

## 受信端末駆動型無線マルチホップネットワークにおける 制御パケットの衝突回避による性能向上

速水 直<sup>†</sup> 小南 大智<sup>††</sup> 菅野 正嗣<sup>†††</sup> 村田 正幸<sup>††</sup> 畠内 孝明<sup>††††</sup>

<sup>†</sup> 大阪大学基礎工学部情報科学科 〒 565-8331 豊中市待兼山町 1-3

<sup>††</sup> 大阪大学大学院情報科学研究科 〒 565-0871 吹田市山田丘 1-5

<sup>†††</sup> 大阪府立大学総合リハビリテーション学部 〒 583-8355 羽曳野市はびきの 3-7-30

<sup>††††</sup> 富士電機システムズ 〒 191-8302 日野市富士町 1

E-mail: †{t-hayamizu,d-kominami,murata}@ist.osaka-u.ac.jp, ††sugano@rehab.osakafu-u.ac.jp,

†††hatauchi-takaaki@fesys.co.jp

あらまし 受信端末駆動型通信方式は、無線マルチホップネットワークにおいて、データ受信側のノードが間欠的に ID パケットを送信し、送信側のノードが受信、応答することで通信を行う。しかしながら、隠れ端末同士の間にあるノードで制御パケットの衝突が発生し、さらにこの衝突が繰り返されることが性能を劣化させる要因になっている。本稿では、このような連続衝突を回避し、性能を向上させる手法を提案する。シミュレーションにより、データ発生頻度の低い環境において 20%、高い環境においては 100% 向上できることを示す。

キーワード センサネットワーク, 隠れ端末問題, 連続衝突, 衝突回避, 性能向上

## Performance Improvement by Collision Avoidance of Control Packet for Receiver-driven Multi-hop Wireless Networks

Tadashi HAYAMIZU<sup>†</sup>, Daichi KOMINAMI<sup>††</sup>, Masashi SUGANO<sup>†††</sup>, Masayuki MURATA<sup>††</sup>, and  
Takaaki HATAUCHI<sup>††††</sup>

<sup>†</sup> Department of School of Information and Computer Sciences, School of Engineering, Osaka University  
1-3 Machikaneyama, Toyonaka, Osaka, 565-8331, Japan

<sup>††</sup> Graduate School of Information Science and Technology, Osaka University  
1-5 Yamadaoka, Suita, Osaka, 565-0871, Japan

<sup>†††</sup> School of Comprehensive Rehabilitation, Osaka Prefecture University  
3-7-30 Habikino, Habikino, Osaka, 583-8355, Japan

<sup>††††</sup> Fuji Electric Systems  
1, Fujicho, Hino, Tokyo, 191-8302, Japan

E-mail: †{t-hayamizu,d-kominami,murata}@ist.osaka-u.ac.jp, ††sugano@rehab.osakafu-u.ac.jp,

†††hatauchi-takaaki@fesys.co.jp

**Abstract** This research focuses on an intermittent receiver-driven data transmission (IRDT) protocol in which communication starts when multiple receiver nodes transmit their own IDs intermittently and a sender node receives them. However, the performance has deteriorated because a collision of control packets occurs between hidden terminals and this collision occurs continuously. In this paper, we propose some methods for improving performance by avoiding such a control packets' continuous collision. By applying the proposal methods, we show that 100% of improvement in performance in the environment where data generating frequency is high, and 20% of improvement in performance where frequency is low.

