

# POET を用いた輻輳制御アルゴリズムの自動生成における ネットワークの変異手法および難易度評価の提案

佐藤創太, 阿部 洋丈, 岡 瑞起

## イントロダクション

ネットワークではたびたびパケットロスと、それによる再送が起こる。これらの原因の一つとして輻輳が挙げられる。輻輳とは、ネットワークの特定の箇所にトラフィックが集中し、通信速度の低下および通信の断絶が発生することである。

輻輳を回避、緩和しネットワークの通信を円滑に進めるための手法として輻輳制御がある。

この制御手法は通信を正常に保ちつつ、通信量を最大化する。手順としては、各ノードが現在の通信の状況を観察し、送信パケット量を徐々に増やす。しかし、通信経路上の混雑の兆候を検知したら、送信パケット量を減少させ、輻輳を緩和する。輻輳制御において、送信パケット量の増やし方、異常検知時の送信パケット量の調整方法は、輻輳制御アルゴリズムによって規定される。

輻輳制御アルゴリズムの研究は 40 年近く続けられており、現在でも数多くのアルゴリズムが存在する。代表的なものとして NewReno [1] や CUBIC [2] などが挙げられる。これらはシステムチックに設計された規則に基づいて輻輳制御を行っている。しかし、これまでの長年の研究に関わらず、未だにスタンダードとなるアルゴリズムは登場していない。この状況の要因として、人間の力のみでアルゴリズムを開発するには限界があるためだと考えられる。

そこで、本研究では輻輳制御アルゴリズムの自動生成を試みる。自動生成の手法として、強化学習アルゴリズムである POET[3]を用いる。POET とはエージェントと環境が共進化する強化学習アルゴリズムである。輻輳制御においては、エージェントが輻輳制御アルゴリズムにあたり、環境部分がネットワークにあたる。

POET において、エージェントの性能を正確に測るためには、適切に調整された環境を用意することが

必要である。環境が極端に複雑な場合は、ほとんどのエージェント

は十分に性能を発揮できない可能性がある。その逆もまた同様であり、このような環境下では、エージェント間で性能差を見出すことは困難である。環境の複雑さを評価するには、ネットワーク環境の難易度や構成要素を数値化する必要がある。本紙では、B.Tadic らが提唱した、ネットワークのトポロジーと輸送プロセスに関する論文[4]を参考にしつつ、ネットワーク環境のパラメータ化に向けたアプローチを試みる。

## 参考文献

- [1] Y. Nishida, The newreno modification to tcp's fast recovery algorithm, Standards Track, PP, pp. 1–16, 2012.
- [2] S. Ha et al., Cubic: A new tcp-friendly high-speed tcp variant, ACM SIGOPS operating systems review, vol. 42, no. 5, pp. 64–74, 2008.
- [3] R. Wang et al., Paired open-ended trailblazer (poet): Endlessly generating increasingly complex and diverse learning environments and their solutions, 2019.
- [4] B.Tadic et al., Transport on Complex Networks: Flow, Jamming and Optimization, Int. J. Bifurcation and Chaos 17 (7), 2363-2385, 2007.