Techno/Ecofeminism in Action: Fair and Responsible Resource Allocation for Sustainable Data Science Pipelines

Genoveva Vargas-Solar[1], José-Luis Zechinelli-Martini[2], Claudio A. Ardagna[4], Nicola Bena[4], Nadia Bennani[1], Barbara Catania[5], Javier A. Espinosa-Oviedo[3], Chirine Ghedira Guégan[1] and Andrea Mauri[1]

[1]CNRS, Univ Lyon, INSA Lyon, UCBL, LIRIS, UMR5205, F-69221, France
[2]Fundación Universidad de las Américas-Puebla, Exhacienda Sta. Catarina Mártir s/n 72820 San Andrés Cholula, Mexico
[4]Department of Computer Science, Università degli Studi di Milano, Italy
[5]Università degli Studi di Genova, Italy
[3]Univ. Lyon, Univ. Lyon 1, UR ERIC, EA 3083, Villeurbanne, France

Abstract

This paper introduces a dispatching approach to allocating computing resources for executing various activities within data science pipelines. The allocation strategy incorporates quantitative metrics—such as workload, performance in time, and memory consumption—and qualitative metrics emphasising fairness, responsibility, and sustainability. These qualitative considerations include the geographic location of servers, their CO_2 footprint, the frugality of data processing and analytics models, the conditions under which the data are produced, and the expected collective benefit of the processing outcomes. By integrating these qualitative metrics into resource-dispatching strategies and decision-making processes, the proposed algorithm aims to transform the execution of data science pipelines into a more ethical and equitable practice. This approach aligns with the principles of techno- and ecofeminism, advocating for technological solutions that prioritize collective social and environmental progress over purely capitalist gains. In this context, techno/ecofeminism provides a critical lens, emphasizing the importance of inclusivity, sustainability, and shared benefits in developing and deploying data-driven technologies. This work challenges extractive and inequitable models by grounding the dispatching strategy in these principles, proposing an alternative framework that leverages technology for holistic and equitable advancement.

Keywords

Data science pipelines, Fair dispatching, Fairness index, Negotiation, Techno/ecofeminism

1. Introduction

This paper introduces a dispatching approach to allocate computing resources for executing various activities within data science pipelines. The allocation strategy incorporates quantitative metrics —such as workload, performance in time, and memory consumption— and qualitative metrics emphasising fairness, responsibility, and sustainability. These qualitative considerations include the geographic location of computing resources, their CO_2 footprint, the frugality of data processing and analytics models, the conditions under which the data are produced, and the expected collective benefit of the processing outcomes. By integrating these qualitative metrics into resource-dispatching strategies and decision-making processes, the proposed approach aims to transform the execution of data science pipelines into a more fair and responsible practice. This approach aligns with the principles of techno and ecofeminism,[1] advocating for technological solutions prioritising collective social and environmental progress over purely capitalist gains. In this context, techno/ecofeminism provide a critical lens, emphasizing the importance of inclusivity, sustainability, and shared benefits in developing and deploying data-driven technologies. This work challenges extractive and inequitable models by grounding the dispatching strategy in these principles, proposing an alternative framework that leverages technology for holistic and equitable advancement.

Accordingly, the remainder of the paper is organised as follows. Section 2 gives a general overview of fair computing resource dispatching strategies for executing data science workloads. Section 3 introduces our dispatching approach. Section 4 reports an experimental validation to test our approach in different scenarios. Finally, Section 5 concludes the paper and discusses future work.

2. Related work

Resource dispatching involves allocating computing resources such as CPU, GPU, memory, and storage to specific tasks in an environment, often under constraints like time, budget, and quality of service (QoS). Traditional cluster schedulers such as Kubernetes and Apache Mesos allocate resources in containerized environments, emphasizing scalability and fault tolerance. Public cloud platforms like AWS, Azure, and Google Cloud offer elasticity and on-demand provisioning but require algorithms to manage dynamic pricing and preemption risks. Emerging paradigms prioritize resource dispatching across distributed and geographically dispersed devices, focusing on latency and energy efficiency.

The efficient dispatching of computing resources is a critical concern in modern computing environments, particularly for data science workloads characterized by diverse and resource-intensive operations. Data science tasks composing data science pipelines, ranging from data preprocessing

Published in the Proceedings of the Workshops of the EDBT/ICDT 2025 Joint Conference (March 25-28, 2025), Barcelona, Spain

*Genoveva Vargas-Solar.
†The authors' list is alphabetical except for the first and second authors.
genoveva.vargas-solar@cnrs.fr (G. Vargas-Solar); joseluis.zechinelli@udlap.mx (J. Zechinelli-Martini); claudio.ardagna@unimi.it (C. A. Ardagna); nicola.benaa@unimi.it (N. Bena); Nadia.Bennani@insa-lyon.fr (N. Bennani); barbara.catania@unige.it (B. Catania); javier.espinosa@univ-lyon1.fr (. J. A. Espinosa-Oviedo); chirine.ghedira-guegan@univ-lyon3.fr (C. G. Guégan); andrea.mauri@univ-lyon1.fr (A. Mauri)

[1]Technofeminism is a theoretical and practical framework that explores the intersections between technology, gender, and power [1]. Ecofeminism, also called ecological feminism, uses the basic feminist tenets of gender equality, revaluing non-patriarchal or nonlinear structures and a view of the world that respects organic processes, holistic connections, and the merits of intuition and collaboration [2].

to machine learning model training, require dynamic resource allocation to optimize execution time, ensure cost efficiency, and meet fairness criteria. This section explores key strategies, algorithms, and fairness considerations in resource dispatching for data science workloads.

