

隠れ端末の影響を調査するためのネットワーク生成アルゴリズム による MAC プロトコル評価

重安 哲也^{†a)} 松野 浩嗣^{††} 森永 規彦[†]

Performance Evaluation of Media Access Control Protocols Using an Algorithm
Generating Graphs for Effect Estimation of Hidden Terminals

Tetsuya SHIGEYASU^{†a)}, Hiroshi MATSUNO^{††}, and Norihiko MORINAGA[†]

あらまし 隠れ端末は CSMA をはじめとする競合方式の MAC プロトコルのスループットを著しく低下させてしまう。そのため、現在まで様々な隠れ端末対策を施した MAC プロトコルが多く提案されている。本論文では、競合方式の MAC プロトコルにおける隠れ端末の影響を評価するため、様々な割合で隠れ端末を含む評価用ネットワークを生成するアルゴリズムを提案する。また、同アルゴリズムを利用して代表的な競合方式の MAC プロトコルである CSMA と、代表的な隠れ端末対策の RTS/CTS 制御を提案した MAC プロトコルである MACA の両プロトコルにおいて、隠れ端末の存在割合の変化に対するスループット特性への影響を明らかにした結果を報告する。

キーワード 隠れ端末問題, CSMA, MACA, ネットワーク生成アルゴリズム

1. ま え が き

端末間のパケット送信を集中的に管理・制御する中央局が存在しないネットワークでは、各端末が自律分散的に送信権を確保できることや、端末の移動、追加、離脱などによるトポロジーの変化に素早く対応できることなどが必要である。

代表的な競合方式の MAC (Media Access Control) プロトコルである CSMA (Carrier Sense Multiple Access) [1] やその改良型のプロトコルは、このような要件を満たすプロトコルとして現在までに広く使用されている。中でも、現在最も広く利用されている無線 LAN の標準規格である IEEE802.11 [2] では CSMA/CA (CSMA with Collision Avoidance) が採用されている。

CSMA 型のプロトコルでは自律的な送信制御のために、自端末の送信開始前にキャリアセンスで周囲の端末の送信状況を調査する。その結果、どの端末も送信を行っていないなければ自端末の送信を開始するが、そうでなければ同時送信によるパケット衝突を避けるため、一定時間の送信延期処理を実行する。CSMA 型のプロトコルでは、このように単純な方式で自律分散制御を実現しているが、同方式では、通信範囲外に位置する端末の送信状況は把握できないという問題点が生じる。そのため、他の端末が送信中であるにもかかわらず、新たな送信を開始することにより、パケット同士が衝突を起こすことがある。これは隠れ端末問題と呼ばれ、競合方式の MAC プロトコルにおいて、スループットを著しく低下させる原因となっている [3]。

現在まで、隠れ端末の影響を受けにくい MAC プロトコルとして、様々な方式が提案されている。MACA (Multiple Access with Collision Avoidance) [4] は隠れ端末問題の原因となるキャリアセンスの代わりに RTS/CTS (Request To Send / Clear To Send) 制御を導入したパケットセンス型のプロトコルである。MACA はデータの送信前に、送受信端末間において RTS, CTS をやり取りする。このとき、受信端末の

[†] 広島国際大学情報通信学科, 呉市

Department of Information Technology, Hiroshima International University, 5-1-1 Hiro-koshingai, Kure-shi, 737-0112 Japan

^{††} 山口大学理学部自然情報科学科, 山口市

Faculty of Science, Yamaguchi University, 1677-1 Yoshida, Yamaguchi-shi, 753-8512 Japan

a) E-mail: sigeyasu@it.hirokoku-u.ac.jp

通信範囲内に存在する端末は、受信端末から送信される CTS パケットを傍受することにより新たな送信の開始を知り、一定時間自端末の送信を延期することができる。

また、MACAW (MACA for Wireless)[5] では MACA のメカニズムに ACK (Acknowledgement) を用いることで MAC 層でのデータ回復機能を実現している。一方、MACA-BI (MACA By Invitation)[6] では、RTS/CTS 交換の失敗回数を低減するために、受信端末から送信を促す RTR (Ready To Receive) を導入した受信局主導型プロトコルの提案を行っている。逆に、受信局主導型では送信遅延が増加するとし、FAMA (Floor Acquisition Multiple Access)[7] では RTS/CTS のパケット長に条件を加え、受信局主導型のプロトコルでデータの衝突を完全に防ぐ方法を導入している。

以上のように、現在までに隠れ端末対策用の MAC プロトコルが数多く提案されているが、これらのプロトコルの隠れ端末に対する耐性評価には、様々なネットワークモデルが使用されている。

例えば、文献 [5] ~ [7] は隠れ端末評価のために妥当であると考えられるネットワークを独自に構成し評価に使用している。隠れ端末に対する耐性評価では、パケット衝突の発生しやすさ、すなわち隠れ端末の存在割合などを様々な変化させたネットワークを用いることにより評価の質を向上できると考えられる。しかしながら、このネットワーク生成手法では、アルゴリズムを用いた機械的な方法でないため、多くのネットワークを準備するには多くの時間を必要としてしまう。

また、文献 [8], [9] は、一定の範囲内に端末をランダムあるいはメッシュ状に配置することで評価用ネットワークを生成している。この方法は、前述の方法と逆に短時間で多くのネットワークを生成できるが、隠れ端末を意識していないため、隠れ端末の影響を直接的に反映するネットワーク以外のものも多く生成されてしまい、効率的な評価が行えない。