**Key words** Sensor Network, Hidden Node Problem, Continuous Collision, Collision Avoidance, Performance Improvement

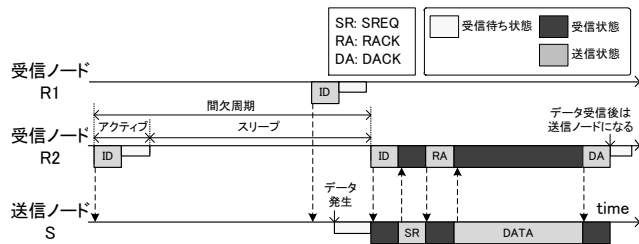


図 1 IRDT 方式の動作シーケンス

## 1. はじめに

端末がバッテリーによって動作するセンサネットワークにおいては、省電力化が重要な課題であり、間欠動作による低消費電力化が有効であるとされている [1]。間欠動作による通信を行う際には、タイミングをノード間で制御する必要があり、これは同期式と非同期式に分類できる。後者はさらに通信開始を通知するパケットを送信するノードがデータの送受信いずれを行うかによって、送信側駆動型 [2] と受信側駆動型 [3, 4] に分類される。非同期式は任意のタイミングで起動して通信を行うため同期の維持は必要ないが、衝突を前提とした制御が必要となる。また、非同期式では通信相手が通信可能となるまでの待ち時間が発生し、待機中のノードは起動状態を維持するため、この期間が消費電力の大きな部分を占めることになる。

電力メータのモニタリングのように、トポロジーが密でパケット発生頻度が小さいシステムにおいては、非同期の受信端末駆動型である Intermittent Receiver-driven Data Transmission (IRDT) 方式が適している [5]。IRDT 方式では、それぞれのノードが、隣接する全てのノードに関しての ID 番号やシンクノードまでのホップ数といった隣接ノード情報を保持しており、制御パケットを利用してこれらの情報を交換することで、最新の隣接ノード情報を保持することができる。このような隣接ノード情報を用いて、例えば相対的な残余電力に基づいた制御によるネットワーク全体の長寿命化などが行われている [6]。

IRDT 方式では、ノードが自身の ID 番号 (ID) を送信した後、送信データを保持するノードがその ID に送信要求パケット (SREQ) を送信することで通信を開始する。送信データを保持する複数のノードが、互いに隠れ端末の関係にある場合には、キャリアセンスが有効に働かず、同時に制御パケットを送信してしまうため衝突が発生する。衝突により制御パケットを正常に受信できなかったノードはスリープ状態になるが、次の ID 送信時に前回と同様、ID に対して複数のノードが SREQ を送信すると、再び制御パケットが衝突する。これがさらに連続して起こることにより、パケットの収集率や消費電力などが劣化するという問題がある。そこで本稿では、このような制御パケットの連続衝突による性能劣化を解決する方法として、バックオフやその他の制御パケットの連続衝突を回避する手法を提案する。特に、IRDT の特徴である隣接ノード情報を適切に用いることによって、パケットの収集率の向上や遅延時間の向上を行い、IRDT 方式の性能の改善を目指す。

本稿は、以下のような構成となっている。初めに 2 章では、IRDT 方式の基本動作と、回避すべき問題である SREQ の連続

衝突問題の概要について説明する。3 章では、連続衝突問題を回避する手法を提案し、その評価を行う。4 章では、隣接ノード情報を活用することで、さらに性能を改善するための手法を提案する。最後に、5 章に本報告のまとめを述べる。

## 2. 受信端末駆動型通信方式 IRDT と連続衝突問題の概要

### 2.1 MAC 層: IRDT の基本的な動作

IRDT 方式の基本的な動作の様子を図 1 に示す。データ受信側であるノード R1 と R2 は、間欠的に自身の ID を送信することで、データ受信が可能な状態であることを周囲に通知する。これに対して、データ送信側のノード S は ID の受信を待機し、受信した ID が通信先として適切なノードからのものであれば送信要求制御パケットである SREQ を送信する。さらに SREQ を受信した受信側のノードが RACK パケット (RACK) を送信することで、リンクが確立されデータを送受信する。

### 2.2 ネットワーク層: マルチホップ転送

本方式はマルチホップ転送を前提としており、各ノードはパケットの中継処理を行う。消費電力や遅延の増加を防ぐためには、宛先ノードまで最小ホップ数となるような経路でパケットを転送することが望ましいが、無線電波の状態や輻輳、端末の故障等の理由から不可能な場合が存在する。そこで本方式では、最小ホップ経路以外の迂回経路も柔軟に選択するようなルーティングプロトコルを用いることで、信頼性の向上を図っている。各ノードは構成情報管理テーブルを持っており、ネットワークの経路情報を定期的に交換することでテーブルを更新し、最新の経路情報を常に保持している。各ノードは、このテーブルからパケットの宛先となるシンクノードまでの中継ホップ数を求めることができる。あるノード S から宛先ノード D までの最小ホップ数が  $h$  である時、ノード S に隣接するノードからノード D までの最小ホップ数は  $h-1, h, h+1$  のいずれかとなり、以降はそれぞれのホップ数のノードを、前向きノード、横向きノード、後向きノードと呼ぶこととする。パケットの転送先としては、中継ホップ数が少なくなる前向きノードが優先される。データ送信時に前向きノードからの ID を受信した場合にはそのまま SREQ を返信して通信を行ない、全ての前向きノードとの通信が失敗した後に、横向きノードへの転送を行う。さらに、全ての横向きノードとの通信が失敗した場合には、後向きへの通信を行う。ここで通信の失敗とは、RACK, DACK の受信が確認できない場合としている。また、データの中継回数にはあらかじめ上限を与えており、この最大中継回数を超えない限りは通信を行う。ただし、横向き/後向きノードへの転送を行うと最大中継回数を超えてしまう場合には、転送を行なわない。