2.1. Algorithms and Strategies for Resource Dispatching

Numerous algorithms and strategies have been proposed to optimize resource dispatching. These algorithms can broadly be categorized into heuristic, optimization-based, and learning-based approaches.

Heuristic-Based Algorithms provide computationally efficient, rule-based strategies for dispatching resources. Common heuristics include:

- First-Come-First-Served (FCFS): Tasks are executed in the order of arrival. While simple, FCFS often leads to resource starvation and inefficient utilization.
- Round-Robin (RR): Resources are evenly distributed among tasks in cyclic order, avoiding starvation but may not optimize resource usage.
- Min-Min and Max-Min: Min-Min first selects tasks with the lowest resource demands, while Max-Min prioritizes those with the highest. These methods aim to balance workloads but may overlook fairness.

Optimization-Based Algorithms model resource dispatching as mathematical problems seeking to minimize or maximize an objective function. Examples include:

- Linear Programming (LP): LP has been applied to model resource allocation problems in environments like high-performance computing clusters [3].
- Integer Programming (IP): Tasks with discrete resource requirements can be addressed using IP, which is computationally intensive but offers precision [4].
- Game Theory: Models like Nash Equilibrium provide frameworks for multi-agent systems where tasks compete for shared resources [5].

Learning-Based Algorithms. The rise of machine learning (ML) and deep reinforcement learning (DRL) has enabled adaptive and predictive resource dispatching:

- Reinforcement Learning (RL): RL models learn optimal policies by interacting with the environment. Algorithms like Q-Learning and Proximal Policy Optimization (PPO) have allocated resources in cloud computing scenarios [6].
- Neural Networks: Deep learning models predict resource demands based on historical workload patterns, improving dispatching decisions over time [7].
- Federated Learning (FL): FL trains models across decentralized devices while addressing privacy concerns, requiring careful resource dispatching to balance computation and communication costs [8].

2.2. Resource Dispatching for Data Science Workloads

Data science workloads are uniquely challenging due to their heterogeneity, high computational demands, and often unpredictable resource needs. Resource dispatching in this context has focused on:

- Workflow-Aware Scheduling: Platforms like Apache Airflow and DAG-based systems optimize resource allocation for multi-stage workflows.
- GPU Optimization: GPU-based workloads like deep learning training require specialized schedulers to minimize idle time and maximize utilization [9].
- Cost Efficiency: Cloud platforms use spot pricing and preemptible instances to reduce costs. Strategies must account for potential disruptions and ensure workload continuity [10].
- Energy Efficiency: Techniques like Dynamic Voltage and Frequency Scaling (DVFS) minimize energy consumption in large-scale data centres [11].

2.3. Fairness in Resource Dispatching

Fairness is an increasingly critical consideration in resource dispatching, particularly for data science workloads where multiple users and tasks compete for limited resources. Key approaches to achieving fairness include:

- Weighted Fair Queuing (WFQ): Assigns weights to tasks based on priority levels, ensuring proportional resource allocation [12].
- Dominant Resource Fairness (DRF): Proposed by Ghodsi et al. [13], DRF extends traditional fairness models to multi-resource environments, ensuring that no single resource type becomes a bottleneck.
- Max-Min Fairness: Ensures that the minimum allocation among tasks is maximized, balancing fairness with efficiency [14].
- Incentive Mechanisms: Game-theoretic approaches incentivize users to truthfully report resource demands, minimizing strategic manipulation [15].

2.4. Discussion

While significant progress has been made in resource dispatching, several critical challenges remain. Scalability is a primary concern, as algorithms must handle increasing workloads and heterogeneous environments without introducing significant overhead. Real-time adaptation poses another challenge, requiring dispatching decisions to dynamically adjust to changes in workloads, resource availability, and user demands. Ethical considerations further complicate resource dispatching, as ensuring fairness in multi-tenant systems while balancing efficiency and cost remains a complex issue. Lastly, sustainability has emerged as a pressing priority, with data centres consuming increasing energy, necessitating resource dispatching strategies prioritising green computing initiatives to reduce environmental impact. Addressing these challenges is essential to advancing the field and ensuring the effectiveness of resource-dispatching systems in diverse computing environments.

3. Fair Dispatching with Qualitative Metrics

Our dispatching algorithm is built on three key hypotheses.

