

# 特別研究報告

## 題目

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

## 指導教員

村田 正幸 教授

## 報告者

速水 直

平成 23 年 2 月 16 日

大阪大学 基礎工学部 情報科学科

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

内容梗概

センサネットワークのような小型のバッテリーで動作する無線マルチホップネットワークにおいては、消費電力の抑制によるネットワークの長寿命化が重要な課題となっており、さまざまな方式が研究されてきた。特に、センサノードを間欠的に動作させ、起動していない期間にはスリープ状態にすることで、消費電力を大幅に削減することが可能である。このような間欠動作を行なう方式のうち、受信端末駆動型の通信方式である Intermittent Receiver-Driven Transmission(IRDT) 方式は、各ノードが間欠的に ID パケットを送信し、送信データを保持するノードが、その ID パケットに対する制御パケットを返信することで、通信を確立する。この方式では、送信ノードが複数の受信ノードを待ち受けることができるため、ノードの密度が高く通信頻度が小さい場合に優れた特性を有している。しかしながら、隠れ端末の関係にある複数のノードが送信データを保持している場合には、制御パケットの衝突が発生し、通信を行うことができない。特に、この衝突が連続的に繰り返されることが、IRDT 方式の性能を劣化させる要因となっている。そこで本報告では、IRDT 方式における制御パケットの連続衝突を回避し、性能を向上させるための手法を 3 種類提案し、シミュレーションによる比較評価を行う。その結果、再送制御の改良によって、既存の IRDT 方式ではパケット収集率が 40%程度しか得られなかった条件下でも、90%まで向上できることを示す。また、IRDT の特徴である隣接ノード情報を活用することで、さらに 5%向上できることを明らかにする。

主な用語

センサネットワーク, 隠れ端末, 連続衝突, 衝突回避, シミュレーション, バックオフ, ポーリング

## 目次

1	はじめに	5
2	受信端末駆動型通信方式 IRDT と連続衝突問題の概要	7
2.1	MAC 層: 既存の IRDT 方式のデータ通信の手法	7
2.2	ネットワーク層: マルチホップ転送とルーティング手法	7
2.3	制御パケットの連続衝突問題	9
3	連続衝突問題を回避する制御	13
3.1	SREQ の衝突発生時に行う制御	13
3.1.1	バックオフ機能	13
3.1.2	確率的に SREQ を再送させる機能	14
3.1.3	ポーリング機能	14
3.2	衝突時以外に行う制御	15
3.3	シミュレーションによる評価	15
4	隣接ノード情報を利用した性能改善	24
4.1	隣接ノード情報の拡張	24
4.2	機能ごとの隣接ノード情報の利用方法	26
4.2.1	バックオフ機能	26
4.2.2	確率的に SREQ を再送させる機能	27
4.2.3	ポーリング機能	28
4.3	シミュレーションによる評価	29
5	おわりに	33
	謝辞	34
	参考文献	35

## 目 次

1	IRDT 方式のプロトコル構成 . . . . .	8
2	IRDT 方式における MAC 層の動作シーケンス . . . . .	8
3	ネットワーク構成情報の例 . . . . .	10
4	SREQ パケットの連続衝突問題 . . . . .	12
5	バックオフ機能 . . . . .	16
6	スロットタイムの倍数の検討 . . . . .	17
7	確率的に SREQ を再送させる機能 . . . . .	18
8	確率 $p$ の値の検討 . . . . .	18
9	ポーリング機能 . . . . .	19
10	送信休止機能 . . . . .	20
11	ネットワークモデル . . . . .	21
12	各機能の性能評価 . . . . .	22
12	各機能の性能評価 . . . . .	23
13	バックオフ機能での隣接ノード情報の利用 . . . . .	26
14	確率的に SREQ を再送させる機能での隣接ノード情報の利用 . . . . .	27
15	ポーリング機能での隣接ノード情報の利用 . . . . .	29
16	隣接ノード情報を利用した各機能の性能評価 . . . . .	31
16	隣接ノード情報を利用した各機能の性能評価 . . . . .	32

