

コールアドミッション問題におけるトラフィック変動による影響の解析 An Experimental Analysis of Effects by Different Traffic Demands in Call Admission Problem

高東 大樹[†] 上土井 陽子[‡] 吉田 典可[‡]

Hiroki Takatoh Yoko Kamidoi Noriyoshi Yoshida

1 はじめに

近年、ネットワークシステムにおいてオンラインで発生するコールに対して QoS を保証するためにコールアドミッションが注目されている。本稿では、ネットワークシステムにおいてオンラインで発生するコールを受理することにより得られる利益の総和を大きくすることを目的としたコールアドミッション問題について考える。コールアドミッション問題は、ネットワークポロジに着目した研究が多くなされている。しかし、GRID コンピューティングのように通信するコンピュータ間の距離よりもコンピュータの処理能力に応じてトラフィックが変化する場合、トラフィックの変化に着目することが重要であると考えられる。そこで、トラフィックの変化とコールアドミッションアルゴリズムの振舞いとの関係について実験的に解析し、ネットワークを変化させずにトラフィックの変動に柔軟に対応するアルゴリズムについて考察する。

2 オンラインコールアドミッション問題

例えば、テレビ会議システムを考えた場合、逐次的に発生する複数ユーザのコールの競合が問題となる。このようにオンラインに発生するコールに対するコールアドミッション問題をオンラインコールアドミッション問題という。本稿では、スーパーバイザに対してコールがオンラインで発生し、発生したコールにスーパーバイザがネットワーク中の経路を割り当てることで利益が得られるものとする。

i 番目のコールを $r_i = (s_i, t_i, b_i, p_i)$ と表す。ここで、 s_i, t_i はそれぞれ始点と終点である。また、 b_i は i 番目のコールが要求するバンド幅、 p_i は i 番目のコールを受け入れることにより得られる利益である。コールの利益にはさまざまな値が用いられるが、本稿では一度受け入れられたコールは永続するものと仮定し、利益はコールのバンド幅に依存するものとする。このとき、受理したコールの利益の総和を大きくすることを目的として各コールを受理するか拒否するか決定する。

3 コールアドミッションアルゴリズム

枝 e の容量 $u(e)$ に対する負荷の割合 $L_j(e)$ は、

$$L_j(e) = \frac{1}{u(e)} \cdot \sum_{\substack{k \in A_j \\ e \in P_k}} b_k$$

と表される。ここで、 A_j は始めから j 番目のコール迄で受理されたコールの添字の集合、 P_k は $k \in A_j$ なるコールに割り当てられた経路（枝の集合）を示す。

文献 [1] によるオンラインコールアドミッションアルゴリズム ($SAAP_\mu$) では、 j 番目のコールが以下の 2 つの条件をともに満たす場合にコールを受け入れる。

- 枝の負荷による s_j から t_j への最短経路を K とし、

$$\sum_{e \in K} b_j \mu^{L_{j-1}(e)} \leq p_j$$

[†]広島市立大学大学院 情報科学研究科 (Graduate School of Information Sciences, Hiroshima City University)

[‡]広島市立大学 情報科学部 (Faculty of Information Sciences, Hiroshima City University)

- 経路 K に含まれる全ての枝 e に対して、

$$L_{j-1}(e) + \frac{b_j}{u(e)} \leq 1$$

ここで、 μ はユーザが決定する定数で、以下に示す意味をもつ。最短経路上の枝の本数をホップ数とする。 L^* をユーザが希望する限界利用率 (critical utilization) とする。このとき、ホップ数が 2 以上のコールは枝の負荷が L^* を越えるときに拒否したいなら、 μ の値を $\mu = 2^{1/(1-L^*)}$ と設定する。本稿では、枝の容量に空きがあればホップ数が 1 のコールを無条件に受け入れるようにするために、コールの利益 (p) を $p = b\mu$ とする。

4 トラフィックパターン

一般的に、コールは始点と終点の距離が小さなコールほど発生し易い。しかし、GRID コンピューティングのような環境では必ずしも距離の小さなコールほど発生し易いとは限らない。そこで、発生頻度が距離に依存したコール列と距離に依存しないコール列について考慮する必要がある。

入力系列におけるペア (s, t) の分布が距離に依存した系列と距離に依存しない系列 (ランダムな系列) を用いる。どちらの入力系列もコールの発生時刻は期待到着間隔 λ を入力としポアソン過程 [2] を用いて実現する。