- Hypothesis-1: The first hypothesis assumes that a data science pipeline to be executed includes fairness requirements for each task explicitly specified by humans (e.g., data scientists, domain experts, data owners, or communities potentially impacted by the analysis). Each task within the pipeline requires multiple resources to execute successfully, and these fairness requirements guide the resource allocation process.
  A data science pipeline is then guided by a *global* fairness objective and an approximation threshold, which defines the extent to which local and global fairness requirements can be met. The global fairness objective is an overarching guideline for ensuring ethical and equitable outcomes across the pipeline. At the same time, the approximation threshold specifies the allowable deviation or trade-offs in achieving fairness at both local (task-specific) and global (pipeline-wide) levels. This balance ensures that fairness is systematically incorporated while accounting for practical constraints.
- Hypothesis-2: The second hypothesis posits that data science tasks are executed using a pool of available resources, each characterized by distinct quantitative and qualitative properties. These properties can be validated and certified through a pre-established certification process, ensuring that the resources meet the necessary standards for the tasks they support [16, 17].
- Hypothesis-3: The third hypothesis assumes that the provision of computing resources follows just-in-time strategies, utilizing a dynamic pool of resources managed within virtualized environments. In this setup, given a set of tasks to be executed, a dispatcher dynamically allocates resources to tasks based on their alignment with both technical and qualitative requirements at both global and local levels. This approach ensures efficient and adaptive resource allocation while meeting the pipeline's broader fairness and performance objectives.

By integrating these three hypotheses, the algorithm ensures resource allocation meets technical demands and adheres to fairness principles, fostering ethical and responsible data science practices.

Our dispatching approach consists of 3 steps: 1) preparation of the execution environment consisting of available computing resources with the fairness properties that can potentially be acceptable to the data science pipeline requirements (Section 3.1); 2) fairness calculation for every computing resource in the pool (Section 3.2); 3) task dispatching seeking for resources that are eligible for a given task select the best resource according to local and global FI (Section 3.3).

3.1. Preparation of the Execution Environment

To prepare the execution environment for a given data science pipeline consisting of tasks with input data estimated computing resources requirement and a fairness objective, we elaborate on our work in Trust Negotiation [18]. The first is to build an execution environment with a pool of available resources tagged with qualitative metrics that totally or partially align to fairness technical and requirements using the following negotiation algorithm.

The trust negotiation algorithm [18] is designed to dynamically establish and manage trust (defined concerning fairness metrics) among resources in virtual environment pools. It evaluates trust values for each resource based on its profiles, including functional and non-functional attributes and their alignment with the trust requirements of other resources and tasks. The algorithm supports partial trust negotiation, allowing resources with suboptimal trust levels to participate under restricted conditions. Trust values are updated continuously as the execution of the data science pipeline tasks state evolves, ensuring adaptability and fairness. A centralized trust proxy coordinates the negotiation process, collecting profiles, enforcing trust policies, and resolving conflicts to maximize participation while maintaining data science tasks' reliability.

3.2. Fairness Calculation

This dispatching strategy leverages a **Fairness Index (FI)** [19] to allocate tasks to the most suitable computing resource in a distributed system. FI is computed on the basis of the following building blocks.

- **Computing Resources**: Each computing resource is defined by multiple attributes (e.g., location score, data provenance, GPU cores) and an initial pool of available resources.
- **Weights**: The importance of each factor in the FI computation is defined in a weights dictionary, allowing customization based on application needs.
- **Tasks** have resource requirements (denoted as `resource_needs`) that the selected computing resource must fulfil.

Function `calculate_fi` computes the Fairness Index (FI) for each computing resource by combining several metrics weighted according to their importance, as follows.

$$\begin{aligned} FainessIndex\ (FI) = & \alpha_1 L_s + \alpha_2 P_s + \\ & \alpha_3 DS_p + \alpha_4 MP + \\ & \alpha_5 T_t + \alpha_6 C_{GPU} + \\ & \alpha_7 C_{cal} + \beta_1 TC_{CO2} + \beta_2 EC \end{aligned} \tag{1}$$

where:

- **Location Score (L_s)**: Proximity or relevance of the computing resource's location to the task.
- **Data Provenance Score (P_s)**: Suitability of the computing resource's data origins for the task.
- **Sovereignty Score (DS_p)**: Compliance with data sovereignty requirements.
- **Model Performance (MP)**: Performance of the models deployed on the computing resource.
- **Training Time (T_t)**: Estimated time required to train models on the computing resource.
- **GPU Cores (C_{GPU})**: Availability of GPU resources.

- **Calibration Cycles (C_{cal})**: Computing resource's capacity to handle calibration demands.
- **Carbon Footprint (T_{CO_2})**: Environmental impact of utilizing the computing resource.
- **Economic Cost (EC)**: Financial cost associated with the computing resource's operation.

The coefficients α_1, …, α_7, β_1, β_2 are the weights assigned to each metric. They reflect the metrics relative importance within the fairness index. These weights can be adjusted based on the data science task's specific priorities or fairness objectives.

3.3. Task Dispatching

The FI guides in equation (1) dispatching by selecting resources best suited to meet a given task's technical and qualitative requirements. The negotiation algorithm allows one to choose the right resources that can participate in the execution of a task. In particular, dispatching function `dispatch_task` selects the most suitable server for a given a task based on its FI and available resources using the following three-steps process.

