

メッシュネットワークにおける帯域割当に関する研究

2007MI158 中村充矩

2007MI185 奥田 亮太

2008MI197 濟田 真典

指導教員 石崎文雄

1 はじめに

1.1 研究背景

近年、無線通信による通信技術やネットワーク構築技術の高度化が進んでいる。一般に無線通信で形成されるネットワークでは、基地局やアクセスポイントが中心となり、個々の端末がそれらの間で通信を行うスター型ネットワークの形態が多い。スター型ネットワークの場合、基地局に電波の届く範囲しか通信する事ができず通信可能な領域が固定的となってしまう。範囲外の端末と通信を行う場合は、新しく基地局やアクセスポイントを作らなければならない、コストがかかってしまう。また災害等により基地局やアクセスポイントにトラブルが発生した場合には、エリア全体でネットワークが停止する危険性がある。これらのスター型ネットワークの通信範囲の問題やトラブルを解決するネットワークとしてマルチホップワイヤレスメッシュネットワーク（以下、メッシュネットワーク）が現在注目されている。有線ネットワークが1960年代後半から1970年代前半に経験したパラダイムシフトにより、巨大な分散ネットワークであるインターネットが生み出された。同じように現在無線ネットワークで進行中のパラダイムシフトにおいて、メッシュネットワークは将来のワイヤレスネットワークの方向性を示していると考えられている [1]。

1.2 メッシュネットワークとは

メッシュネットワークとは、無線通信機能を持った端末同士が相互にデータを送受信することにより形成されるネットワークである。メッシュネットワークはメッシュルータとメッシュクライアントから構成される。各ノードはルータのような役割を持ち、各ノードのデータは隣のノードからノードへとパケットリレー式に伝送され、隣のノードへ段階的にホップを繰り返すことで目的のノードへと運ばれる。そのため離れている端末同士であってもメッシュネットワークに参加している場合において、全ての端末からネットワークに接続することができ、通信範囲を大幅に拡大することができる。

メッシュネットワークはアドホック型ネットワークのため基地局やアクセスポイントを必要としない。またメッシュネットワークにはノード間の接続性を維持する自己修復機能があるため端末が破損したり離脱したりしても継続的に接続、再構成を繰り返すことで、送信先に達するまでノードからノードへとデータの転送を行うことができ、代替りの通信路を複数確保することができる。そのため非常にトラブルや障害にも強く、維持コストも低く抑えることができる。さらにメッシュネットワークにはノードが自動的にネットワークを構築する自己組織化機能があるため、設置コストを低く抑えることがで

きる。このようにメッシュネットワークは拡張性とコスト面においてメリットが大きい。有線の通信インフラの設営が時間的・コスト的に困難な場所や状況でも低コストで導入、運営が可能のため、過疎の山間地での通信インフラ整備、途上国等の通信インフラの未熟な地域でのネットワーク整備などでの利用も期待されている。

しかしメッシュネットワークにも問題点がある。アクセスポイントや基地局が存在しない、中心を持たないアドホック型の構造であるため、通信経路の探索や伝送制御を行う高度なルーティング技術が必要とされる。加えて、規模が大きくなりホップ数が増えるほどデータの送受信やルーティングが複雑になり、通信速度の低下や通信障害等の通信品質の低下が懸念される。

メッシュネットワークは従来のアドホックネットワークと似ているが、アドホックネットワーク用に設計されたプロトコルやアーキテクチャをメッシュネットワークに適用しても、非常に貧弱な性能になってしまう [2]。さらにメッシュネットワークとアドホックネットワークでは、その最適設計基準も違っている。メッシュネットワークとアドホックネットワークの設計上の差異は、これらのネットワークのアプリケーションと設置目的の違い、および資源制約の違いから来ている。例えばアドホックネットワークは非常に移動性の高いマルチホップ環境を想定している。一方、メッシュネットワークは固定されたあるいは移動が制限された環境を想定している。そのためアドホックネットワーク用に設計されたプロトコルをメッシュネットワークに適用すると非常に貧弱な性能しか得られないことになる。さらにメッシュネットワークにおける資源はアドホックネットワークにおける資源よりも豊富である。また特定のネットワークトポロジーが想定されるメッシュネットワークのアプリケーションも多い。このような場合、その特定のネットワークトポロジーを最大限利用したプロトコル設計をすることができる。したがって従来のアドホックネットワーク用に設計されたプロトコルや制御方式ではなく、メッシュネットワーク用に設計された新たなプロトコルや制御方式を開発する必要がある。なお、奥田亮太は主に研究背景を、濟田真典は主にモデルから定式化までを、中村充矩は主にスケジューリング以降を担当した。

