

M-22 入力系列パターンに着目したコールアドミSSIONアルゴリズムの実験的性能解析
An Experimental Analysis on Call Admission Algorithm
in View of Input Sequence Patterns

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、テレビ会議システムに代表されるように、音声・映像などを複数の参加者で送受する場合、ユーザ毎の利用可能な帯域や受信能力に応じた柔軟な配送を効率良く実現する必要性が生じてきた。このようなユーザ主体のマルチキャスト配信では逐次的に発生する複数ユーザの要求の競合が問題となっている。そこで、サービスの質を落さないためにコールアドミSSIONが重要となってきた。

コールアドミSSION問題とは、各枝が容量をもつネットワーク中に逐次発生するコール(要求)に対して、そのコールを受理するか拒否するかを決定する問題である。本稿ではオンラインのコールアドミSSION問題に対するアルゴリズムの性能と入力系列パターンとの関係について実験的に解析し、応用分野への適用における課題を探る。

2 オンラインコールアドミSSION問題

入力ネットワークは節点集合と各枝 e に容量 $u(e)$ をもつ無向枝集合からなるものとする。 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 は始点、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であり、 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。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コールを受理するか拒否するかを決定する。なお、コールを受理した場合にはネットワーク中の経路を割り当てなければならず、一度受け入れられたコールは永続するものと仮定する。

3 コールアドミSSION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SSION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、

$$\sum_{e \in K} b_j \mu^{L_{j-1}(e)} \leq p_j \text{ を満たす。}$$

- 経路 K に含まれる全ての枝 e に対して、

$$L_{j-1}(e) + \frac{b_j}{u(e)} < 1 \text{ を満たす。}$$

†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 Information Sciences, Hiroshima City University)

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、枝に容量の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 入力系列による性能解析の方針

入力系列におけるコールの始点 s と終点 t のペア (s, t) の分布とコールアドミSSIONアルゴリズムの性能の関係を調べるため、以下に示すパターンを入力系列の候補とする。

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 を入力としてポアソン過程[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した。

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入力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

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、 $\mu = 2^{1/(1-L^*)}$ としたとき、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、コールの種類を11種類(●印)、400種類(+印)としたときの結果を次ページの図1に示す。

図1より、枝の容量を最大限使うことを希望するよりも余裕を持たせておいた方が実際の拒否率が小さくなる事が確認できる。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。

$\mu = x^{1/(1-L^*)}$ とし、コールの種類が11種類における拒否率が最小となるときの拒否率と L^* の値を表1に示す。ここで、表中の括弧内の数値は $x = 2$ とした場合

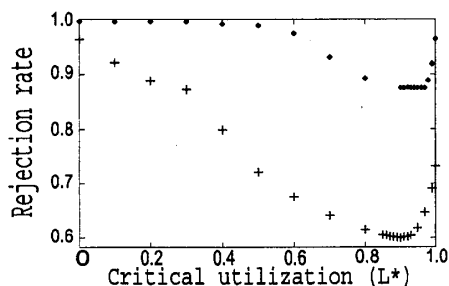


図 1: 実験結果 1

合に拒否率を最小とする μ の値により各 x に相当する L^* を計算した値である。

表 1: 最小拒否率と L^* の値

		L^* の値	拒否率
x	2	0.93	0.875
	3	0.89 (0.889)	0.875
	4	0.85 (0.860)	0.874

表 1 より、実験結果の L^* の値と計算して求めた括弧内の値がほぼ同じであることが確認できる。これは、入力データとネットワークが同じ場合、拒否率が最小のときはパラメータ μ は一定の値となり、アルゴリズムの振舞いとしては同じものとなることを示している。つまり、 $\mu = x^{1/(1-L^*)}$ の x の値として 1 つの数値を用いて拒否率が最小となるときの L^* の値を求めれば、任意の x の値に対して拒否率が最小となる L^* の値を算出することができる。

5.2 実験 2 (入力系列パターンと拒否率との関係)