1. **Iterate Over Computing Resources**: For each computing resource, the FI is calculated using `calculate_fi`.
2. **Eligibility Check**: A computing resource is considered eligible if:
   - It has an FI close to the task requirements among available computing resources.
   - It has sufficient resources to meet the task's requirements: `available_resources` $\geq$ `resource_needs`.
3. **Select the Best Computing Resource**: The computing resource with the maximum FI that satisfies the resource constraints and can contribute to achieving the expected global FI is chosen.

Task Allocation Once a suitable computing resource is selected:

- **Resource Deduction**: The task's resource requirements are deducted from the available computing resources.
- **Task Assignment Notification**: The task is marked as assigned to the computing resource, and a confirmation message is sent.

If no suitable computing resource is found, an error is raised.

Example Workflow Let us assume that two servers ("A" and "B") have one available computing resource each to execute a data science pipeline. First, for each resource we compute its FI, and then match and allocate the tasks in the pipeline. Considering (0.1, 0.15, 0.2, 0.25, 0.1, 0.05, 0.05, 0.05, 0.05) as weights for metric values (0.8, 0.9, 0.7, 0.95, 0.8, 8, 3, 0.2, 0.5), listed in the order used in equation (1), the FI is computed as follows:

1. **FI Calculation**:
   - Server A: $FI = 0.1 \cdot 0.8 + 0.15 \cdot 0.9 + 0.2 \cdot 0.7 + 0.25 \cdot 0.95 + 0.1 \cdot 0.8 + 0.05 \cdot 8 + 0.05 \cdot 3 + 0.05 \cdot 0.2 + 0.05 \cdot 0.5$.
   - Server B: Similar calculation using its respective attributes.
2. **Best Server Selection**: Compare FI scores and resource availability. Assign the task to the server with the highest eligible FI.
3. **Task Allocation**: Deduct the task's resource needs from the selected server's resources and confirm the assignment.

3.4. Fairness-Aware AI Resource Dispatching: A Case Study in Global Health Research

This document presents a fairness-aware resource dispatching approach for AI-based model training in global health research. The study focuses on training a tuberculosis (TB) diagnostic model using X-ray images from hospitals across the Global South. The goal is to allocate computational resources while ensuring fairness, sovereignty, and environmental sustainability.

3.4.1. Available Computing Resources

Table 1 presents the available computing resources, including qualitative attributes (e.g., sovereignty, energy type) and quantitative metrics (e.g., training speed, CO_2 emissions).

3.4.2. Pipeline Fairness Constraints

The AI training pipeline consists of two tasks: Data Exploration and Preprocessing, and Model Training. Each task has FI requirements ensuring respect for sovereignty, sustainability, and computational efficiency (see Table 2).

3.4.3. Fair Dispatching and Negotiation Rounds

Round 1: Initial Task Allocation **Data Exploration:**

- Eligible servers: S1 (South Africa, FI = 0.9), S4 (Kenya, FI = 0.85).
- Best match: S1 (Solar-powered, high sovereignty, low CO_2 emissions).
- **Initial allocation:** S1 **OK**.

Model Training:

- Eligible servers: S1, S2, S5.
- Best match: S2 (Brazil, fastest GPU, moderate sovereignty, moderate CO_2 emissions).
- **Initial allocation:** S2 **OK**.

Round 2: Adjustments for Fairness
Data Exploration:

- S1 requests workload redistribution due to underutilization.
- S4 is added as a backup node to balance workload and redundancy.
- **Final allocation:** 70% of workload on **S1**, 30% on **S4**.

Model Training:

- S2 alone does not meet fairness goals.
- S5 (Argentina) is added to improve fairness in regional distribution.
- **Final allocation:** S2 (60%) + S5 (40%).

Table 1
Server profiles with qualitative and quantitative attributes

Server ID	Region	CPU (Cores)	GPU (TFLOPS)	RAM (GB)	Energy Type	Sovereignty	CO_2 (kg/hr)	Training Speed
S1	South Africa	64	120	512	Solar	0.9	0.1	500
S2	Brazil	128	200	1024	Hydro	0.7	0.2	700
S3	India	48	90	256	Coal	0.5	1.2	400
S4	Kenya	32	75	128	Wind	0.85	0.05	300
S5	Argentina	96	150	768	Nuclear	0.6	0.4	600

Table 2
Pipeline fairness constraints

Task	Sovereignty Min	CO_2 Max (kg/hr)	Training Speed Min (images/sec)
Data Exploration	≥ 0.8	≤ 0.3	N/A
Model Training	≥ 0.6	≤ 0.8	≥ 500

3.4.4. Final Task Allocation

The final task allocation is shown in Table 3. The table outlines the optimized distribution of tasks across servers following a negotiation process. It highlights how tasks such as Data Exploration and Model Training are allocated to specific servers, with percentages indicating the workload distribution. For instance, Data Exploration is split between Server 1 (70%) and Server 4 (30%), ensuring a balanced workload and incorporating redundancy for reliability. Similarly, Model Training is divided between Server 2 (60%) and Server 5 (40%), with adjustments aimed at improving fairness in the training process. The table reflects a careful consideration of workload balancing, fairness, and system reliability, suggesting that the allocation was designed to optimize resource utilization and prevent overloading any single server.