### 2.3 制御パケットの連続衝突問題

SREQ の連続衝突の様子を図 2 に示す。各ノードはパケット送信に先立ってキャリアセンスを行い、送信を制御することで衝突を回避する。しかしながら、隠れ端末の関係にあるノード同士では、キャリアセンスが有効ではなく、衝突が発生してしまう。図 2 では、受信側のノード R が ID を送信した後、同時

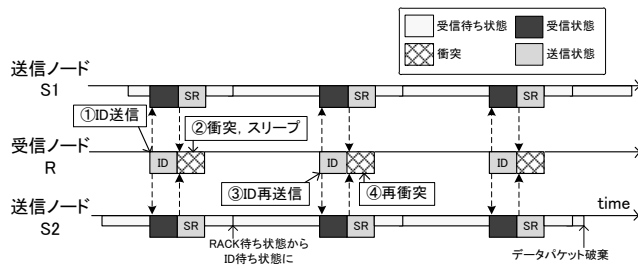


図 2 SREQ パケットの連続衝突

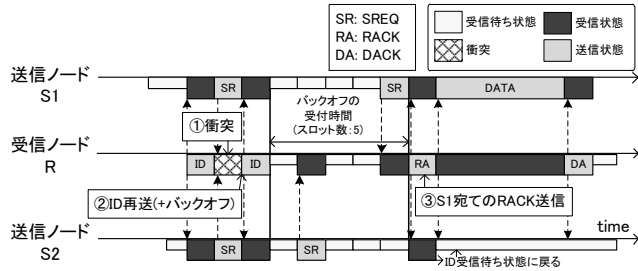


図 3 バックオフ機能

に ID を受信したデータ送信側の隠れ端末の関係にあるノード S1, S2 がともに SREQ を送信するため、衝突が発生する。衝突によりどちらの SREQ も正常に受信できなかった受信側のノード R はスリープ状態になるが、S1, S2 がデータを保持したままであれば、ID の送信によって再び SREQ の衝突が発生してしまうことになる。最悪の場合には、SREQ の連続衝突を繰り返すうちに、データに用意されたパケット破棄時間が経過して、パケットが破棄されてしまう。この問題が、現在の IRDT における収集率の低下や消費電力の増大などの原因となっている。そこで本稿では、このような SREQ の連続衝突を回避するための方式を提案する。

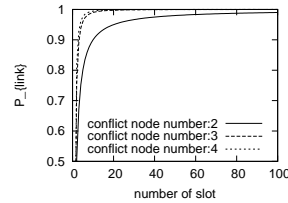
### 3. 連続衝突問題を回避する制御

#### 3.1 SREQ の衝突発生時に行う制御

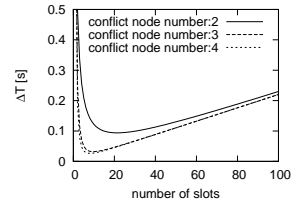
受信側のノードが SREQ の受信待ち状態の間に正常でないパケットを受信したとき、複数の送信ノードからの SREQ が衝突したと判断する。そこで、そのような場合に、SREQ の連続衝突を回避する方法として、バックオフ、確率的な再送要求、ポーリングの 3 通りの手法を IRDT 方式に適用し、その評価を行う。これらの手法は組み合わせることで適用することが困難であるため、それぞれ別に評価を行ない、比較する。