距離に依存した入力系列：各ペア (s, t) に対するコールの期待到着間隔 $\lambda_{(s,t)}$ を s, t 間の距離に比例した値とする。

ランダムな入力系列：全てのペア (s, t) に対するコールの期待到着間隔を一定の値とする。

本稿では入力系列に出現するコール数を限定し、かつ、距離による期待到着間隔をもつコール数とランダムなコール数との割合を変化させたいいくつかのトラフィックパターンについて考察する。また、以下では始点、終点のペア (s, t) によりコールの種類を区別する。

5 シミュレーション実験

本節では、ネットワークを変化させずにトラフィックの変動に柔軟に対応するアルゴリズムについて考察するためトラフィックの変化とコールアドミッションアルゴリズムの振舞いの関係について実験する。

$SAAP_\mu$ のアルゴリズムを Sun マイクロシステムズ社 SPARCstation5 上に C 言語を用いて実現した。ネットワークの構成として文献 [3] のモデルを用い、 50×50 のマトリクスを基に節点数 50 のランダムネットワークを作成した。全てのコールのバンド幅を 1、つまり $r_i = (s_i, t_i, 1, \mu)$ とし、各入力系列でのコール数を 5000 個とした。

5.1 実験 1 ($SAAP_\mu$ の基本的な振舞い)

全てのコールが距離に依存した入力系列に対して、枝数を 224 本 (密度 9%)、枝の容量を 25 から 30、 $\mu = 2^{1/(1-L^*)}$ とし、 $SAAP_\mu$ の基本的な振舞いを調べるためにコールの種類数を変えて実験した。コールの種類を 11 種類 (印)、400 種類 (+ 印) としたときの結果を図 1 に示す。

図 1 より、枝の容量を最大限使うことを望むよりも L^* の値を適切に設定して枝の容量に余裕をもたせておいた方が実際の拒否率が小さくなることが確認できる。

また、コールの種類が多いと各コールの始点と終点の組が重なる頻度が減少し、同じ種類のコールが後から到着しても経路数が増えにくくなるため、拒否率が小さくなることも確認できる。

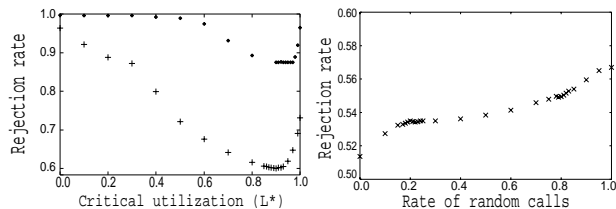


図 1: 拒否率と L^* の値

5.2 実験 2 (拒否率とトラフィックパターンとの関係)

本節では、 $SAAP_\mu$ の拒否率とトラフィックパターンとの関係を探るために実験する。ここで、トラフィックパターンは入力された全てのコールに対するランダムなコールの比率の値により区別するものとする。なお、実験 2 における全ての実験結果での拒否率の値として、 L^* の値を変化させて最小となったときの拒否率の値を採用した。

枝数 234 本 (密度 10%) のネットワークを構成し、コールを 500 種類、 μ の値として $\mu = 2^{1/(1-L^*)}$ とした。このとき、各枝の容量が 25 から 30 とした結果を図 2 に示し、各枝の容量のばらつきを大きくし各枝の容量が 1 から 50 とした結果を図 3 に示す。さらに、各枝の容量が 1 から 50 の場合における、ネットワーク全体の容量に対する受理されたコールの容量の比率 (利用率) 及び L^* の値とトラフィックの変化との関係を図 4 に示す。なお、図 4 における印は L^* の値を示し、×印は利用率を示す。

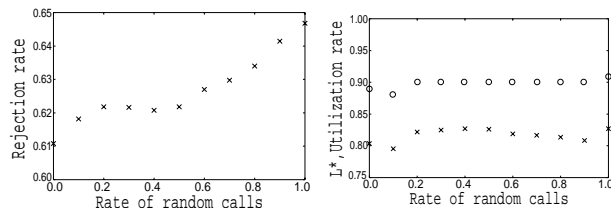


図 3: 実験 2 (枝容量 1-50)

図 2, 3 より、ランダムなコールの割合が低いときと高いときには拒否率が急激に大きくなっており、その間では拒否率はほぼ一定である。つまり、ランダムなコールの割合が増加しても拒否率があまり変化しない領域が存在することが分かる。