枝数を 224 本、枝の容量を 25 から 30、コール 500 種類、 $\mu = 2^{1/(1-L^*)}$ とし、ランダムなコールの割合を変化させることにより入力系列パターンと拒否率の関係調べた実験結果を図 2 に示す。

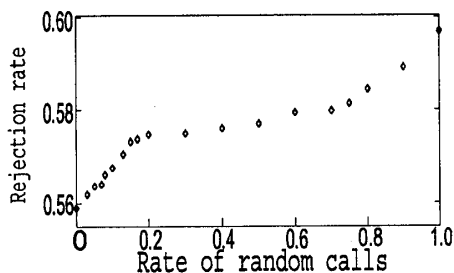


図 2: 実験結果 2

図 2 より、ランダムなコールの割合が増えるにつれて拒否率も大きくなることを確認できる。特に、ランダムなコールの割合が 0 ~ 0.2 のときと、0.7 ~ 1 のときの上昇幅が大きい。この原因として、前者の場合ホップ数の大きいコールが増加し L^* を越えて受理されるコールが減少したためと考えられる。後者の場合はホップ数の大きいコールが非常に増加したためにネットワーク全体の枝の使用率が高くなったためと考えられる。一方、ランダムなコールの割合が 0.2 ~ 0.7 の場合には拒否率の上昇幅が小さい。これは、ホップ数の大きい

コールが増加し L^* を越えて受理されるコールは減少したが、一方で L^* までの容量に空きがある枝が存在するためと考えられる。

この関係についてより詳しく検討するため、図 2 で示されるコールが 500 種類の場合を基準とし、ネットワーク全体の容量に対する使用率が高い場合と低い場合を調べる。なお、ネットワーク全体の容量に対する使用率を変更するために、各枝の容量を変えるものとする。つまり、使用率を高くするためには枝の容量を小さくし、使用率を低くするためには枝の容量を大きくするものとする。

前の実験と同様に、枝数 224 本 (密度 0.09) の低密度のネットワークを構成し、コールの種類を 500 種類、 μ の値として $\mu = 2^{1/(1-L^*)}$ とした。各枝の容量を 3 から 8 とした場合を図 3 に各枝の容量を 65 から 70 とした場合を図 4 にそれぞれ示す。

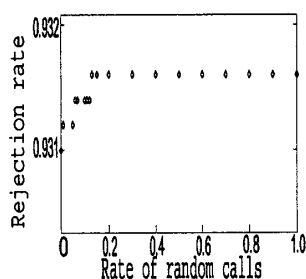


図 3: 実験結果 3-1

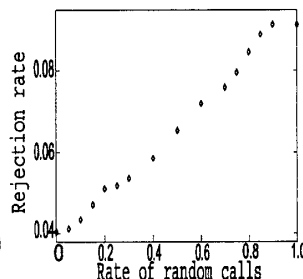


図 4: 実験結果 3-2

図 3 のように枝の容量を小さくして拒否率が大きくなるようにすると、ホップ数の大きいコールの増加に伴い L^* を越えて受理されるコールが減少し拒否率が大きくなる。その後、さらにホップ数の大きいコールを増加させることによりネットワーク全体の枝の使用率が高まり拒否率がさらに大きくなる。しかし、もともと枝の容量が少ないため各枝の容量が限界となり拒否率が平衡状態になることが確認できる。

次に、図 4 のように枝の容量を大きくして拒否率が小さくなるようにすると、ランダムなコールの割合の増加により L^* を越えて受理されるコールが減少し拒否率が大きくなる。さらにランダムなコールの割合を増加させると、 L^* を越えて受理されるコールは減少したが、一方で L^* までの容量に空きがある枝が存在するために拒否率が一時的に一定となることが確認できる。

6 まとめ

本稿では、入力系列パターンによりコールアドミッショナルゴリズムの解の拒否率が変化し、その値の上昇幅にも差があるという実験結果を得た。この結果から、応用分野においては拒否率が一定あるいは平衡状態となる性質を利用することが有効であると考えられる。拒否率以外の項目に注目した場合についても考察することが今後の課題である。

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, p.5, 1995.