works better than the existing method for maritime communication workflows. Through simulation with 1000 heterogeneous virtual machines running 500, 1000, 1500, and 2000 heterogeneous tasks each, MCWO showed improved throughput, decreased execution cost, and lower latency. The strategy's classification of tasks and resources, prioritization of tasks, and effective use of resources are all factors in its better performance. MCWO intelligently distributes workload, reduces wait times, and improves system performance by assigning tasks based on their characteristics and resource capabilities. A viable strategy for tackling the difficulties of work scheduling in cloud environments, enhancing system effectiveness, and lowering execution costs is the suggested MCWO scheduling. The suggested approach has significant benefits for marine communication procedures. Sensitive communications are given priority by MCWO, which assures effective use of computing and storage resources, reducing latency and enhancing overall communication performance. The flexibility of MCWO scheduling to change task scheduling and resource allocation through categorization in response to changing situations provides for uninterrupted communication workflows. The cost-effectiveness and long-distance communication optimization of MCWO makes it a viable option for improving marine communication, providing stakeholders and vessel operators with quick and effective communication services.

Task scheduling is the process of efficiently assigning resources to tasks in a way that optimizes the use of available resources. We have already defined that the dependent tasks will be executed sequentially; thus, there is no question that a high-performance task could only be executed after receiving the return from a data-intensive task. Mapping the communication workflows, including the dependent and independent tasks, is performed using CloudSim by creating heterogeneous cloudlets with variations in resource requirements. Virtual machines (VMs) were also created with variable characteristics. As reflected in Algorithm 1, the tasks are classified according to their characteristics, including priority, computational intensity, and data orientation. Cloud resources in terms of VMs were also categorized according to their capabilities. The sequence of execution is determined by the dependencies between the tasks, which ensure the dependent operations proceed in a sequential manner and independent tasks can be executed concurrently. Resource optimization reduces the wastage of resources while taking into account the cost associated with data transportation. The most effective solution for maritime communication workflows is determined when comparing the cost of the proposed MCWO framework with the CJS framework. The proposed scheduling framework outperformed due to factors including task categorization and resource optimization.

The experimental setup with 500, 1000, 1500, and 2000 heterogeneous tasks is a demonstrative environment applicable for the management and scheduling of all the cloud-based tasks of maritime communication workflows. When it comes to certain marine communication scenarios, such as GPS location services, route planning, weather forecasts, and navigation assistance, the proposed algorithm offers significant optimization. GPS location services contain GPS data retrieval, data preprocessing, location updates, and geo-fencing. The proposed algorithm is able to schedule and manage these tasks efficiently by prioritizing critical location updates, optimizing data preprocessing pipelines for speed and accuracy, and allocating resources based on the frequency and importance of tasks. Route planning contains path calculation, route optimization, and data integration from various sources, including AIS (Automatic Identification System), weather data, and navigational charts. The proposed algorithm schedules and manages these tasks by taking AIS as a computation-intensive task since it includes a lot of processing, whether data-oriented tasks due to its large storage, or prioritizing routes based on vessel schedules and safety requirements. Weather forecasts contain data collection, numerical modeling, and data analysis. The proposed algorithm has the ability to schedule and manage these tasks by parallelizing numerical modeling processes, optimizing data storage and retrieval, and ensuring that the most critical forecasts receive priority. Navigation assistance includes real-time monitoring, alert generation, and communication tasks. The proposed algorithm can be an effective solution to schedule and manage these tasks by streamlining real-time monitoring and ensuring low-latency communication channels for navigation guidance.