##### 3.1.1 バックオフ機能

バックオフ機能の動作シーケンスを図 3 に示す。受信側のノードは、SREQ 衝突の検出に続いて、SREQ 送信に要する時間であるスロットタイムの整数倍の受付時間を設定し、その整数値を ID に付加して送信する。この ID を受信した ID 受信待ち状態のノードは、受信した ID が通信先として適当なノードのものであれば、スロットタイムの倍数に基づいてランダムにタイミングを選択し、SREQ を送信する。ID を送信したノードは、設定したスロットタイムの倍数の間待機して複数の SREQ を受信したのちに、受信成功した SREQ の中からランダムに 1 つを選択して、それを送信したノードに対する RACK を送信しリンクを確立する。自分宛ての RACK を受信することがで



(a) リンク確立成功率  $P_{link}$



(b) 遅延時間の変位  $\Delta T$

図 4 スロットタイムの倍数の検討

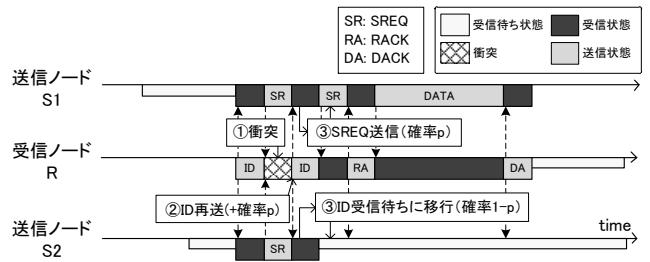


図 5 確率的に SREQ を再送させる機能

きなかったノードは、再度 ID の受信待ち状態に戻る。少なくとも 1 つの SREQ が受信できれば、リンクを確立することができるので、リンク確立の成功率  $P_{link}$  を  $1 - (\text{全ての SREQ が衝突によって受信できない確率})$  とする。 $P_{link}$  のスロットの倍数によるグラフは図 4(a) となる。また、スロットタイムの倍数が変化すると、リンク確立の成功率  $P_{link}$  とともに遅延時間も変化する。その変位を  $\Delta T$  とすると、 $\Delta T$  は、スロットタイムの倍数を  $S$ , 1 スロットタイムの時間を  $T_{slot}$ , 間欠周期を  $T_{cycle}$  として、(1) 式のように表せる。

$$\Delta T = T_{slot} \times S + T_{cycle} \times P_{link} \quad (1)$$

(1) 式に現在のシミュレーションの設定値を代入した結果が、図 4(b) である。図 4 から、SREQ の衝突は主に 2, 3 ノードと考えると、スロットタイムの倍数の初期値は、十分に SREQ の受信成功率が十分に高く、かつ  $\Delta T$  の小さい 40 とする。

##### 3.1.2 確率的に SREQ を再送させる機能

確率的に SREQ を再送させる制御機能の動作シーケンスを図 5 に示す。この機能では SREQ の衝突発生後、衝突を検出したノードが ID を再送する時に、確率  $p$  を付加して送信する。この ID を受信した ID 受信待ち状態のノードは、受信した ID が通信先として適切なノードからのものであれば、ID に含まれていた確率  $p$  で SREQ を送信する。また、再度 SREQ が衝突した場合や、どの送信側のノードからも SREQ が送信されなかった場合には、ID を再送したノードは、再度確率  $p$  を追加した ID を送信する。

##### 3.1.3 ポーリング機能

ポーリング機能の動作シーケンスを図 6 に示す。ID 再送時に、自身の隣接ノードのうちから SREQ の送信を許可するノードを選択し、それを示した情報を付加して送信する。この ID を受信した ID 受信待ち状態のノードは、自身が SREQ を送信する許可を受けており、受信した ID が通信先として適当なノードのものであれば、SREQ を送信する。ポーリングを行っている受信側のノードは、ID を送信した後、SREQ を受信し