## 表目次

1	シミュレーションの設定値 . . . . .	16
2	既存の IRDT 方式でパケットに載せている隣接ノード情報 . . . . .	24
3	拡張した隣接ノード情報 . . . . .	25
4	バックオフ機能でパケットに載せている隣接ノード情報 . . . . .	25
5	確率的に SREQ を再送させる機能でパケットに載せている隣接ノード情報 . . . . .	27
6	ポーリング機能でパケットに載せている隣接ノード情報 . . . . .	28

## 1 はじめに

近年、無線ノード同士が自律的にネットワークを形成するアドホックネットワークに対する関心が強まっている。その中で、複数のセンサ端末が無線によるネットワークを形成するセンサネットワークでは、特別なインフラストラクチャを必要とせずにセンシング情報を収集・管理することが可能なため、その広い応用性に注目が集まっている。一方、センサ端末の小型化や低コスト化などセンサネットワークにおいて解決すべき技術的課題は残っており、中でも容量が有限のバッテリーによって動作するセンサ端末においては、消費電力の低減化は重要な課題である。消費電力の低減化には様々なレイヤからのアプローチがあり、例えば、半導体技術の高度化や充電機器の追加、ノードにスリープ状態と起動状態を設けることで間欠的にデータを送受信するようなアクセス制御、通信距離を短くするためのマルチホップを前提としたルーティングアルゴリズム等が研究されている [1]。その中で本報告では、ローコストでの性能改善を目標として、間欠動作による低消費電力化を行なう無線マルチホップネットワークを対象とする。すなわち、ノードが間欠的に動作状態となり、それ以外の時間帯はスリープ状態となることによって消費電力を抑制するものである。ノードのスリープ状態時に消費される電力は起動状態時の数百から数千分の一であり、通信を行わない場合にスリープ状態となることは消費電力の観点から有効である [1] が、通信を行う際には起動するタイミングをノード間で制御する必要がある。間欠動作の制御方式は同期式と非同期式に分類できる。また、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、送信端末駆動型と受信端末駆動型の方式に分類される。さらに、非同期式の場合は送信端末駆動型 [2] と受信側駆動型 [3, 4] に分類され、そのいずれにおいても任意のタイミングで起動、通信を行うため同期を維持する必要はないが、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 番号や残余電力、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) を含む短いパケットを送信した後、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。パケット送信に先立ち、各ノードは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、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、次回の 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。そこで本報告では、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、バックオフアルゴリズムの適用、確率的に送信要求パケットを再送させる方法、ポーリングといった連続衝突を回避する手法を提案し、それらと既存の IRDT 方式とをデータパケットの収集率や到着までの遅延時間、各ノードの消費電力の面で比較して評価する。特に、IRDT の特徴である隣接ノード情報を適切に拡張、活用することによって、さらに性能を向上させることを目指す。

本報告は、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

ここでは IRDT の基本的な動作について説明する．図 1 では，IRDT 方式の主なプロトコルとそれぞれの位置づけを示す．本章では，まず IRDT におけるデータリンク層のプロトコルであるリンク確立手法について，次にネットワーク層のプロトコルである構成情報テーブル管理とルーティング手法について，最後に，回避すべき問題である SREQ の連続衝突問題について述べる．

### 2.1 MAC 層: 既存の IRDT 方式のデータ通信の手法

IRDT 方式における MAC 層の動作シーケンスを図 2 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，送信データを保持している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．データの受信が完了した後，受信側のノードは DACK パケット (DACK) を送信することで，通信が完了したと判断し，次は送信側のノードとして ID の受信を待機する．送信側のノードは DACK を受信することで，通信が完了したと判断し，保持していたデータを破棄し，スリープ状態に移行する．

### 2.2 ネットワーク層: マルチホップ転送とルーティング手法

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 (図 3(a))，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．例えば，図 3(a) に示すノード C において，前向きノードはノード A と B，横向き