2 研究目的, 研究方法

本研究では、メッシュネットワークにおける適切な帯域割当に関する問題に着目する。この問題のために、メッシュネットワーク内のフローをいくつかの優先権クラスに分類し、クラスごとに適切に帯域を割当ててパケットスケジューリングを行う手法を考える。本研究では、その問題を数理計画法として定式化し、整数計画問

題を解くことによりクラスごとの適切な帯域割当を決定する。様々な環境下での数値結果を得て、それらを精査することにより提案手法と定式化の有効性について論じる。

本研究は、時間分割モード TDMA(Time Division Multiple Access) 方式を採用した IEEE 802.16 メッシュネットワークについて考察する。TDMA は伝送に用いる搬送周波数をタイムスロット単位で分割して共有する多重化方式のことである。本研究では、特にメッシュネットワークのトポロジとして応用上重要であると思われるツリー型ネットワークトポロジを考える。このネットワークにおいて、MAC 層が単独のキャリアチャネルを使用し、TDMA 方式でスケジューリングを行うと想定する。各ノードに課せられた帯域割当制約条件等の制約条件を満たしながら、フレーム長を最小にするスケジューリングを得ることを考える。このことにより、資源を最大限利用し、良好な QoS(Quality-of-Service) を提供できることになる。本研究では TDMA スケジューリングのために、先行研究 [3] で行われている定式化を見直し、新たな定式化を行う。TDMA スケジューリングは整数計画問題として定式化され、その整数計画問題を解くことによってフレーム長を最小にするスケジューリングを得ることができる。

3 モデル

本研究で使用するメッシュネットワークのモデルについて説明する。ネットワークは 1 つの BS(Base Station) と N 個の SS(Subscriber Stations) から構成されるものとする。BS 以外のすべてのノードは BS と通信するのが目的とする。BS との間に無線リンクを持つノードは BS と直接通信し、BS との間に無線リンクを持たないノードは、そのノードと BS との間にあるノードによって中継が行われることにより BS と通信する。したがってネットワークトポロジは図 1 のような BS を根としたツリー型ネットワークであると仮定する。ノード 0 を BS とし、それ以外のノード 1 からノード N をすべて SS とする。ノード間の通信は、すべて双方向の無線リンクを持つ全二重通信である。