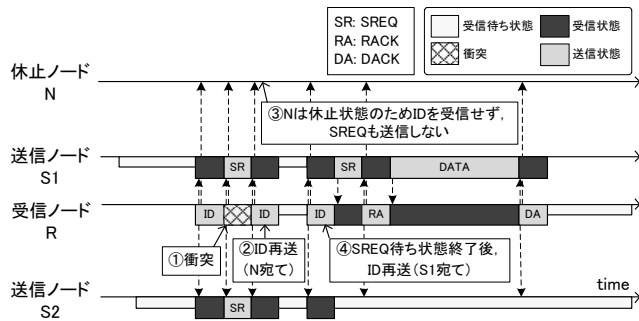


図 6 ポーリング機能

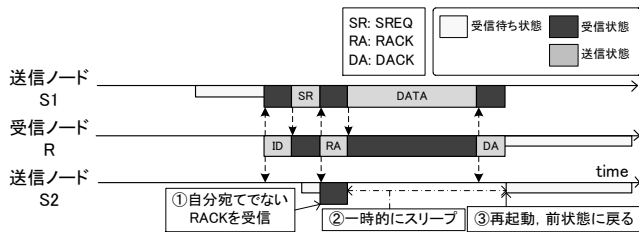


図 7 一時休止機能

なかった場合は、ポーリングを行ったノードが SREQ の送信をしなかったか、休止状態であると判断して、別の隣接ノードに対して SREQ の送信を許可する情報を追加して ID を再送する。最終的にどの隣接ノードからの SREQ も受信しなかった場合は、スリープ状態となる。ポーリングの対象ノードの順番は、自身の後向きノードの中からランダムに決定し、偏りがないようにしている。

### 3.2 衝突時以外に行う制御

次に、SREQ の連続衝突を回避する制御以外の、パケットの衝突を抑制する制御である一時休止機能について説明する(図 7)。これはリンクを確立していない状態で受信待ち状態 (ID や SREQ, RACK の受信待ち状態) のときに、自身宛てでない RACK を受信したときに行われる。図のノード S2 が RACK を受信したとき、自分の通信範囲内でデータの受信が行われると判断し、その通信時間中は衝突を避けるためパケットの送信を行わず、その通信が終了する時間まで一時的にスリープ状態になる。これにより、S2 が別の受信ノードからの ID が送信された場合に、SREQ を送信してしまうことを防ぎ、受信ノード R でのノード S1 からのデータとノード S2 からの SREQ による衝突の発生を抑制できる。この機能は、デフォルトの IRDT 方式や前節で説明した 3 手法と組み合わせることによって、その効果を評価する。

### 3.3 シミュレーションによる評価

図 8 のネットワークモデルにおいてシミュレーションによる評価を行った。シミュレーションの設定値は表 1 の通りである。何も機能を追加していない場合 (default)、バックオフ機能を追加した場合 (backoff)、確率的に SREQ を再送させる機能を追加した場合 (probability)、ポーリング機能を追加した場合 (polling)、さらにそれぞれに一時休止機能を追加した場合 (+pause) のシミュレーションを行い、そのデータ発生率と発生したデータの収集率、遅延時間、消費電力、衝突回数を図 9 に示す。

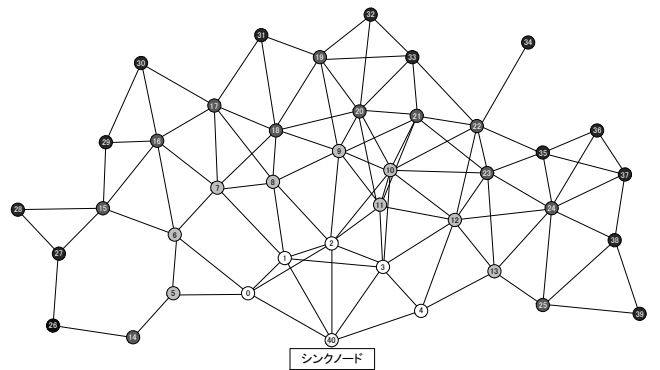


図 8 ネットワークモデル

表 1 シミュレーションの設定値

パラメータ名	値
間欠周期	1 秒
データ保持期限	間欠周期 * 5 秒
通信可能範囲	100 m
最大中継回数	最小ホップ + 5 回
送信時電流	20 mA
受信時電流	25 mA
スリープ時電流	0 mA
データサイズ	128 byte
無線速度	100 kbps