OSI参照モデル	IRDT方式
ネットワーク層	<ul style="list-style-type: none"> <li>ネットワーク構成情報テーブル管理</li> <li>マルチホップルーティング</li> </ul>
データリンク層	<ul style="list-style-type: none"> <li>リンク確立制御</li> <li>衝突回避 (CSMA/CA)</li> </ul>
物理層	<ul style="list-style-type: none"> <li>スリープ制御</li> <li>Modulation (GFSK)</li> </ul>

図 1: IRDT 方式の protocol 構成

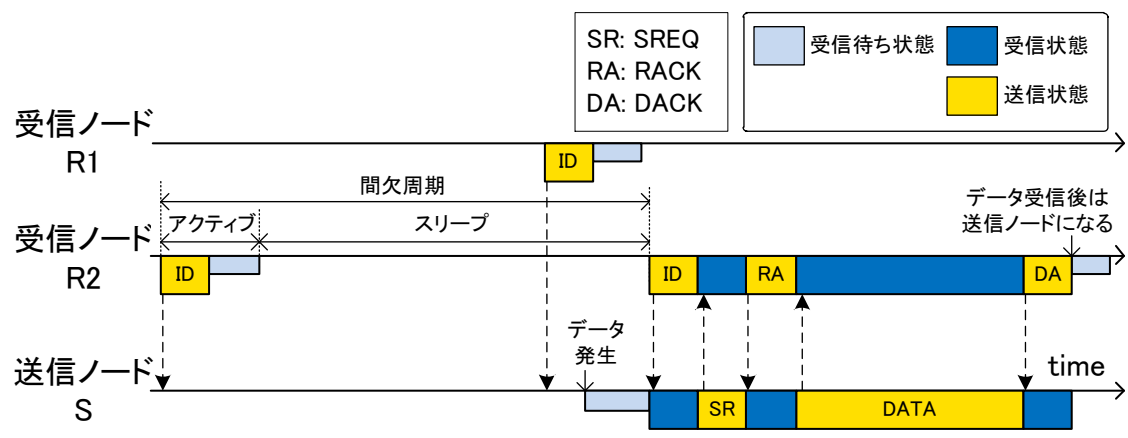


図 2: IRDT 方式における MAC 層の動作シーケンス

ノードはノード D，後向きノードはノード E である．各ノードは構成情報テーブルを参照しながら，受信した ID に対して SREQ を送信するかを判定することが，ルーティング方法の基本となる．図 3(a) のようなネットワークを想定する．図 3(a) において A-E はノードであり，各ノードと実線で結ばれているノードが無線回線の通信範囲内のノードであることを示す．各ノードは自身に隣接するノードの構成情報を保持している．例えば，ノード A が保持している構成情報テーブルは図 3(b) のとおりである．

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．現状のルーティング手法を以下のように定義する．

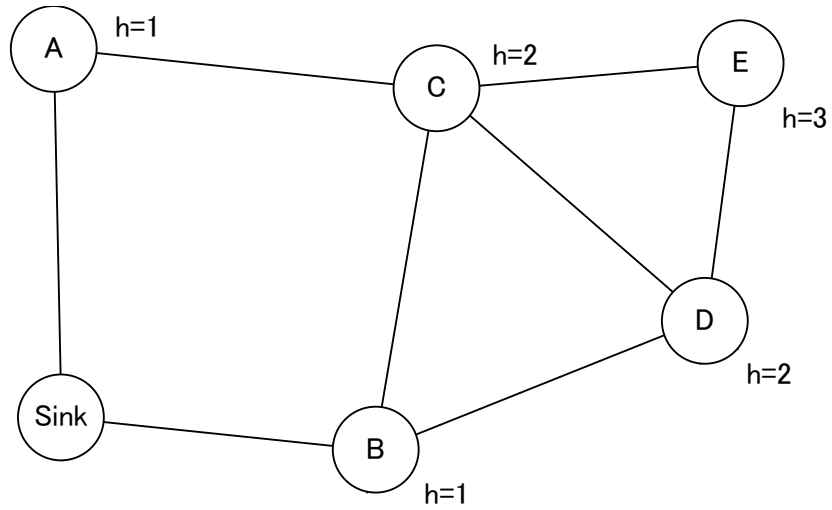
1. 自身の前向きノードからの ID を待ち受け，データ送信を試みる．
2. 全ての前向きノードへの送信に失敗した後に，前向きノードからの ID には 100%，横向きノードからの ID には 50% の確率でデータ送信を試みる．