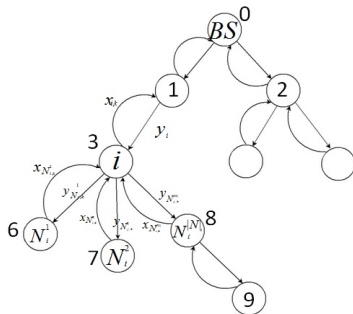


図 1 ツリー型モデル

3.1 変数の定義

ここでは、例として図 1 のような 1 個の BS ノードと 9 個の SS ノードから構成されるモデルを元に変数の定義を行う。

ノード 0 から任意のノードに向かう方向を下流、任意のノードからノード 0 に向かう方向を上流とする。ネットワークにおける SS ノードの個数を N で、TDMA のフレーム長を K で表す。ノード 0 からノード i までのホップ数を $h(i)$ で表す。ノード i と隣接し、かつノード i からみて下流にあるノードの集合を $N(i)$ とする。図 1 の例では、 $i = 3$ で $N(i)$ はノード 6, 7, 8 からなる集合である。さらに、ノード i と隣接し、かつノード i からみて下流にある各ノードに適当に一意的な番号を付し、その l 番目のノードを N_i^l で表す。ノード i が BS から要求される情報量を d_i で表す。つまり d_i は、ノード i で発生する情報量である。 d_{0i} は、 d_i とは逆に、BS がノード i から要求される情報量を表す。ノード i の下流にあるすべてのノードの集合を $C(i)$ で表す。 $N(i)$ がノード i に隣接したノードのみを指していたの 비해、 $C(i)$ はノード i の下流すべてのノードから構成されるノードの集合である。図 1 の例では、 $i = 3$ で $C(i)$ はノード 6, 7, 8, 9 からなる集合である。ノードのキャパシティを P で表す。ノードのキャパシティが大きいほど複数のアップリンクトラフィックやダウンリンクトラフィックを同時に送受信することができ、小さいほど同時に送受信することができなくなる。ノード i がフレームの k 番目のスロットにおいて、自分の上流にある親ノードに送るアップリンクトラフィック量を $x_{i,k}$ で表す。 $x_{i,k}$ は、ノード i がフレームの k 番目のスロットにおいてデータを送っているときは $x_{i,k} = 1$ 、送っていないときは $x_{i,k} = 0$ の値をとる決定変数である。図 1 の例では、 $x_{i,k}$ はノード 3 から親であるノード 1 への上向きの矢印にあたる。ノード i がフレームの k 番目のスロットにおいて、自分の上流にある親ノードから受け取るダウンリンクトラフィック量を $y_{i,k}$ で表す。 $y_{i,k}$ は、 $x_{i,k}$ と同様に 0 か 1 の値をとる決定変数である。図 1 の例では、 $y_{i,k}$ はノード 1 からノード 3 への下向きの矢印にあたる。本研究ではフレーム長 K を最小化するスケジュールを求めることが課題である。

4 定式化

本節では TDMA スケジューリングのための定式化を行う。TDMA スケジューリングは整数計画問題として定式化され、その整数計画問題を解くことによってフレーム長を最小にするようなスケジュールを得ることができる。

4.1 整数計画問題への定式化

本節では、前節で定義した変数を使用して、本研究で考える TDMA スケジューリング問題の数理計画問題への定式化を行う。TDMA スケジューリング問題は、変数 K を十分大きな正の整数として、次の (1) から (9) までの数理計画問題として定式化する。

$$\min \sum_{i=1}^N \sum_{k=1}^K C_k(x_{i,k} + y_{i,k}) \quad (1)$$

s. t.

$$\sum_{k=1}^{K-h(i)} x_{i,k} = d_i + \sum_{j \in C(i)} d_j, \quad i = 1, \dots, N \quad (2)$$

$$\sum_{k=1}^K y_{i,k} = d_{0i} + \sum_{j \in C(i)} d_{0j}, \quad i = 1, \dots, N \quad (3)$$

$$x_{i,k} + \sum_{j \in N(i)} y_{j,k} + y_{i,k} + \sum_{l \in N(i)} x_{l,k} \leq P, \quad i = 1, \dots, N : k = 1, \dots, K \quad (4)$$

$$\sum_{t=1}^k x_{i,t} - \sum_{j=1}^{k-1} \sum_{l=1}^{|N(i)|} x_{N^l_i, j} \leq d_i, \quad i = 1, \dots, N : k = 1, \dots, K \quad (5)$$

$$\sum_{t=1}^k \sum_{j \in N(i)} y_{j,t} \leq \sum_{t=1}^{k-1} y_{i,t}, \quad i = 1, \dots, N : k = 1, \dots, K \quad (6)$$

$$y_{i,k-1} = 0, \quad k \leq h(i), \quad i = 1, \dots, N : k = 1, \dots, K \quad (7)$$

$$x_{i,k} = 0, \quad k > K + 1 - h(i), \quad i = 1, \dots, N : k = 1, \dots, K \quad (8)$$

