*Time Complexity*:複数のRLエージェントの訓練実行時間に基づいてアクセスされた（表VI）。この結果は、実環境に近い環境でのRLエージェントのトレーニングが著しく高速であり、RL環境が運用環境におけるRL手法の適用性を高速化できることを強調している。A2CとMaskable PPOは、Deepsets PPOとDeepsets DQNよりもかなり高速である。

*Training*両方の報酬関数について、2000エピソードの学習結果を図3と図4に示す。また、グラフのスパイクを減らすために、200エピソードの平滑化ウィンドウを適用している。遅延報酬関数のDeepSets PPOは、報酬が著しく低い期間があったにもかかわらず、すべてのアルゴリズムが1000エピソード前後に収束するようである。拒否された要求の数については、すべてのアルゴリズムが、その割合が20%以下の状態に収束するようであり、Maskable PPOは0%に達することが可能である。配備コストに関しては、どのアルゴリズムも平均配備コストは6～12に達し、予想通り遅延報酬関数ではわずかに高い値を示した。さらに、トレーニング中、利用可能なクラスタ数は4つに設定される。これは、全体的な展開コストを低く抑えながら、高い割合のリクエストを受け入れることが難しい可能性があることを意味する。最後に、待ち時間に関しては、すべてのアルゴリズムがトレーニング中に待ち時間を大幅に削減することができ、待ち時間報酬関数に対して大幅に低い値（平均50ミリ秒以下）を得ることができた。

*Testing*テストは、すべてのアルゴリズムについて、2000回のトレーニングの後に保存された設定で、100回のエピソードの間に実行された。表 VII は、異なるアルゴリズムについて、累積報酬、拒否されたリクエストの割合、平均展開コスト、および平均待ち時間に関するテスト段階で得られた結果を示している。コストを考慮した戦略では、平均で1100の累積報酬が得られ、3.9%という低い拒否されたリクエストの割合に対応し、4.3ユニットという全体的な展開コストが得られた。レイテンシを考慮した機能では、DeepSetsPPOは平均して0%の拒否率を達成し、11.7ユニット のデプロイメントコストと28.57msの平均レイテンシを達成しました。DeepSets DQNは、平均で5.76ユニットという低いデプロイメント・コストを達成し、拒否率は0.3%、平均待ち時間は42.16msとなります。

*Generalization*クラスター・サイズ[4, ..., 128]を変化させることで、DeepSetsの両アルゴリズムの汎化性が評価された。結果は、DeepSetsニューラルネットワークの大きな可能性を示している。両アルゴリズムは、小規模なセットアップで訓練した場合でも、両方の戦略に対してほぼ最適な割り当てスキームを見つけることができる。クラスタ数が増加するにつれて、レイテンシ目標はコスト目標よりもかなり複雑になる。実験を通してレイテンシは増加するが、両エージェントとも、レイテンシを考慮した場合は100ms以下、コストを考慮した場合は300ms以下と、両戦略に対して適切なレイテンシ値を達成している。DQNはPPOよりわずかに低いレイテンシを達成したが、その代償として拒絶率はほぼ4%であった。結論として、どちらのアルゴリズムも、学習済みのセットアップよりも32倍高いマルチクラスターセットアップでマイクロサービスの配置を最適化できる。

*In summary*要約すると、本稿では、よく知られたK8sプラットフォームと最近のトレンドであるRLに焦点を当て、効率的なマルチクラスターオーケストレーション戦略を調査した。2つの相反する目的により、RLアルゴリズムが累積報酬を最大化する適切なアクションを見つけることができることを実証した。オフラインRL環境では、ほとんどのアルゴリズムが著しく高い性能を達成したため、RLアプローチが検証された。Karmadaのマルチクラスタオーケストレーションソリューションは、全てのレプリカを単一のクラスタに配置するか、複数のクラスタに分散させるかの間で、ほぼ最適なバランスを見つけるので、我々のGTMの恩恵を受けるだろう。このトレードオフは、本稿で示すように、様々な目的に対して見出されている。テスト段階では、すべてのアルゴリズムが両方の戦略で高い報酬を得た。最後に、DeepSetsニューラルネットワークは、その大きな可能性を示した。これらのRLアルゴリズムは、クラスタサイズの異なるマルチクラスタ環境に直接適用でき、学習時間を大幅に短縮できる。DeepSetsがなければ、RLアルゴリズムはその特定のクラスタサイズに対して再トレーニングが必要であり、これはかなりコストがかかる。

*VII. CONCLUSIONS*

本稿では、マルチクラスターシナリオにおけるマイクロサービスの効率的なスケジューリングについて研究する。評価では、本稿で扱うマルチクラスターオーケストレーション問題に対するRLの実行可能性を示す2つの相反する戦略を検討する。結果はまた、典型的なRLアルゴリズムにDeepSetsニューラルネットワークを組み込むことで汎化が達成可能であることを示し、訓練されたものよりも32倍高いシナリオに対して高い性能を達成した。今後の課題として、多目的定式化およびマルチエージェントRLシナリオを研究し、対立するscehduling戦略の最適な組み合わせを見つける予定である。我々の研究は、オープンソースで公開されたフレームワークを提供することで、研究者がスケジューリングのアイデアを評価し、より効率的なスケジューリングアルゴリズムの開発を導くことを可能にし、この分野に貢献している。

*ACKNOWLEDGMENT*

本研究は、MUR Missione 4 - Next Generation EU(NGEU)の資金提供による、Italian Research Center on High-Performance Computing, Big Data and Quantum Computing(ICSC)のSpoke 1 「FutureHPC&BigData 」の一部支援を受けている。Jos´e Santosは、フランダース研究財団（FWO）の助成金（助成番号1299323N）を受けています。