そこで、本論文では隠れ端末の影響を定量的に判断できる評価用ネットワーク生成アルゴリズムの提案を行う。同アルゴリズムを利用すれば、様々な割合で隠れ端末が存在するネットワークを作成し、スループットへの影響を容易に明らかにすることができる。また、同アルゴリズムでは、隣接端末の偏りによる送信機会の不公平さ [10] などの問題を生じることなく、隠れ端末の存在割合がスループットへ及ぼす影響のみを

純粋に評価するために、どの端末に着目した場合でも、その隣接端末数や隠れ端末数が偏ることのないネットワークを生成する。

本論文の構成は以下のとおりである。まず無線ネットワークをグラフに対応づけた場合に隠れ端末となるノードの定義を与える。次に、隠れ端末を様々な割合で含むグラフを生成するアルゴリズムを提案する。その後、代表的な競合方式の MAC プロトコルである CSMA と代表的な隠れ端末対策である RTS/CTS 制御を提案した MAC プロトコルである MACA の両プロトコルを取り上げ、隠れ端末の存在割合の変化が両プロトコルのスループット特性へ及ぼす影響を提案アルゴリズムにより明らかにした結果を報告する。

2. 無線ネットワークの表現方法と隠れ端末問題

2.1 無線ネットワークの表現方法

本論文では、無線ネットワークを文献 [3] で定義される hearing graph と同様の手法で表現する。まず、ネットワーク中の各端末をグラフ $G = (V, E)$ のノードに対応させる (ここで V はノードの集合、 E はエッジの集合である)。次に、任意の二つの端末間で双方向に通信が可能であるときかつそのときに限りこれらに対応するノード u, v 間にエッジを引く。上記の手法を用いた場合の無線ネットワーク表現の例を図 1 に示す。同図 (a) において、破線で示される通信範囲をもつ 5 端末から構成される無線ネットワークは、同図 (b) に示されるグラフで表現される。

無線ネットワーク中のノード間の距離について以下の定義を加える。

[定義 1] グラフ $G = (V, E)$ 中の任意の二つのノ

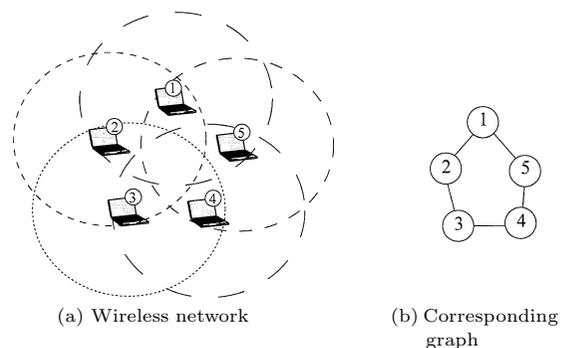


図 1 グラフを用いた無線ネットワーク表現

Fig. 1 Graph representation of a wireless network.

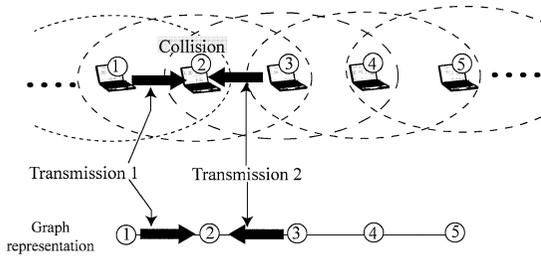


図 2 隠れ端末問題
Fig. 2 Hidden terminal problem.

ド $x_1, x_2 \in V (x_1 \neq x_2)$ に対し, $(x_1, x_2) \in E$ であれば, ノード x_1 と x_2 は隣接しているという.

[定義 2] グラフ $G = (V, E)$ 中の任意の二つのノード $x_1, x_2 \in V$ を考える. このとき, $(x_1, y), (y, x_2) \in E$ であり, $(x_1, x_2) \notin E$ であるような V 中のノード $y \neq x_1, x_2$ が存在する場合かつそのような場合に限りノード x_1 はノード x_2 から 2 ホップ離れているという.

2.2 隠れ端末問題と並行送信

2.2.1 隠れ端末問題

Kleinrock らによって提案された CSMA [1] は送信前のキャリアセンスにより, パケットの同時送信を控えるが, 通信範囲外の端末の送信を把握することができない. そのため, これらの端末と同時に送信を行うことによりパケット衝突が起こる. これは隠れ端末問題と呼ばれ, 競合方式の MAC プロトコルのスループットを著しく低下させる問題としてよく知られている.

隠れ端末問題によるパケット衝突の例とそのグラフ表現を図 2 に示す. このネットワークでは, 端末 1 が端末 2 に送信 (Transmission 1) を行ったとしても, 端末 3 はこれをキャリアセンスで知ることができない. そのため, 端末 3 は端末 2 に新たな送信 (Transmission 2) を開始してしまう可能性がある. その場合, 端末 2 上で二つの送信パケットは衝突し, 結果としてどちらの送信も失敗する. このとき, 端末 1 (3) は端末 3 (1) の隠れ端末であるという.

以上のことから, 隠れ端末を次のように定義する.

[定義 3] グラフ $G = (V, E)$ 中の二つのノード x_1, x_2 間に次の条件が成り立つとき, ノード x_1 はノード x_2 の隠れ端末であるという.

- (i) x_1 と x_2 は 2 ホップ離れている.
- (ii) ノード x_1 とノード x_2 の両方に隣接しているあるノード y に対して, ノード x_1 がデータを送信している.

2.2.2 隠れ端末問題と平行送信

隠れ端末の発生する状況をより詳しく検討するため, 任意の送信端末に着目した場合に, その他の端末を送信端末からのホップ数により次のように分類してみる.