Table 3
Final Allocation After Negotiation

Task	Final Server Allocation	Fairness Adjustments
Data Exploration	S1 (70%), S4 (30%)	Balanced workload, redundancy
Model Training	S2 (60%), S5 (40%)	Improved fairness in training

3.4.5. Impact of Fair Dispatching

- **Improved Regional Fairness:** Avoids bias by distributing tasks across multiple Global South regions.
- **Energy-Aware Allocation:** Prioritizes solar and wind-powered servers for lower carbon footprint.
- **Preserved Data Sovereignty:** Ensures data governance laws are respected in high-sovereignty regions.
- **Optimized Compute Efficiency:** Training tasks leverage high-GPU servers while balancing fairness.

This case study demonstrates how the Fairness Index (FI)-based dispatching enables equitable AI model training for global health research while optimizing environmental impact, sovereignty, and computational fairness. The negotiation mechanism ensures balanced allocations, preventing regional bias and enhancing responsible AI development.

4. Fair and Responsible Dispatching in Practice

The experimental setting evaluates the proposed fair dispatching approach by simulating a representative scenario with diverse resource pools, fairness metrics, and data science pipelines. The goal is to demonstrate how the dispatcher allocates resources to meet the global Fairness Index (FI) expectations associated with data science pipelines.

Resources Pools. We created three distinct patterns of resource pools, each with varying capacity and fairness metrics:

- *High-Capacity, Low-Fairness Pool*: Large servers with extensive computational resources (64 GPU cores, 512 GB RAM).
  Fairness Metrics:
  ```
  Location Score: Low (0.3)
  Sovereignty Score: Low (0.4)
  Carbon Footprint: High (0.8)
  Economic Cost: Medium (0.6)
  ```
- *Balanced-Capacity, Medium-Fairness Pool*: Mid-sized servers with moderate computational resources (32 GPU cores, 256 GB RAM).
  Fairness Metrics:
  ```
  Location   Score:     Medium   (0.6)
  Sovereignty  Score:    Medium  (0.7)
  Carbon Footprint: Medium (0.5)
  Economic Cost: Low (0.4)
  ```
- *Low-Capacity, High-Fairness Pool*: Small servers with minimal computational resources (8 GPU cores, 64 GB RAM).
  Fairness Metrics:
  ```
  Location   Score:      High   (0.9)
  Sovereignty  Score:    High  (0.85)
  Carbon Footprint: Low (0.2)
  Economic Cost: Low (0.3)
  ```

Data Science Pipelines. We defined three pipelines with varying tasks, resource requirements, and global FI expectations:

- Pipeline A - High Computational Demand. Prioritizes computational efficiency, cost over fairness, and higher weights for GPU cores and training time.
  - Weights:
    ```
    Location Score: 0.1
    Data Provenance Score: 0.1
    Sovereignty Score: 0.1
    Model Performance: 0.3
    Training Time: 0.2
    GPU Cores: 0.2
    ```
  - Tasks:
    Data preprocessing: Requires medium resources (16 GPU cores, 128 GB RAM).
    Model training: Requires high resources (48 GPU cores, 256 GB RAM).
  - Global FI Expectation: Medium FI ≥ 0.65).

- Pipeline B - Low Computational Demand, High Fairness. Prioritize fairness metrics like location, sovereignty, and carbon footprint over computational efficiency.
  - Weights:
    ```
    Location Score: 0.3
    Data Provenance Score: 0.2
    Sovereignty Score: 0.3
    Model Performance: 0.1
    Training Time: 0.05
    GPU Cores: 0.05
    ```
  - Tasks:
    Data cleaning: Requires low resources (8 GPU cores, 64 GB RAM).
    Model tuning: Requires medium resources (16 GPU cores, 128 GB RAM).
  - Global FI Expectation: High FI ≥ 0.8.
- Pipeline C - Balanced Computational and Fairness Requirements.
  - Weights
    ```
    Location Score: 0.2
    Data Provenance Score: 0.2
    Sovereignty Score: 0.2
    Model Performance: 0.2
    Training Time: 0.1
    GPU Cores: 0.1
    ```
  - Tasks:
    Feature engineering: Requires medium resources (16 GPU cores, 128 GB RAM).
    Model training: Requires medium resources (32 GPU cores, 256 GB RAM).
  - Global FI Expectation: Medium-High FI $\geq$ 0.75.

Experimental Scenario for Evaluating Fair Dispatching We considered three scenarios to evaluate the fair dispatching approach under diverse conditions. We describe the setup, objective, and expected outcome for each scenario. The *Baseline Scenario* provides a reference for understanding default behavior. The *Increased Capacity Scenario* highlights the impact of resource abundance on fairness outcomes. The *Dynamic Fairness Adjustment Scenario* tests the adaptability of the algorithm to evolving fairness goals and resource constraints.