$$x_{0,k} = 0, \quad k = 1, \dots, K \quad (9)$$

ここに、 $C_k = (2L)^{(k-1)}k!$ で、 L はネットワークにおける無線リンクの総数を表す。

この整数計画問題を解き、その解 $x_{i,t}, y_{i,t}$ ($i = 1, \dots, N; t = 1, \dots, K$) から

$$K^* = \max\{m \mid x_{i,t} > 0, \text{ or } y_{i,t} > 0, i = 1, \dots, N\}$$

を得れば、 K^* が最小フレーム長になっている。このことの証明は 4.2 節で行う。またその時のスケジュールは $x_{i,t}, y_{i,t}$ ($i = 1, \dots, N; t = 1, \dots, K^*$) で与えられる。

次に、各制約条件の意味を説明する。制約条件 (2) は、BS からノード i とノード i の子に対して要求している情報量の合計はノード i からノード i の親へのアップリンクトラフィック量と等しくなければならないという制約を表している。制約条件 (3) は、ノード i とノード i の子から BS への要求している情報量の合計はノード i の親からノード i へのダウンリンクトラフィック量と等しくなければならないという制約を表している。制約条件 (4) は、ノード i において親や子へのアップリンクトラフィック量とダウンリンクトラフィック量の合計はノード i のキャパシティ P 以下にならなければならないという制約を表している。制約条件 (5) は、ノード i で発生するアップリンクトラフィック量は BS からノード i に対して要求する情報量以下でなければならないという制約を表している。制約条件 (6) は、ノード i から下流ノードへのダウンリンクトラフィック量の合計は、ノード i の親からノード i へのダウンリンクトラフィック量以下でなければならないという制約を表している。制約条件 (7) は、スロット k よりも、ノード i から BS へのホップ数の方が大きい場合、ノード i の親からノード i へのダウンリンクトラフィック量は 0 にならなければならないという制約を表している。制約条件 (8) は、スロット k がフレーム長 $K + 1 -$ ホップ数以上ならば、ノード i からノード i の親へのアップリンクトラフィック量は 0 でなければならないという制約を表している。制約条件 (9) は、ノード 0 すなわち BS は根であるためそのアップリンクトラフィックは常に 0 であるという制約を表している。

4.2 目的関数の最小化と最小フレーム長

ここでは、4.1 節で示した整数計画問題を解き、その解 $x_{i,t}, y_{i,t}$ ($i = 1, \dots, N; t = 1, \dots, K$) から

$$K^* = \max\{m \mid x_{i,t} > 0, \text{ or } y_{i,t} > 0, i = 1, \dots, N\}$$

を得ることで、 K^* が最小フレーム長になっていることを証明する。

目的関数を最小化する解 $x_{i,t}, y_{i,t}$ ($i = 1, \dots, N; t = 1, \dots, K$) を目的関数に代入して、

$$\begin{aligned} \sum_{i=1}^N \sum_{k=1}^K C_k(x_{i,k} + y_{i,k}) &= \sum_{i=1}^N \sum_{k=1}^{K^*} C_k(x_{i,k} + y_{i,k}) \\ &\leq 2L \sum_{k=1}^{K^*} C_k \\ &= 2L \sum_{k=1}^{K^*} (2L)^{k-1} k! \\ &\leq K^* (2L)^{K^*} K^*! \end{aligned} \quad (10)$$

を得る。ここで、 $m \geq 1$ に関して

$$\begin{aligned} C_{K^*+m} &= (2L)^{K^*+m-1} (K^* + m)! \\ &\geq C_{K^*+1} = (2L)^{K^*} (K^* + 1)! \\ &> K^* (2L)^{K^*} K^*! \end{aligned}$$

$$\geq \sum_{i=1}^N \sum_{k=1}^K C_k(x_{i,k} + y_{i,k}) \quad (11)$$

であるから, $x_{K^*+m} > 0$, あるいは, $y_{K^*+m} > 0$ となる m が存在すれば, それは目的関数を最小化しないことがわかる. このことから, 目的関数を最小化する解によって定まる K^* が最小フレーム長を与えることがわかる.