- (1) 送信端末から 1 ホップ
- (2) 送信端末から 2 ホップ
- (3) 送信端末から 3 ホップ以上

送信端末がキャリアセンスにより送信状態を参照するのは上記 (1) に属する端末である. そのため, 送信端末はこれらの端末と同時刻に送信は行わず, パケット衝突を回避できる^(注1).

一方, (2), (3) に属する端末は, とともに送信端末から送信状況を参照されることはないが, (2) に属する端末は, 状況によって隠れ端末となる可能性がある. 逆に, (3) に属する端末はどのような状況においても送信端末の送信を妨害することはない. このことは, 送信端末として図 2 の端末 1 に着目した場合, 端末 4, 5 が (3) に属する端末となるが, これらは端末 1 と互いに影響を全く及ぼさないことから分かる. そのため, MAC プロトコルにおける隠れ端末の影響を明らかにするには, この (3) に属する端末を排除したネットワークで評価を行うことが望ましい.

文献 [11] では, 上記の (3) に属する端末を排除したネットワークにおいて, (1), (2) に属する端末の存在割合を変化させたプロトコル評価を行っている. その結果, (2) に属する端末は, 隠れ端末となりパケット衝突を引き起こす一方で, 並行送信により, スループットを向上させる場合もあると報告している. これと同様の現象を図 1 で説明できる. 同図は, どの端末に着目した場合にも, その他はすべて (1), (2) のいずれかに属するネットワークである. 今, 端末 1 が 2 に送信を開始したとする. 端末 1 から 2 ホップの端末は, 端末 3, 4 であるが隠れ端末となるのは端末 3 であり, 端末 4 は端末 3 に向けて送信を行ったとしてもパケットは衝突せずに, 正しく受信される. したがって, 図 1 (b) の場合, 前述のような並行送信が成功してしまい, 隠れ端末のみの影響を正確に評価できなくなってしまふ.

本論文で提案するアルゴリズムはこのような場合をすべて排除したネットワークを生成することができる.

(注1): これらの端末が送信を開始してから, 端末 c にパケットの先頭が到着するまでの時間内 (すなわち, 伝搬遅延時間内) に端末 c が送信を開始する場合はキャリアセンスを用いても衝突を回避できないが, ここではそれを考えない.

3. 評価用ネットワーク生成アルゴリズム

以下は、グラフ中の任意のノード v に対して、その他のすべてのノードが隣接ノードか、隠れ端末のどちらかであるようなグラフ $G = (V, E)$ を生成するアルゴリズムである。あるノードの集合 S が存在し、アルゴリズムにおいて、 $H(h) \subseteq S$ と記述した場合、集合 $H(h)$ は h ($1 \leq h \leq |S|$)^(注2)個の要素をもつ集合 S の部分集合であるとする。

[アルゴリズム 1]

入力： $n = p(h + 1)$ を満たすような自然数 n 個から構成されるノード集合 U 。ただし、 p は 2 以上の整数。
 出力：任意のノード $v \in V$ に対し、 n 個のノードが v の隠れ端末であり、残りの $n - h - 1$ 個のノードがすべて v の隣接ノードであるような n 個のノードから構成されるグラフ $G = (V, E)$

```

begin
  E = ∅;
  V = ∅;
  while |U| ≥ 2(1 + h) do
    begin
      v ∈ U;
      H(h) ⊆ (U - {v});
      for each x ∈ (U - H(h) - {v}) do
        begin
          E = E ∪ {(v, x)};
          for each u ∈ H(h) do
            E = E ∪ {(u, x)};
          end;
          U = U - {v} - H(h);
          V = V ∪ {v} ∪ H(h);
        end
      end
    end
  V = V ∪ U
end
    
```

提案アルゴリズムのフローチャートを図 3 に示す。また、 $h = 2, n = 6$ として、アルゴリズムにより生成したグラフの例を図 4 に示す。

以下の定理はこのアルゴリズムの正当性を証明している。

[定理 1] $n = p(h + 1)$ ($p \geq 2$) を満たす整数 n と h が与えられたとき、アルゴリズムは次のような n 個のノードから構成されるグラフ $G = (V, E)$ を生成す

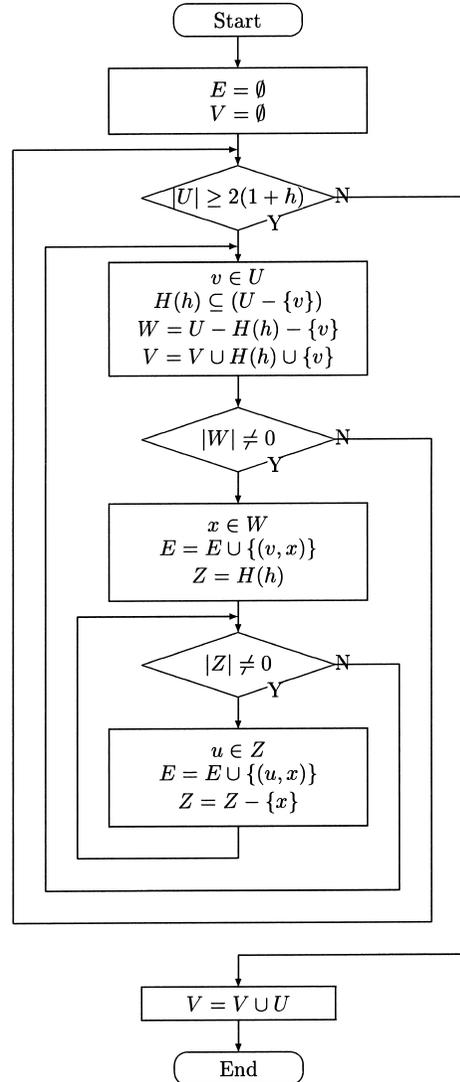


図 3 提案アルゴリズムのフローチャート
 Fig. 3 A flowchart of proposed algorithm.