Baseline Scenario: It aims to evaluate the default behavior of the fair dispatching algorithm without any adjustments to resource capacities, fairness weights, or pipeline requirements.
Setup: A predefined set of resource pools with varying capacities and fairness properties (e.g., location, carbon footprint, economic cost). Data science pipelines with diverse tasks (data preparation, model training) and fixed fairness expectations.
Key Focus: Understand how well the algorithm meets fairness requirements using the existing resources and configuration without negotiation or dynamic adjustments.
Expected Outcome: A clear baseline to identify pipelines that succeed or fail to meet fairness expectations and highlight areas for improvement.

Increased Capacity Scenario: It aims to evaluate how increasing resource availability affects the ability of the dispatcher to meet fairness requirements.
Setup: Resource pools have their capacities increased by a fixed factor (e.g., 1.5x or 2x) while retaining the same fairness properties. Data science pipelines remain unchanged in terms of tasks and fairness expectations.
Key Focus: Examine whether additional resource availability reduces negotiation calls, improves global FI scores, or leads to better task allocation outcomes.
Expected Outcome: Insights into the impact of resource abundance on fairness and efficiency in dispatching, demonstrating the scalability of the approach.

Dynamic Fairness Adjustment Scenario: It aims to evaluate the ability of the dispatching algorithm to adapt to scenarios where fairness weights or expectations are dynamically adjusted.
Setup: Fairness weights for qualitative metrics (e.g., sovereignty, carbon footprint) are modified to prioritize specific fairness dimensions over others. Data science pipelines have dynamic FI expectations based on task priority or external conditions. The dispatcher employs negotiation strategies to adapt to these changes.
Key Focus: Evaluate the flexibility of the algorithm to handle shifting priorities and fairness goals while maintaining efficient resource allocation.
Expected Outcome: Demonstration of the adaptability of the approach, with insights into how fairness trade-offs affect allocation outcomes.

Dispatching Process We implemented a dispatching process as follows.

1. Input: Resource pools, pipelines, and associated weights.
2. Task Allocation: Calculate the FI for each resource for every task based on pipeline-specific weights. Select the resource with the highest FI that satisfies the task's resource requirements.
3. Global FI Validation: After all tasks are dispatched, compute the overall pipeline FI as the weighted average of the allocated resources' FIs. Ensure the global FI meets the pipeline's expectations.

Figure 1 shows our results. The plot represents the Global FI (Fairness Index) scores of three pipelines (Pipeline_A, Pipeline_B, and Pipeline_C) under different scenarios. The height of the bars represents the global FI scores achieved for each pipeline, and annotations indicate the number of negotiation calls required during the resource allocation process.

The dashed line indicates the threshold FI value, a benchmark for evaluating whether the pipelines meet fairness expectations. Its example value, 0.75, means that a pipeline must achieve a global FI score of at least 0.75 for the resources allocated to its tasks to be considered fair according to the predefined criteria. Pipelines above the threshold have a global FI score greater than or equal to 0.75. This indicates that the resource allocation meets or exceeds fairness requirements. In the plot, these bars are coloured green. Pipelines below the threshold have a global FI score of less than 0.75. This indicates that the allocation of resources did not meet the fairness criteria.

The threshold serves as a critical benchmark for evaluating pipeline performance against fairness expectations. It enables comparative analysis by distinguishing pipelines that meet fairness criteria from those that fall short. For pipelines below the threshold, it provides guidance for improvement, highlighting the need for adjustments such as

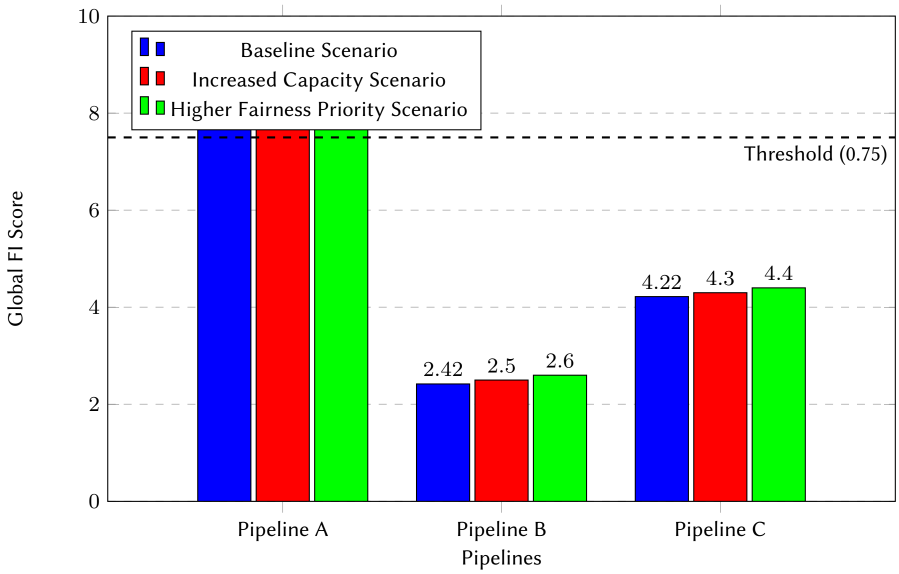

Figure 1: Experimental results of dispatching with negotiation across different pipelines and scenarios

changing fairness metric weights, increasing resource availability, or using negotiation processes to adjust task requirements. In addition, the threshold is highly adjustable, allowing it to be tailored to the specific fairness priorities of the system. For example, a higher threshold imposes stricter fairness requirements, ensuring more equitable resource allocation, while a lower threshold relaxes these requirements, increasing the likelihood that pipelines will meet expectations. This flexibility makes the threshold a versatile tool for evaluating and improving the fairness of resource allocation strategies.