5 実行結果

表 1 $P = 3, doi = di = 1$ (縦軸: ノード, 横軸: フレーム長)

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|------|---|---|---|---|---|---|---|---|---|
| x[0] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x[1] | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| x[2] | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| x[3] | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| x[4] | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x[5] | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x[6] | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x[7] | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x[8] | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x[9] | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| y[0] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| y[1] | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| y[2] | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| y[3] | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| y[4] | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| y[5] | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| y[6] | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| y[7] | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| y[8] | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| y[9] | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

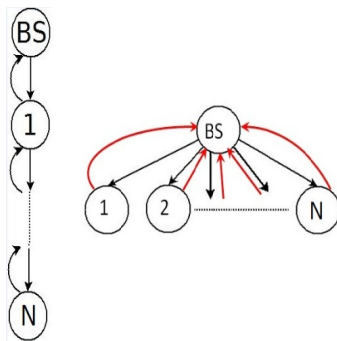


図 2 ホップ数の異なるモデル

前項で提示した整数計画問題をキャパシティを 3, BS から要求する情報量と, BS に要求する情報量を 1 と設定して解いた. この整数計画問題を解くにあたって Gurobi Optimizer(高性能線形・整数計画ソルバー)を

用いた. その結果, 最小フレーム長の値は 8 となった. また, 同じように, 図 2 の左のような一本道のようなノードで数値を変えずに整数計画問題を解いた. その結果, 最小フレーム長の値は 9 となった. この結果から, 条件は同じでもノードの配置が違えば最小フレーム長の値が変わってくる. これはホップ数が増えることで中継するノードが増えるためフレーム長も増えるといえる. また, 図 2 の右のような BS に 1 ホップで到達するノード配置図でもキャパシティが 3 の場合は最小フレーム長は 6 となるがキャパシティが 1 の場合は最小フレーム長は 18 となった. このことから BS に近くても適切なキャパシティを設定しないと効率が悪くなるということがいえる.

6 まとめ

IEEE 802.16 のメッシュモードにおける TDMA スケジューリング問題を整数計画問題の形で定式化を行った. そして, 提示した整数計画問題を解くことによってフレーム長を最小にするようなスケジューリングを得ることができた.

提案した定式化から, 最小フレーム長の値は, メッシュネットワークの構成とノードの個数, ノードのキャパシティ等によって変化する. モデルの形状に関しては, 例えば図 2 の左側のようなネットワークのようにノードが一直線に並んでいると極端に最小フレーム長の値が大きくなり, 情報伝達効率が悪い結果がでた. 図 2 の右側のようなネットワーク構成でも, ノードのキャパシティが一定以上ないと情報伝達効率が悪い. キャパシティを大きく設定すれば最小フレーム長の値は最小化されるが, キャパシティがある一定値以上は最小フレーム長の値は変化しないためコストがかかるだけである.

したがって, BS からのホップ数を出来る限り小さくし, 適切なキャパシティを設定することが情報伝達効率の点からは大切だといえる.

今回はノードの配置, キャパシティなどは簡単で単純なものしか解いてないため, 今後の課題として, 様々な環境で提示した整数計画問題を解き, その数値結果からネットワークの特性を調べる. 具体的には, ノード配置を複雑にし, 各ノードで発生する情報量を増減させ, それが最小フレーム長にどのように影響するか検証する. また, 現在のモデルでは各アップリンクトラフィックや各ダウンリンクトラフィックに対して優先度を与えていないので, 優先度を与えたモデルについても考察する.

参考文献

- [1] Y. Zhang, J. Luo and H. Hu, Wireless Mesh Networking: Architectures, Protocols and Standards, Auerbach Publications, 2006.
- [2] I. F. Akyildiz and X. Wang, Wireless Mesh Networks, Wiley, 2009.
- [3] G. Aggelou, Wireless Mesh Networking, McGraw-Hill Professional, 2008.