る；任意のノード $v \in V$ に対して、隠れ端末の個数は h であり、その他のすべてのノード、つまり $n - h - 1$ 個のノードは、 v に隣接している。

(証明) 整数 p に対する帰納法により定理 1 を証明する。

$p = 2$ の場合：この場合はアルゴリズム中の while ループは一度のみ実行されることに相当する。集合 V から任意にノード v を選択し、以下のように集合 $H(h)$ を構成する。

(注2): 任意の集合 A に対し、 $|A|$ は集合 A の要素数を表す。

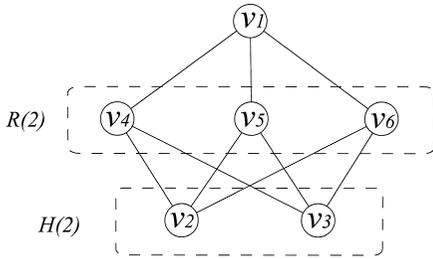


図 4 $n = 6, h = 2$ のときにアルゴリズムによって生成されるグラフの例

Fig. 4 A example of graph produced by the proposed algorithm in the case of $n = 6$ and $h = 2$.

$$H(h) = \{u_1, u_2, \dots, u_h\} \subseteq (V - \{v\}).$$

更に、集合 $R(h)$ を以下のように構成する。

$$R(h) = V - H(h) - \{v\}.$$

ここで、 $|H(h)| = h$ であるので、 $|R(h)| = n - h - 1 = 2(h + 1) - h - 1 = h + 1$ となる。

以下では、集合 $\{v\} \cup H(h) \cup R(h) (= V)$ に属するすべてのノード x に対し、 x から 2 ホップ先のノードが h 個存在すること、及び、 x の隣接ノードが $n - h - 1 = h + 1$ 個存在することを示す。

(1) ノード v に対して以下が成り立つことは、アルゴリズムの作り方より明らかである。

- $H(h)$ のすべてのノードは、 $R(h)$ のどのノードを経由しても、ノード u_i より 2 ホップで到達することができる。また、
- $R(h)$ のノードはすべて、ノード v の隣接ノードである。

(2) 集合 $H(h)$ の任意のノード $u_i (1 \leq i \leq h)$ に対し、

- ノード v はノード u_i から 2 ホップ離れていることは明らかであり、
- $j \neq i$ であるようなノード $u_j \in H(h)$ (つまり、ノード u_j は $H(h) - \{u_i\}$ に属するノード) は $R(h)$ 中のどのノードを経由した場合にもノード v より 2 ホップで到達する。また、
- 集合 $R(h)$ 中のすべてのノードはノード u_i の隣接ノードである。

(3) 集合 $R(h)$ 中の任意のノード $r_i (1 \leq i \leq 1+h)$ に対し、

- ノード v はノード r_i と隣接しており、
- 集合 $H(h)$ 中のすべてのノードはノード r_i と隣接している。また、
- $j \neq i$ となるようなノード $r_j \in R(h)$ (つまり、 r_j

は集合 $R(h) - \{r_i\}$ の要素) は集合 $H(h) + \{v\}$ 中のどのノードを経由した場合にもノード r_i より 2 ホップで到達できる。

以上より、 $p = 2$ の場合において、定理が成り立つことが分かる。

$p = k$ から $p = k + 1$ の場合： $p = k$ の場合に定理が成り立つと仮定し、 $p = k + 1$ の場合についてもこれが成立することを示す。 $|V| = (k + 1)(h + 1)$ であるようなグラフ $G = (V, E)$ を考える。このとき集合 $H(h)$ を以下のような集合とする。

$$H(h) = \{u_1, u_2, \dots, u_h\} \subseteq (V - \{v\}).$$

更に、集合 $R(h)$ を以下のように定義する。

$$R(h) = V - H(h) - \{v\}.$$

ここで、 $|H(h)| = h$ であるので、 $|R(h)| = n - h - 1 = (k + 1)(h + 1) - h - 1 = k(h + 1)$ となる。

$|R(h)| = k(h + 1)$ 並びに仮定条件より、ノード集合が $R(h) (|R(h)| = k(h + 1))$ であるような次の①、②を満たすグラフを構成することができる；①各ノード $v \in R(h)$ に対し、このノードから 2 ホップ離れたノードが h 個存在する、②他の $k(h + 1) - h - 1$ 個のノードは v の隣接ノードである。

以下では、集合 $\{v\} \cup H(h) \cup R(h) (= V)$ 中のどのノード x に対しても、 x から 2 ホップの端末は h 個であり、また、ノード x は $n - h - 1 = k(h + 1)$ 個の隣接ノードをもつことを示す。

(1) ノード v について以下のことが成り立つのは、アルゴリズムの作り方より明らかである。

- 集合 $H(h)$ 中の任意のノードには集合 $R(h)$ 中のどのノードを経由してもノード v から 2 ホップで到達できる。
- 集合 $R(h)$ 中のすべてのノードはノード v と隣接している。

(2) 集合 $H(h)$ 中のノード $u_i (1 \leq i \leq h)$ の任意のノードに対して、

- ノード u_i は、ノード v から 2 ホップ離れていることは明らかである。
- $u_j \in H(h)$ かつ $i \neq j$ (つまり、ノード u_j は集合 $H(h) - \{u_i\}$ の要素) であるどのノードに対しても集合 $R(h)$ 中のノードを経由して、ノード u_i から 2 ホップで到達できる。