7. Conclusions and Future Work

An important development in the maritime sector is the optimization of maritime communication workflows using a task-oriented scheduling framework in cloud computing. Traditional communication methods struggle to keep up with the constantly changing demands of the sector, despite the important role that maritime communication plays in maintaining the safety, efficacy, and efficiency of marine operations. However, this research proposes a potential remedy to deal with these issues by combining task-oriented scheduling and cloud computing. A flexible and scalable infrastructure that can effectively manage the varied and distributed nature of marine communication operations is provided by the usage of cloud computing. Stakeholders in the marine industry may manage massive amounts of data, improve decision-making, and guarantee seamless communication even during busy times by utilizing cloud resources. The proposed task-oriented scheduling architecture also intelligently prioritizes communication tasks, ensuring that urgent messages are addressed right away while maximizing resource use. The research demonstrates the efficacy and practical application of the proposed framework through simulations. The outcomes show that it can increase task prioritizing, optimize communication workflows, and boost overall communication effectiveness. The results of this study show the effectiveness of the proposed framework for marine communication workflows. The combination of task-oriented scheduling and resource aware cloud computing environment results in cost savings, improved throughput, and minimum latency. The proposed approach has the potential to dramatically enhance maritime emergency response, cargo operations, navigation, and traffic management, as well as regulatory compliance.

In future, the proposed framework will be enhanced to make it an energy-efficient framework with dynamic load balancing mechanism. We also intend to make the proposed framework as a fog-enabled maritime communication workflows management system.

Author Contributions: Conceptualization, Z.A. and T.A.; Data curation, Z.A. and T.A.; Formal analysis, Z.A., T.A. and W.K.; Funding acquisition, W.K.; Investigation, T.A. and W.K.; Methodology, Z.A.; Project administration, T.A. and W.K.; Resources, T.A. and W.K.; Software, Z.A.; Supervision, T.A. and W.K.; Validation, Z.A.; Visualization, Z.A. and W.K.; Writing—original draft, Z.A.; Writing—review and editing, T.A. and W.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by standard development project funded by the Ministry of Science, Republic of Korea, ICT grant number RS-2023-00228316129820768700001.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No data were used to support this study. However, any queries about the research conducted in this paper are highly appreciated and may be asked from the corresponding author upon request.