図 4 より利用率は L^* の値に依存すると考えられる。また、拒否率が一定となる領域では L^* の値は 0.90 と一定になった。他のネットワークに対しても拒否率が一定となる領域では L^* の値はほぼ一定となった。そこで、このような L^* の値を設定することで、トラフィックが変化しても柔軟に対応できると考えられる。

5.3 実験 2 の考察

トラフィックの変化によって大きく拒否率が変化する領域と、あまり変化しない領域が存在する原因を考察する。なお、ランダムなコールの比率が増加するということは、ホップ数の小さなコールがホップ数の大きなコールに置き換えられることであると考えられる。

まず、ランダムなコールの比率が低い場合、ホップ数の小さなコールが非常に多く、主な置き換えの対象となる。ホップ数の大きなコールの増加にもない L^* より下の領域で受理されていたコールが減少し、拒否率が大きく上昇する。次に、ランダムなコールの比率が中程度の場合、中程度のホップ数のコールが置き換えられる確率が徐々に上昇してくる。このような中程度のホップ数のコールは受理されていない場合もあるため、拒否率はあまり上昇しない。最後に、ランダムなコールの比率が高い場合、置き換えられたコールが

多く存在するため、受理されているコールが置き換えられる確率が再び上がり拒否率も大幅に上昇する。

この考察を確認するため、受理されたコール、拒否されたコールそれぞれにおいて各ホップ数のコールが占める比率の実験を行なう。実験 2 における各枝の容量が 25 から 30 のネットワークに対して受理されたコールの分布に対する結果を図 5 に、拒否されたコールの分布に対する結果を図 6 にそれぞれ示す。棒グラフ中での異なるパターンで記された矩形のそれぞれの高さは図の下から順にホップ数 1, 2, 3, 4 のコールの数量を示している。さらに、一番上側がホップ数 5 以上のコールの数量を示している。

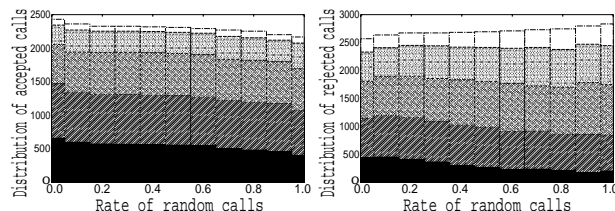


図 5: コール分布 (受理)

図 5, 6 より、拒否率が大幅に上昇する領域では受理されたコールにおけるホップ数の小さなコールが減少し、拒否率がほぼ一定の領域では拒否されたコールにおいてコールの分布が大きく変化していることが確認できる。なお、他のネットワークにおいても同様の結果が得られた。

6 応用

実験 2 より、拒否率が一定となる領域では L^* の値が一定となることが分かった。そこで、このような拒否率が一定となる領域での L^* の値を用いて、拒否率とトラフィックパターンとの関係について実験する。

各枝の容量が 1 から 50 の場合に L^* の値を 0.90 と一定にした結果を図 7 に示す。なお、図中の印は L^* の値を変更して拒否率を最小とした結果、×印は L^* の値を一定とした結果を示す。

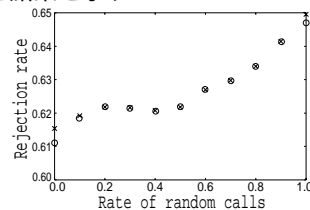


図 7: 拒否率

図 7 より、拒否率が最小となるように L^* の値を設定した場合と同様にトラフィックが変化しても拒否率があまり変化しない領域が存在していることが分かる。

7 まとめ

本稿では、トラフィックの変動に関するコールアドミシヨナルアルゴリズムの性能影響について解析した。シミュレーション実験により、トラフィックの変化によって大きく拒否率が変化する場合と、ほとんど変化しない場合という相異なる特徴が明らかとなり、ネットワークを変化させずにトラフィックの変動に柔軟に対応するアルゴリズムの性能評価について実験的に考察した。今後の課題としては、各枝の容量にばらつきがある場合にネットワークを効率よく使用するための $SAAP_\mu$ アルゴリズムの分散化が挙げられる。

参考文献

- [1] A. Borodin and R. El-Yani: "Online Computation and Competitive Analysis," Cambridge University Press, pp.226-228, pp.234-237, 1998.
- [2] S. M. Ross: "Simulation: Second Edition," Academic Press, pp.74-77, 1996.
- [3] L. Wei and D. Estrin: "The trade-offs of multicast trees and algorithms," Technical Report, Computer Science Department University of Southern California, pp.1-15, 1995.