• 集合 $R(h)$ 中のすべてのノードは、ノード v 並びに集合 $H(h)$ 中のすべてのノードと隣接している。

(3) 集合 $R(h)$ に対する上記の (1) と (2) の条件よ

り、集合 $R(h)$ の任意のノード r_i に対し、 $R(h) - \{r_i\}$ 中の h 個のノードはノード r_i から 2 ホップ離れており、残りのすべてのノードはノード r_i と隣接している。このことと、集合 $R(h)$ のすべてのノードはノード v と隣接している、集合 $H(h)$ 中のどのノードも r_i に隣接しているという二つの事実より、集合 $R(h)$ 中の任意のノードに対し、グラフ G のノードのうち h 個がこのノードと 2 ホップ離れており、他のノードはこのノードに隣接していることがいえる。

以上より、 $p = k + 1$ の場合においても本定理が成り立つことが明らかとなった。 □

4. CSMA と MACA における隠れ端末の影響調査

提案アルゴリズムを使用し、計算機シミュレーションにより競合方式の MAC プロトコルの隠れ端末に対する耐性を評価する。以下では、代表的な競合方式の MAC プロトコルである CSMA と、代表的な隠れ端末対策である RTS/CTS 制御を提案した MAC プロトコルである MACA の両プロトコルの結果を報告する。

4.1 計算機シミュレーション諸元

計算機シミュレーションでは、トラフィック G をデータパケット長当りの新しく生じたパケットと再計画されて生じたパケットの総和とする（このとき、生じたパケットはすべてが送信されるとは限らず、生じた際にチャンネルがビジーであると判断された場合、そのパケットは送信されずに再計画される^(注3)）。なお、本シミュレータでは、ネットワーク全体で 5,000 パケットの受信が成功するまで、同一のパケット生成レートでシミュレーションを継続する。このとき、シミュレーションに要した時間をもとに以下のようにトラフィック、スループットを算出し、グラフ中の 1 プロットを求める。

- $Throughput = P_{success} \times P_{length}/t$
- $TrafficG = P_{generate} \times P_{length}/t$

(ただし、 $P_{success}$ = 受信に成功したパケットの総数 = 5,000, $P_{generate}$ = 生じたパケットの総数, P_{length} = データパケット長 = 1.0, t = シミュレーションの所要時間である。)

データパケット長で正規化された伝搬遅延時間 (a)、RTS 認識時間 (β)^(注4) がそれぞれ、0.01, 0.005 の場合における CSMA と MACA のトラフィック-スループット特性に関して、文献 [1], [12] で与えられる理論式とシミュレーションの両方により得られた結果を図 5、

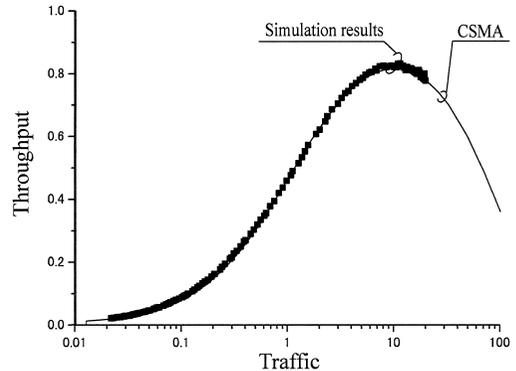


図 5 CSMA の理論式とシミュレーション結果 $a = 0.01$
Fig. 5 Throughput performances of CSMA obtained by theoretical equation and computer simulation. ($a = 0.01$)

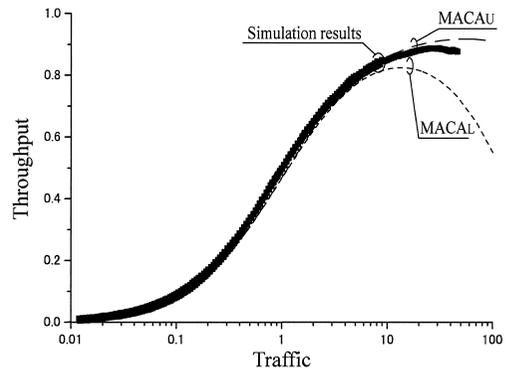


図 6 MACA の理論式とシミュレーション結果 ($a = 0.01$, $\beta = 0.005$)
Fig. 6 Throughput performances of MACA obtained by theoretical equation and computer simulation. ($a = 0.01$, $\beta = 0.005$)

図 6 にそれぞれ示す。ここで、伝搬遅延時間は、送信端末がパケットの送信を開始してから、受信端末にパケットの先頭が到達するまでの時間であり、この長さは CSMA, MACA の両プロトコルの性能に影響する。一方、RTS 認識時間は受信端末が RTS パケットが到着した時点から、それが RTS であると認識されるまでの時間であり、この長さは MACA の性能に影響する [12]。図 6 における $MACA_U$ と $MACA_L$ は文献 [12] で求められた MACA のスループットの上限値、下限値をそれぞれ示している。そのため、実際の

(注3): これは、CSMA, MACA の性能を解析的に導出した文献 [1], [12] と同じ条件下でシミュレーションを行うためである。

(注4): 文献 [12] では RTS 認識時間を記号 x で表現しているが、本論文は既に同記号をアルゴリズム 1 中で使用しているため、ここでは記号 β を用いる。