Interpretation of Initial Experiments. In the Baseline Scenario (see Figure 1) Pipeline_A, achieves the highest FI score ($\sim 7, 8$), well above the threshold, with no need for negotiation. Pipeline_B and Pipeline_C have lower FI scores ($\sim 2, 42$ and $\sim 4, 22$, respectively) and require two negotiation calls each. Thus, they did not meet the fairness expectations. There was no significant difference in FI scores for the increased capacity scenario compared to the baseline scenario, indicating that increasing resource capacity did not directly influence the allocation or fairness outcomes. Negotiation calls remain the same, implying that the adjustments in this scenario did not alleviate the need for negotiations in resource allocation. For the higher fairness priority case, FI scores for Pipeline_B and Pipeline_C improve slightly compared to the baseline (e.g., Pipeline_B's score increases from $2, 42$ to $2, 60$). The number of negotiation calls remains constant, but the adjustments in fairness weights reflect a positive impact on pipelines with lower FI scores. Pipeline_A remains unaffected due to its already high FI score. For the reduced expectations scenario, FI scores and negotiation calls remain unchanged compared to the baseline. This scenario indicates that lowering fairness expectations (e.g., reducing FI thresholds) does not impact the allocation process but would allow more pipelines to "pass" the evaluation if the threshold is considered.

Pipeline_A consistently performs well, regardless of the scenario, suggesting it aligns better with the resource pool or has fewer resource constraints. Pipeline_B and Pipeline_C struggle to meet fairness expectations across all scenarios, with relatively low FI scores and the need for negotiation to adjust resource allocations. Higher fairness priority improves fairness for pipelines with lower FI scores, making it the most promising scenario for addressing disparities. The increased capacity and reduced expectations scenarios do not significantly change the allocation outcomes, highlighting that resource availability or relaxed thresholds alone are insufficient to improve fairness outcomes.

Advantages

- **Customizable Fairness**: Weights allow prioritization of sustainability, cost, or performance.
- **Dynamic Allocation**: The strategy adapts to server attributes or task requirements changes.
- **Fair Resource Utilization**: Ensures resource allocation considers technical and qualitative factors.

Limitations

- **Complex Weight Tuning**: Achieving an optimal balance among factors requires careful weight configuration.
- **Scalability**: Performance may degrade with many servers and tasks due to computational overhead in FI calculations.

5. Conclusions and Future Work

This paper presents a pioneering approach to fair and responsible resource dispatching for data science pipelines by incorporating technical and qualitative metrics into the allocation process. Integrating fairness metrics provides a foundation for equitable computational resource management, aligning with technofeminism and ecofeminism principles. The proposed dispatching mechanism shows potential for balancing computational efficiency with social and environmental fairness.

Fair resource dispatching, guided by qualitative fairness metrics, aligns deeply with the principles of technofeminism and ecofeminism by challenging systemic inequities and promoting inclusivity in allocating computational and environmental resources. Technofeminism, which seeks to dismantle the gendered biases embedded in technological systems, benefits from fairness-driven resource allocation by ensuring marginalized voices —often excluded from decision-making processes —have equitable access to technological infrastructure. Qualitative fairness metrics provide a framework for identifying and correcting historical imbalances, such as privileging Global North projects over Global South initiatives or reinforcing patriarchal priorities.

Ecofeminism, focusing on the interconnectedness of environmental justice and gender equality, similarly intersects with fair dispatching practices. Prioritizing energy-efficient, sustainable resource management and fair dispatching supports ecofeminism's aim to mitigate environmental harm caused by unchecked technological expansion. Together, these frameworks foster a redistribution of resources that values diverse perspectives, reduces systemic harm, and integrates sustainability with social justice in technology.