(a) 主な設定値

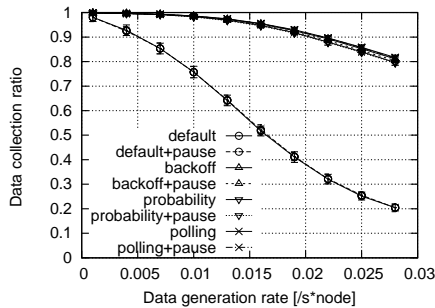
(b) データの発生モデル

バックオフ機能、確率的に SREQ を再送させる機能、ポーリング機能はそれぞれ、何も機能を追加していない場合と比べると、データの収集率、遅延時間、消費電力の性能が向上している。データ収集率では、データ発生頻度が 0.01 ~ 0.15/s・node、と低い場合では 0.95 を超え、データ発生頻度が 0.15 ~ 0.25/s・node、と高い場合でも、0.8 を超えており、それぞれ、何も機能を追加していない場合より 20%、100%性能が向上している。これにより、各機能は制御パケットの連続衝突問題を回避し、性能を向上できることが示された。

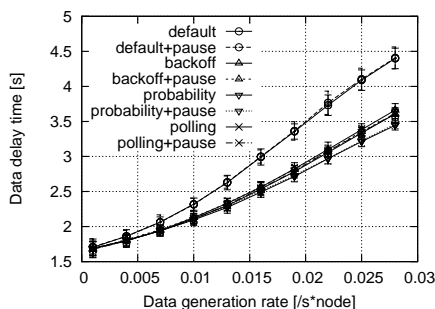
また、一時休止機能の有無は、データの収集率、遅延時間、消費電力にほとんど影響がないが、衝突回数ある程度抑制できている。これは、本稿において問題としている連続衝突とは別の単発的な衝突なので、データの収集率などには影響しないためと思われる。データの収集率などの性能を悪化させずに衝突回数を抑制できていることから、この機能には一定の効果があることがわかった。

## 4. 隣接ノード情報を利用した性能改善

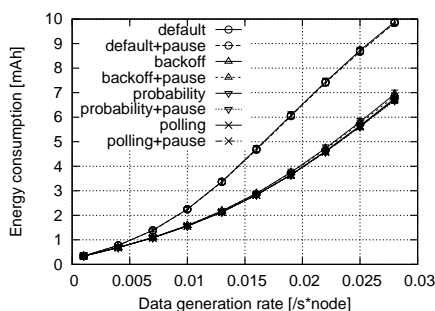
前章で示した各機能に対して IRDT 方式の特徴である隣接ノード情報を利用し、さらに性能を改善するための手法を検討する。隣接ノード情報とは、各ノードが所持している、全ての隣接するノードに関する情報である。隣接ノード情報の更新は、各ノードが ID や SREQ に情報を載せて送ることで行われている。既存の IRDT 方式では、隣接ノード情報として各ノードの ID 番号と残余電力情報を持っている。本稿では、この隣接ノード情報を拡張することで新たな情報を追加し、それを利用した性能改善手法を検討する。



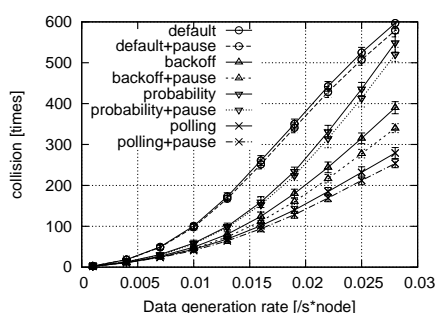
(a) 平均パケット収集率



(b) 平均パケット遅延時間



(c) 平均消費電力



(d) 平均衝突回数

図 9 各機能の性能評価

#### 4.1 隣接ノード情報の拡張

新たに追加した情報は、表 2 のとおりである。最新通信時刻は、隣接するノードとのデータを送受信した最新の時刻である。受信側のノードはデータの受信時、送信側のノードは DACK の受信時に更新する。この情報と現在の時刻を用いて、そのノードと最後に通信を行ってから経過した時間を計算できる。データ非保持平均時間は、個々のノードがデータを送信/破棄してから、データ発生/受信するまでの時間の平均値を保存している。この情報は ID や SREQ に追加して送信され、受信したノード