スループット特性は両曲線の間中値をとることとなる。両図において、理論式とシミュレーション結果が一致していることが確認できる。

4.2 隠れ端末の影響調査

提案アルゴリズムにより生成されたネットワークでは、すべての端末が h 個の隠れ端末をもつと同時に、この h 個の端末同士もまた互いに隠れ端末となるよう構成される(図 4)。そのため、ある端末 v が隠れ端末とパケット衝突を起こした場合にも、残りの $h-1$ 個の隠れ端末にこれが検知されることはない。したがって、隠れ端末によるパケット衝突は最大で連続 h 回発生し、衝突が終了するまでチャネルはビジーとなる。このように、隠れ端末によって引き起こされるビジー区間の最大値は h の値に左右されてしまう。

そこで、本論文では隠れ端末の存在割合のみの影響を純粹に評価するため、 h を固定し、 h_r のみを変化させて評価を行う。また、計算機シミュレーションでは、端末の振舞いを詳細にシミュレーションできることから、ノード数が多くなるに伴いシミュレーションの質も向上するが、同時にシミュレーション時間も長くなる。そこで、今回は扱うネットワークの最大ノード数を 210 とし、アルゴリズムの入力 $n = p(h+1)$ のうち $h = 10$, p を 2, 3, 4, 5, 10, 20 とし、ネットワークを 6 種類生成した。このようにして生成したネットワークに占める隠れ端末の割合 h_r は

$$h_r = \frac{h}{n-1}$$

により、それぞれ 0.48, 0.31, 0.23, 0.18, 0.09, 0.04 となる。以下では、これらの 6 種類のネットワークに隠れ端末が存在しないネットワーク^(注5)を加えた計 7 種類のネットワークにおいて評価を行う。また、隠れ端末の存在割合以外にプロトコルの性能を左右するパラメータとして、伝搬遅延時間 (a), RTS 認識時間 (β) を変化させる。以下では、隠れ端末が存在しない状況での両プロトコルの性能を評価した文献 [12] と比較するために、同文献と同じ条件である伝搬遅延時間 $a = 0.1, 0.01$, RTS 認識時間 $\beta = 0.05, 0.005$ の場合の結果について報告する。

(a) 伝搬遅延時間 $a = 0.1$, RTS 認識時間 $\beta = 0.05$ の場合

この場合の結果を図 7, 図 8 に示す。図 7 より、CSMA では、ネットワーク中にわずかな隠れ端末しか存在しない場合でも、隠れ端末なしの場合と比較して大幅にスループットが低下することが分かる。これに

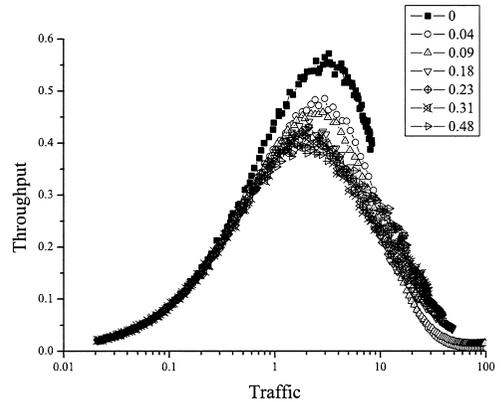


図 7 隠れ端末の割合が CSMA のスループット特性へ及ぼす影響 ($a = 0.1$ の場合)

Fig. 7 Effect of hidden terminal ratio on throughput performance of CSMA. ($a = 0.1$)

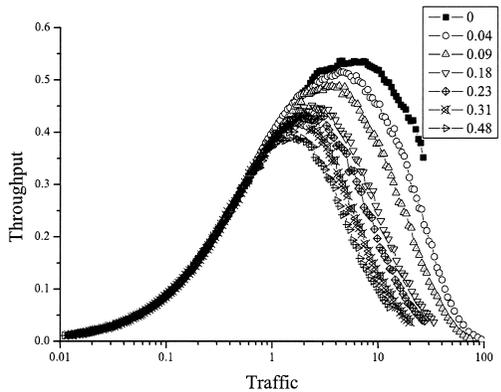


図 8 隠れ端末の割合が MACA のスループット特性へ及ぼす影響 ($a = 0.1, \beta = 0.05$ の場合)

Fig. 8 Effect of hidden terminal ratio on throughput performance of MACA. ($a = 0.1, \beta = 0.05$)

対し、MACA は隠れ端末の増加に対し、スループットは比較的緩やかに低下することが図 8 より分かる。これらと同様の傾向は図 9 から確認することができる。同図は隠れ端末の存在割合が 0 のときの各々のプロトコルの最大スループットでその他の存在割合の最大スループットを正規化して求めた、最大スループットの変化である。この結果から、CSMA では、隠れ端末の存在割合が 4% の場合であっても、最大スループットは存在割合が 0 の場合の 85% 程度に低下する。一方、MACA が同程度に低下するのは隠れ端末の存

(注5): 隠れ端末が存在しないネットワークは完全グラフとして表現することができる。また、このようなネットワークでのスループット-トラフィック特性は文献 [1], [12] で与えられる理論式と一致することに注意されたい。

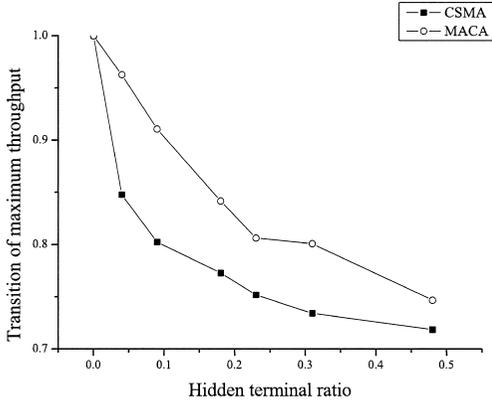


図 9 隠れ端末の割合と正規化された最大スループットの变化 ($a = 0.1, \beta = 0.05$ の場合)