Acknowledgments: The authors appreciate the support from standard development project funded by the Ministry of Science, Republic of Korea, ICT grant number RS-2023-00228316129820768700001.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jahanbakht, M.; Xiang, W.; Hanzo, L.; Rahimi Azghadi, M. Internet of Underwater Things and Big Marine Data Analytics—A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 904–956. [CrossRef]
2. Tran, K.; Keene, S.; Fretheim, E.; Tsikerdekis, M. Marine Network Protocols and Security Risks. *J. Cybersecur. Priv.* **2021**, *1*, 239–251. [CrossRef]
3. Acarer, T. The Turkish model for improving Imo survey results and reducing ship accidents. *Dokuz Eylül Üniversitesi Denizcilik Fakültesi Derg.* **2019**, *11*, 117–136. [CrossRef]
4. Acarer, T. Yeni Nesil Mobil İletişim Sistemlerinin Gemi Kara Arasındaki Deniz Haberleşmesinde Kullanım Olanağı. *Akıllı Ulaşım Sist. Uygulamaları Derg.* **2023**, *6*, 139–154. [CrossRef]
5. Acarer, T. Denizcilik Araştırmaları Dergisi: Amfora Journal of Maritime Research: Amphora VHF Kısa Mesafe Deniz Haberleşmesinin Data İletişimine Dönüşmesinin Deniz İşletmelerinin Gemi Yönetimleri İçin Temin Edeceği Olanaklar Possibilities to be Provided by Maritim. 2023, pp. 15–31. Available online: https://dergipark.org.tr/en/pub/amfora/issue/78609/1272782 (accessed on 1 October 2023).
6. Periola, A.A.; Alonge, A.A.; Ogudo, K.A. Architecture and System Design for Marine Cloud Computing Assets. *Comput. J.* **2020**, *63*, 927–941. [CrossRef]
7. Chilipirea, C.; Petre, A.; Groza, L. An Integrated Architecture for Future Studies in Data Processing for Smart Cities. *Microprocess. Microsyst.* **2017**, *52*, 335–342. [CrossRef]
8. Saeed, W.; Ahmad, Z.; Jehangiri, A.I.; Mohamed, N.; Umar, A.I. A Fault Tolerant Data Management Scheme for Healthcare Internet of Things in Fog Computing. *KSII Trans. Internet Inf. Syst.* **2021**, *15*, 35–57.
9. Rehman, A.U.; Ahmad, Z.; Jehangiri, A.I.; Ala'Anzy, M.A.; Othman, M.; Umar, A.I.; Ahmad, J. Dynamic energy efficient resource allocation strategy for load balancing in fog environment. *IEEE Access.* **2020**, *8*, 199829–199839. [CrossRef]
10. Nadeem, S.; Amin, N.u.; Zaman, S.K.u.; Khan, M.A.; Ahmad, Z.; Iqbal, J.; Khan, A.; Algarni, A.D.; Elmannai, H. Runtime Management of Service Level Agreements through Proactive Resource Provisioning for a Cloud Environment. *Electronics* **2023**, *12*, 296. [CrossRef]
11. Ahmad, Z.; Jehangiri, A.I.; Iftikhar, M.; Umer, A.I.; Afzal, I. Data-Oriented Scheduling with Dynamic-Clustering Fault-Tolerant Technique for Scientific Workflows in Clouds. *Program. Comput. Softw.* **2019**, *45*, 506–516. [CrossRef]
12. Wang, S.; Zhao, T.; Pang, S. Task Scheduling Algorithm Based on Improved Firework Algorithm in Fog Computing. *IEEE Access* **2020**, *8*, 32385–32394. [CrossRef]
13. Chen, B.; Li, J. Smart Health Management Framework for Maritime Communication System. *J. Coast. Res.* **2020**, *104*, 286–290. [CrossRef]
14. Hu, H.; Li, Z.; Hu, H.; Chen, J.; Ge, J.; Li, C.; Chang, V. Multi-objective scheduling for scientific workflow in multicloud environment. *J. Netw. Comput. Appl.* **2018**, *114*, 108–122. [CrossRef]
15. Zeng, X.; Garg, S.; Barika, M.; Zomaya, A.Y.; Wang, L.; Villari, M.; Chen, D.; Ranjan, R. SLA Management for Big Data Analytical Applications in Clouds. *ACM Comput. Surv.* **2021**, *53*, 1–40. [CrossRef]
16. Kaur, S.; Bagga, P.; Hans, R.; Kaur, H. Quality of Service (QoS) Aware Workflow Scheduling (WFS) in Cloud Computing: A Systematic Review. *Arab. J. Sci. Eng.* **2019**, *44*, 2867–2897. [CrossRef]
17. Yang, T.; Feng, H.; Yang, C.; Wang, Y.; Dong, J.; Xia, M. Multivessel Computation Offloading in Maritime Mobile Edge Computing Network. *IEEE Internet Things J.* **2019**, *6*, 4063–4073. [CrossRef]
18. Nomikos, N.; Gkonis, P.K.; Bithas, P.S.; Trakadas, P. A Survey on UAV-Aided Maritime Communications: Deployment Considerations, Applications, and Future Challenges. *IEEE Open J. Commun. Soc.* **2023**, *4*, 56–78. [CrossRef]
19. Callaghan, S.; Deelman, E.; Gunter, D.; Juve, G.; Maechling, P.; Brooks, C.; Vahi, K.; Milner, K.; Graves, R.; Field, E.; et al. Scaling up workflow-based applications. *J. Comput. Syst. Sci.* **2010**, *76*, 428–446. [CrossRef]
20. Chen, W.; Ferreira, R.; Deelman, E.; Sakellariou, R. Balanced Task Clustering in Scientific Workflows. In Proceedings of the 2013 IEEE 9th International Conference on e-Science, Beijing, China, 22–25 October 2013; IEEE: Piscataway, NJ, USA; pp. 1–8.
21. Farid, M.; Latip, R.; Hussin, M.; Hamid, N.A.W.A. A survey on QoS requirements based on particle swarm optimization scheduling techniques for workflow scheduling in cloud computing. *Symmetry* **2020**, *12*, 551. [CrossRef]
22. Zahoor, S.; Mir, R.N. Resource management in pervasive Internet of Things: A survey. *J. King Saud Univ.—Comput. Inf. Sci.* **2021**, *33*, 921–935. [CrossRef]
23. Zhou, J.; Wang, T.; Cong, P.; Lu, P.; Wei, T.; Chen, M. Cost and makespan-aware workflow scheduling in hybrid clouds. *J. Syst. Archit.* **2019**, *100*, 101631. [CrossRef]
24. Praveenchandar, J.; Tamilarasi, A. RETRACTED ARTICLE: Dynamic resource allocation with optimized task scheduling and improved power management in cloud computing. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 4147–4159. [CrossRef]
25. Mansouri, N.; Javidi, M.M. Cost-based job scheduling strategy in cloud computing environments. *Distrib. Parallel Databases* **2020**, *38*, 365–400. [CrossRef]