は情報を更新する。この情報によって、隣接ノードがデータを保持する頻度がわかり、時間が短いほど頻度が高い、すなわち負荷が高いことがわかる。隣接ノードのリストは、ノードが隣接しているノードの番号を格納している。このシミュレーションではネットワークモデルが変化しないので、シミュレーションの開始時のみ更新している。自身と隣接ノードのこの情報を組み合わせることで、隠れ端末の関係にあるノードの組がわかる。

#### 4.2 機能ごとの隣接ノード情報の利用方法

##### 4.2.1 バックオフ機能

バックオフ機能では、送信側のノードは SREQ を送信する際に、隣接ノード情報とは別に、いま保持しているデータが破棄されるまでの時間を追加して送信する。受信側のノードは SREQ の受付時間が経過した後、SREQ を送信してきたノードから、以下の優先度で RACK を送信し、リンクを確立するノードを選択する。

- (1) パケット破棄までの残り時間が最大の半分以下のノード
- (2) 最新通信時刻からの時間 > 非保持平均となるノード
- (3) 隠れ端末の関係にあるノードの数が多いいノード

パケット破棄までの残り時間が少ないノードを優先することで、パケット破棄を抑制し、データ収集率を向上させる。また、最新通信時刻からの時間 > 非保持平均時間となるノードを優先させることで、それぞれの負荷に応じた頻度で通信を行い、データ到着遅延時間やデータ収集率を改善できる。どのノードも以上の 2 つに該当しない場合は、隠れ端末の関係にあるノードの数が多いいノードを優先し、隠れ端末同士によるパケットの衝突を抑制する。

##### 4.2.2 確率的に SREQ を再送させる機能

この機能では、最新通信時刻やデータ非保持平均時間をもとに、最新通信時刻にデータを保持/破棄したと判断し、その時刻から現在までの時間とデータ非保持平均時間とを比較することで隣接ノードがデータを持っているかどうかを予測する。データを持っていると思われるノードの数  $n_{est}$  を基に、確率  $p$  を変更して ID を送信する。確率  $p$  は  $p = 1/n_{est}$  で決定している。衝突が発生していることから、 $n_{est}$  は 2 以上であることを前提として、 $n_{est} < 2$  のときは、 $n_{est} = 2$  とする。このように式を用いることで、確率  $p$  での SREQ の再送の回数を減少させることができ、データ到着遅延時間の改善ができる。

##### 4.2.3 ポーリング機能

ポーリング機能では、後向きの隣接ノードの中から、以下の優先度の順でポーリングを行っている。

- (1) 最新通信時刻からの時間 > 非保持平均時間となるノード
- (2) 隠れ端末の関係にあるノードの数が多いいノード
- (3) 最新通信時刻が古いノード

バックオフ機能と同様の理由で、最新通信時刻からの時間 > 非保持平均となるノードを優先させることで、それぞれの負荷に応じた頻度で通信を行い、パケットを効率的にシンクノードまで送信でき、データ到着遅延時間やデータ収集率を改善できる。また、隠れ端末の関係にあるノードの数が多いいノードを優先さ

表 2 拡張した隣接ノード情報

	拡張内容	更新タイミング	利用する情報
最新通信時刻	データ, DACK 受信時の時刻を保存する	通信完了時	そのノードと通信していない時間
データ非保持平均時間	個々がデータ非保持状態の期間の平均値をとる	ID, SREQ 受信時	各ノードにかかっている負荷
隣接ノードのリスト	自身に隣接するノードのリストを格納する	シミュレーション開始時	自身に対する隠れ端末の関係の組