Fig. 9 Transition of normalized maximum throughput by various hidden terminal ratios. ($a = 0.1, \beta = 0.05$)

在割合が 18% の場合であることが分かる。

次に、図 7、図 8 のスループット値を比較する。隠れ端末が存在しない場合の CSMA, MACA の両プロトコルの最大スループットは、それぞれ約 0.6, 0.5 と MACA は CSMA に劣る。これは、MACA の RTS, CTS 交換のオーバーヘッドが大きいためであると考えられる。そのため、MACA の隠れ端末の増加によるスループット低下は CSMA のそれと比べて小さいが、どちらのプロトコルも隠れ端末が存在する場合の最大スループットに大きな差は確認できない。

以上より、 $a = 0.1, x = 0.05$ の条件において、隠れ端末の増加に対し MACA は CSMA よりもスループット低下を低く抑えるものの、RTS/CTS 制御のオーバーヘッドが大きいため、MACA の隠れ端末対策としての有効性は確認できなかった。

(b) 伝搬遅延時間 $a = 0.01$, RTS 認識時間 $\beta = 0.005$ の場合

この場合の結果を図 10, 図 11 に示す。図 10 より、CSMA は図 7 と同様にわずかな隠れ端末しか存在しない場合にも、大幅にスループットが低下している。一方、MACA も図 8 と同様に、隠れ端末の存在割合の増加に対して、緩やかにスループットが低下している。図 12 の最大スループットの変化を表す結果からも、MACA は CSMA よりも隠れ端末の増加によるスループット低下を低く抑えていることが分かる。

次に、図 10, 図 11 のスループット値で両プロトコルを比較する。隠れ端末が存在しないネットワークにお

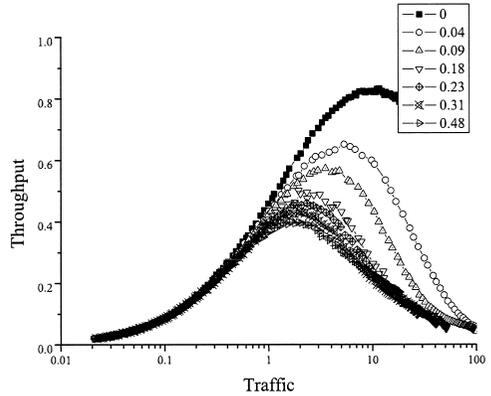


図 10 隠れ端末の割合が CSMA のスループット特性へ及ぼす影響 ($a = 0.01$ の場合)

Fig. 10 Effect of hidden terminal ratio on throughput performance of CSMA. ($a = 0.01$)

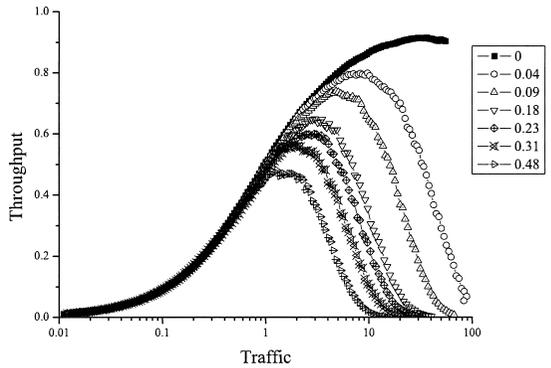


図 11 隠れ端末の割合が MACA のスループット特性へ及ぼす影響 ($a = 0.01, \beta = 0.005$ の場合)

Fig. 11 Effect of hidden terminal ratio on throughput performance of MACA. ($a = 0.01, \beta = 0.005$)

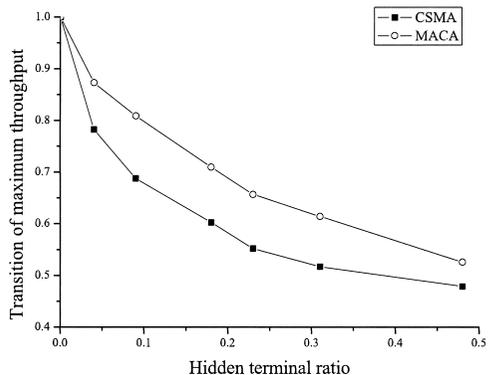


図 12 隠れ端末の割合と正規化された最大スループットの変化 ($a = 0.01, \beta = 0.005$ の場合)

Fig. 12 Transition of normalized maximum throughput by various hidden terminal ratios. ($a = 0.01, \beta = 0.005$)

ける最大スループットは両プロトコルとも大きな差はない。そのため、その他の場合のスループット値では、隠れ端末の増加に対してスループット低下を緩やかに抑える MACA が CSMA よりも高いスループットを示す。これについて、(a)、(b) の両シミュレーション実験で用いたパラメータを対比してみる。どちらの場合も、 $a : \beta = (a) 0.1 : 0.05 = (b) 0.01 : 0.005 = 2 : 1$ から分かるように伝搬遅延時間に対する RTS 認識時間間の相対的な大きさは同じであるが、 a 及び β とともにデータパケット長で正規化しているので、(b) は (a) に比べてより長いデータパケットを送信している場合に相当する。隠れ端末が存在しない環境における MACA の性能を解析した文献 [12] では、MACA は比較的大きなデータパケットを送信する場合に RTS/CTS 制御のオーバーヘッドが小さくなり、プロトコル性能を發揮すると報告しているが、隠れ端末が存在する場合も、これと同様の特性を (a)、(b) のシミュレーション結果が示していると考えられる。