Open issues and Future work. While the results demonstrate the feasibility and relevance of the approach, it represents an initial step toward a broader vision of fair resource dispatching. Several directions for future work emerge from this study:

- Scalability and Realism: Experimentation with large-scale and more realistic resource pools and pipelines, including heterogeneous and dynamic resource environments. Deployment in real-world settings such as federated learning systems or global data science collaborations.
- Dynamic and Adaptive Weights: Development of algorithms that dynamically adjust fairness weights based on pipelines or systems' evolving priorities and constraints.
- Stability feature of resource allocation: Consider this feature as constant reallocation in response to minor changes can lead to inefficiencies; thus, incorporating stability mechanisms in future solutions would help avoid unnecessary reallocations, contributing to overall cost reduction efforts.
- Inclusion of feedback loops to learn from past allocations and refine the weight configuration over time.
- Negotiation Strategies: Advanced negotiation algorithms for handling resource shortages or conflicts while maintaining fairness. Integration of predictive analytics to proactively anticipate negotiation needs and optimize the resource allocation process.
- Cross-Domain Applications: Extension of the framework to interdisciplinary domains such as climate modelling, medical research, and global development projects, where fairness and resource optimization are critical.
- Enhanced Qualitative Metrics: Expansion of the fairness index to include new dimensions such as cultural representation, gender inclusivity, and accessibility. Use of machine learning models to quantify qualitative metrics more accurately.
- Transparency: Development of visualization tools for stakeholders to understand and monitor the allocation process and its fairness outcomes.
- Governance: Design of governance mechanisms to ensure accountability in fairness-driven dispatching decisions.

Addressing these challenges can evolve the proposed framework into a robust, scalable, and adaptable system for fair resource dispatching. Future experiments should involve diverse datasets and scenarios to validate the approach under varying conditions and demonstrate its utility for advancing ethical and responsible computational practices.

6. Acknowledgements

The work reported in this paper is performed in the context of the project FRIENDLY [2] funded by the inter-group program of the laboratory LIRIS, Lyon.

References

[1] J. Wajcman, Technocapitalism meets technofeminism: women and technology in a wireless world, Labour & Industry: a journal of the social and economic relations of work 16 (2006) 7–20.
[2] F. d'Eaubonne, Feminism or Death: How the Women's Movement Can Save the Planet, Verso Books, 2022.
[3] D. Bertsimas, J. N. Tsitsiklis, Introduction to Linear Optimization, Athena Scientific, 1997.
[4] J. Kleinberg, Éva Tardos, Algorithm Design, Pearson, 2006.
[5] T. Roughgarden, Selfish Routing and the Price of Anarchy, MIT Press, Cambridge, MA, 2005.
[6] H. Mao, M. Alizadeh, I. Menache, S. Kandula, Resource management with deep reinforcement learning, in: Proceedings of the 15th ACM Workshop on Hot Topics in Networks (HotNets), 2016.
[7] Z. Chen, B. Li, H. Jin, Y. Pan, Efficient resource allocation for cloud-based deep learning platforms, IEEE Transactions on Cloud Computing (2018).
[8] P. K. et al., Advances and open problems in federated learning, Foundations and Trends in Machine Learning (2021).
[9] R. Kumar, K. Kumar, A. Kaushik, A survey on resource allocation techniques in cloud computing for gpu-based workloads, Journal of Cloud Computing 10 (2021).
[10] Q. Zhang, L. Liu, C. Pu, Spot pricing for cloud workloads: Trends, challenges, and opportunities, IEEE Transactions on Cloud Computing (2019).
[11] A. Beloglazov, J. Abawajy, R. Buyya, Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing, Future Generation Computer Systems 28 (2012) 755–768.
[12] G. You, Y. Zhao, A weighted-fair-queuing (wfq)-based dynamic request scheduling approach in a multi-core system, Future Generation Computer Systems 28 (2012) 1110–1120.
[13] A. Ghodsi, M. Zaharia, B. Hindman, A. Konwinski, S. Shenker, I. Stoica, Dominant resource fairness: Fair allocation of multiple resource types, Proceedings of

[2] http://vargas-solar.com/friendly/

the 8th USENIX Symposium on Networked Systems Design and Implementation (NSDI) (2011).

[14] B. Radunovic, J.-Y. Le Boudec, A unified framework for max-min and min-max fairness with applications, IEEE/ACM Transactions on networking 15 (2007) 1073–1083.

[15] A. Agarwal, S. Kannan, A. Roth, Incentivizing fair resource allocation in distributed systems, Proceedings of the 21st ACM Conference on Economics and Computation (EC) (2020).

[16] C. A. Ardagna, N. Bena, Non-functional certification of modern distributed systems: A research manifesto, in: Proc. of IEEE SSE 2023, Chicago, IL, USA, 2023.

[17] C. A. Ardagna, N. Bena, N. Bennani, C. Ghedira-Guegan, N. Grecchi, G. Vargas-Solar, Revisiting trust management in the data economy: A road map, IEEE Internet Computing 28 (2024).

[18] N. Bena, G. Vargas-Solar, N. Bennani, N. Grecchi, C. Ghedira-Guegan, C. A. Ardagna, Trust negotiation in dynamic service-based applications, Preprint submitted to Elsevier (2025). URL: https://github.com/SESARLab/trust-negotiation-dynamic-service-based-applications.

[19] G. Vargas-Solar, N. Bennani, J. A. Espinosa-Oviedo, A. Mauri, J.-L. Zechinelli-Martini, B. Catania, C. A. Ardagna, N. Bena, Decolonizing federated learning: Designing fair and responsible resource allocation, in: Proceedings of the IEEE/ACS International Conference on Computer Systems and Applications (AICCSA), IEEE, 2024.