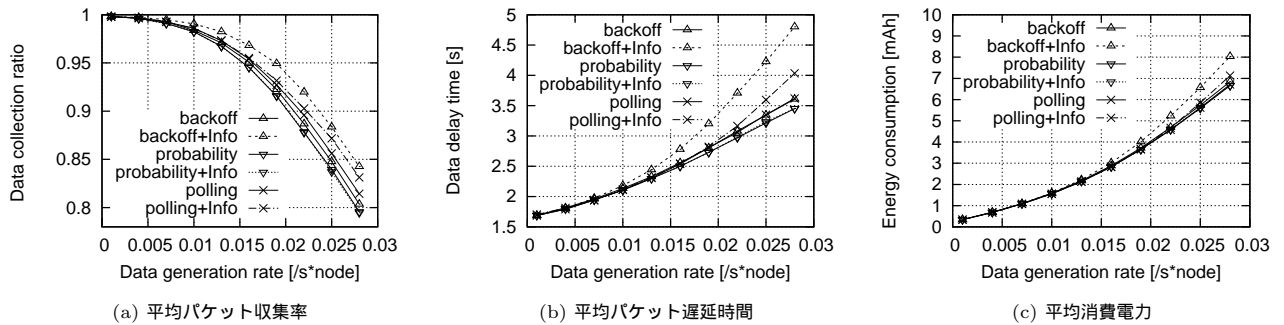


図 10 隣接ノード情報を利用した各機能の性能評価

せることで、隠れ端末同士によるパケットの衝突を抑制する。

### 4.3 シミュレーションによる評価

シミュレーションの設定は、3.3 節の設定と同じものを用いる。バックオフ機能、確率的に SREQ を再送させる機能、ポーリング機能それぞれを追加した場合と、何も機能を追加していない場合の比較は 3.3 節で行ったため、省略する。また、このシミュレーションでは 3.2 節で説明した一時休止機能は常に適用している。

バックオフ機能を追加した場合 (backoff)、確率的に SREQ を再送させる機能を追加した場合 (probability)、ポーリング機能を追加した場合 (polling)、さらに各機能に隣接ノード情報を用いた場合 (+info) のシミュレーションをそれぞれ行い、そのデータ発生率と発生したデータの収集率、遅延時間、消費電力を図 10 に示す。

バックオフ機能、ポーリング機能では隣接ノード情報を用いた場合、パケット収集率が向上した反面、遅延時間やわずかに消費電力が上がってしまっている。これはパケット収集率が向上し、ノードがデータを保持する時間が伸びた影響によるものと思われる。パケット収集率では、バックオフ機能、ポーリング機能ともに向上しているが、バックオフ機能ではその時点でのパケット破棄までの残り時間がわかり、それを前提とした優先度を決められるので、ポーリング機能よりもパケットの破棄を抑えることができている。また、確率的に SREQ を再送させる機能では、隣接ノード情報を用いても、性能が殆ど変わらない。これは、この機能が ID の再送信を繰り返すことで通信を行っているため、2~4 ノードの競合で、確率  $p$  が 0.5~0.25 と変動した程度では ID の再送信が数回変わる程度の変化しかなく、性能に影響しないためと思われる。

## 5. おわりに

本稿では、間欠動作に基づいた受信端末駆動型 IRDT 方式による無線マルチホップネットワークを対象として、送信要求パケットの連続衝突を回避する手法により、ネットワークの性

能を向上させる手法を提案した。シミュレーションによる評価の結果、データ発生頻度の低い環境において 20%、データ発生頻度の高い環境においては 100% 向上させられることが明らかになった。また、各ノードの持つ隣接ノード情報を拡張し、利用することで、提案手法の性能をさらに向上できることが分かった。

今後の課題として、通信品質が悪く伝送誤りが発生するような環境において、制御パケットの損失が発生した場合の影響を明らかにすることが挙げられる。

## 文 献

- [1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol.52, pp.2292-2330, Aug. 2008.
- [2] M. Buettner, G.V. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," *Proceedings of the 4th international conference on Embedded Networked Sensor Systems (SenSys 2006)*, pp.307-320, Oct. 2006.
- [3] E.-Y.A. Lin, J.M. Rabaey, and A. Wolisz, "Power-efficient rendez-vous schemes for dense wireless sensor networks," *Proceedings of IEEE International Conference on Communications (ICC 2004)*, pp.20-24, June 2004.
- [4] Y. Sun, O. Gurewitz, and D.B. Johnson, "RI-MAC: a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (SenSys 2008)*, pp.1-14, Nov. 2008.
- [5] 小南大智, 菅野正嗣, 村田正幸, 畠内孝明, 福山良和, 四蔵達之, "受信端末始動型間欠動作データ転送方式の性能評価," *信学技報*, IN2008-155, pp.139-144, Feb. 2009.
- [6] D. Chuluunsuren, 小南大智, 菅野正嗣, 村田正幸, 畠内孝明, "受信端末駆動型無線マルチホップネットワークにおける残余電力に基づいた長寿命化手法の提案," *信学技報*, AN2010, pp.11-14, May 2010.