以上より、 $a = 0.01$ 、 $\beta = 0.005$ の条件、すなわち、データパケット長がより長い条件下においては、隠れ端末の増加に対し MACA は CSMA よりもスループット低下を低く抑えることが分かる。また、比較的大きなデータパケットを送信するために、RTS/CTS 制御のオーバーヘッドは小さくなり、結果として、MACA は隠れ端末対策として効果的に機能することが確認できた。

5. む す び

本論文では、競合方式の MAC プロトコルのスループット性能を著しく低下させる隠れ端末問題の影響を明らかにするための評価用ネットワーク生成アルゴリズムの提案を行った。

次いで、同アルゴリズムを利用して、CSMA と MACA の隠れ端末の影響を明らかにするための計算機シミュレーションを行い、RTS/CTS を用いて隠れ端末対策を行う MACA は、隠れ端末の存在割合の増加に対して CSMA よりもスループットが低下しにくい特性を有すること、並びに、文献 [12] と同様に、隠れ端末が存在するネットワークにおいても、比較的大きなデータパケットを送信する場合に MACA は効果を發揮することを確認した。

また、本論文における計算機シミュレーション結果から、隠れ端末が発生しにくい通信環境では CSMA でも問題ないが、隠れ端末の発生が予測される場合に

は、システムスループットの急激な変化を避けるために MACA を適用することが望ましいことが分かった。

今後は、CSMA や MACA の拡張版として実用化され、広く使用されている IEEE802.11 系無線 LAN のように RTS 認識時間が伝搬遅延時間に比べて十分に大きく設定されるものなどの様々な MAC プロトコルの性能評価を行っていくつもりである。

文 献

- [1] L. Kleinrock and F.A. Tobagi, "Packet switching in radio channels: Part I—Carrier sense multiple access modes and their throughput delay characteristics," IEEE Trans. Commun., vol.COM-23, no.12, pp.1400–1416, 1975.
- [2] Editors of IEEE802.11, "Wireless LAN medium access control (MAC and physical layer (PHY) specifications, Draft Standard)," IEEE802.11, 1997.
- [3] F.A. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part II—The hidden terminal problem in CSMA and busy-tone solution," IEEE Trans. Commun., vol.COM-23, no.12, pp.1417–1433, 1975.
- [4] P. Karn, "MACA—A new channel access method for packet radio," ARRL/CRRL Amateur Radio 9th, Computer Networking Conference, pp.134–140, 1990.
- [5] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LAN's," Proc. ACM SIGCOMM '94, pp.212–225, 1994.
- [6] F. Talucci, M. Gerla, and L. Fratta, "MACA-BI (MACA By Invitation): A receiver oriented access protocol for wireless multiple networks," PIMRC '97, pp.435–439, Helsinki, Finland, 1997.
- [7] C.L. Fullmer and J.J. Garcia-Luna-Aceves, "Solutions to hidden terminal problems in wireless networks," Proc. ACM SIGCOMM '97, pp.39–49, 1997.
- [8] Y. Ko, V. Shankarkumar, and N.H. Vaidya, "Medium access control protocols using directional antennas in ad hoc networks," Proc. IEEE INFOCOM, pp.13–21, March 2000.
- [9] S. Ray, J. Carruthers, and D. Starobinski, "RTS/CTS-induced congestion in ad-hoc wireless LANs," Proc. IEEE WCNC 2003, pp.1516–1521, 2003.
- [10] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan, "Achieving MAC layer fairness in wireless packet networks," Proc. ACM/IEEE MobiCom, pp.87–98, 2000.
- [11] 重安哲也, 松野浩嗣, 森永規彦, "競合方式 MAC プロトコルにおける送信端末から 2 ホップ先の端末の影響," 信学論 (B), vol.J87-B, no.1, pp.93–97, Jan. 2004.
- [12] 松野浩嗣, 石中秀幸, 重安哲也, "MACA における伝搬遅延時間と RTS 認識時間の影響," 信学論 (B), vol.J86-B, no.12, pp.2449–2459, Dec. 2003.

(平成 17 年 3 月 14 日受付, 9 月 12 日再受付)



重安 哲也 (正員)

平 12 山口大・理・自然情報卒。平 14 同大大学院博士前期課程了。同年、広島国際大学社会環境科学部情報通信学科助手。無線環境下、特にある特定の基地局に依存しないネットワーク環境下における媒体通信制御プロトコルの改良並びに、特性評価に関する研究に従事。IEEE、情報処理学会各会員。



松野 浩嗣 (正員)

昭 57 山口大・工・電子卒。昭 59 同大大学院修士課程了。昭 59~62 山口短期大、昭 62~平 6 大島商船高専勤務。平 7 山口大・理・助教授。平 17 同教授。計算機ネットワーク構築技術と生命のシステムの理解に関する研究に従事。理博。IEEE、情報処理学会各会員。



森永 規彦 (名誉員：フェロー)

昭 43 阪大大学院博士課程了。同大工学部助手、講師、助教授を経て、昭 62 同大工学部教授。平 15 大阪大学名誉教授、広島国際大学情報通信学科教授、平 17 同大社会環境科学部長、現在に至る。工博。無線通信、衛星通信、移動通信、光通信などの通信方式に関する研究、並びに通信システムに対する干渉雑音対策技術に関する研究に従事。電子情報通信学会元副会長、元関西支部長、宇宙航空研究開発機構客員開発部員、自治体衛星通信機構評議員、総務省電気通信事業紛争処理委員会委員(委員長代理)、平 7 年度本会論文賞、平 10 年度同業績賞、平 16 年度同功績賞、平 17 同名誉員、平 10 年度情報化月間通産大臣表彰。IEEE Fellow。