26. Memari, P.; Mohammadi, S.S.; Jolai, F.; Tavakkoli-Moghaddam, R. A latency-aware task scheduling algorithm for allocating virtual machines in a cost-effective and time-sensitive fog-cloud architecture. *J. Supercomput.* **2022**, *78*, 93–122. [CrossRef]
27. Gong, R.; Li, D.; Hong, L.; Xie, N. Task scheduling in cloud computing environment based on enhanced marine predator algorithm. *Cluster Comput.* **2023**. [CrossRef]
28. Arshed, J.U.; Ahmed, M. RACE: Resource Aware Cost-Efficient Scheduler for Cloud Fog Environment. *IEEE Access* **2021**, *9*, 65688–65701. [CrossRef]
29. Calheiros, R.N.; Ranjan, R.; Beloglazov, A.; De Rose, C.A.F.; Buyya, R. CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Softw. Pract. Exp.* **2011**, *41*, 23–50. [CrossRef]
30. Buyya, R.; Ranjan, R.; Calheiros, R.N. Modeling and simulation of scalable cloud computing environments and the cloudsim toolkit: Challenges and opportunities. In Proceedings of the 2009 International Conference on High Performance Computing and Simulation, HPCS 2009, Leipzig, Germany, 21–24 June 2009; Springer: New York, NY, USA, 2009; pp. 1–11.
31. Priya, B.; Gnanasekaran, T. Optimization of Cloud Data Center using CloudSim—A methodology. In Proceedings of the 2019 3rd International Conference on Computing and Communications Technologies (ICCCT), Chennai, India, 21–22 February 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 307–310. [CrossRef]
32. Iranmanesh, A.; Naji, H.R. DCHG-TS: A deadline-constrained and cost-effective hybrid genetic algorithm for scientific workflow scheduling in cloud computing. *Cluster Comput.* **2021**, *24*, 667–681. [CrossRef]
33. Rahbari, D.; Nickray, M. Low-latency and energy-efficient scheduling in fog-based IoT applications. *Turk. J. Electr. Eng. Comput. Sci.* **2019**, *27*, 1406–1427. [CrossRef]
34. Ahmad, Z.; Jehangiri, A.I.; Ala'anzy, M.A.; Othman, M.; Latip, R.; Zaman, S.K.U.; Umar, A.I. Scientific Workflows Management and Scheduling in Cloud Computing: Taxonomy, Prospects, and Challenges. *IEEE Access* **2021**, *9*, 53491–53508. [CrossRef]
35. Rajab, H.T.; Younis, M.F. Dynamic fault tolerance aware scheduling for healthcare system on fog computing. *Iraqi J. Sci.* **2021**, *62*, 308–318. [CrossRef]
36. Alatoun, K.; Matrouk, K.; Mohammed, M.A.; Nedoma, J.; Martinek, R.; Zmij, P. A novel low-latency and energy-efficient task scheduling framework for internet of medical things in an edge fog cloud system. *Sensors* **2022**, *22*, 5327. [CrossRef] [PubMed]
37. Hosseinzadeh, M.; Ghafour, M.Y.; Hama, H.K.; Vo, B.; Khoshnevis, A. Multi-Objective Task and Workflow Scheduling Approaches in Cloud Computing: A Comprehensive Review. *J. Grid Comput.* **2020**, *18*, 327–356. [CrossRef]
38. Ghasemzadeh, M.; Arabnejad, H.; Barbosa, J.G. Deadline-budget constrained scheduling algorithm for scientific workflows in a cloud environment. In Proceedings of the 20th International Conference on Principles of Distributed Systems (OPODIS 2016), Madrid, Spain, 13–16 December 2016; Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik: Wadern, Germany, 2017. [CrossRef]