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，転送を行わない．

本報告では，ID を送信することで，送信側のノードの SREQ 送信を制御して，ネットワークの性能を改善することを目的としており，また実際のハードウェアの適応させる際の拡張を容易とするため，バッファを持たないことを前提とする．そのため，データを受信したノードは，次に送信ノードとなり，前向きノードからの ID の受信を待機する．

### 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 4(a)，動作シーケンスを図 4(b) に示す．パケットを送信する際，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，他のノードが送信していることを検出した場合は，送信を停止することで衝突を回避する．しかしながら，パケットを送信するノードに対して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 4 では，受信側のノード R が ID を送信した後，同時に 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．データを保持しているノードの



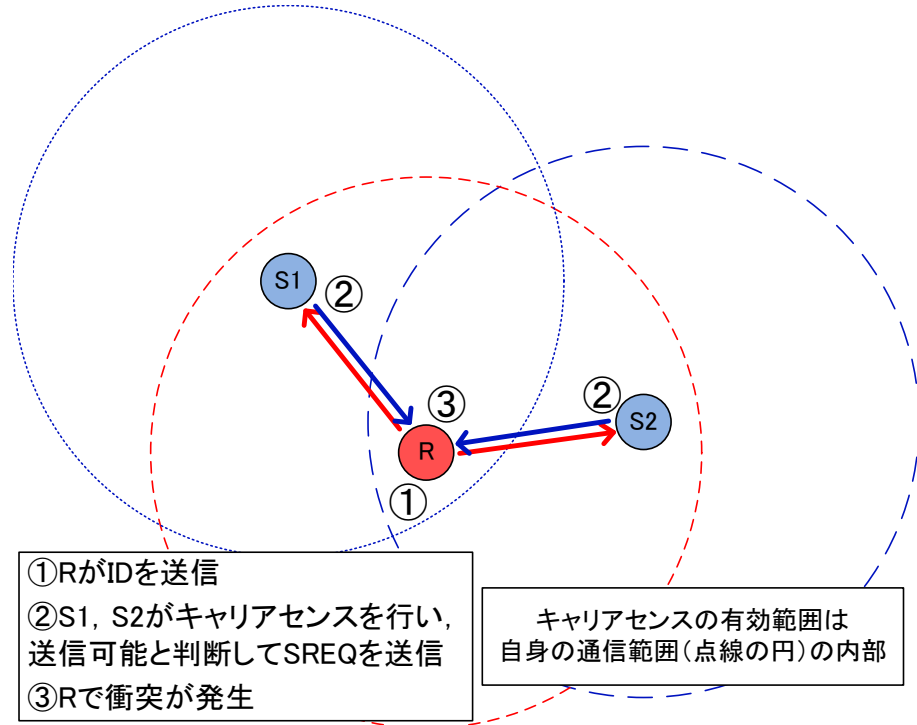
(a) ネットワーク構成

ノードBに関する 構成情報	A	B	C	D	E	Sink
	2	-	1	1	2	1
ノードCに関する 構成情報	A	B	C	D	E	Sink
	1	1	-	1	1	2
ノードDに関する 構成情報	A	B	C	D	E	Sink
	2	1	1	-	1	2
ノードEに関する 構成情報	A	B	C	D	E	Sink
	2	2	1	1	-	3

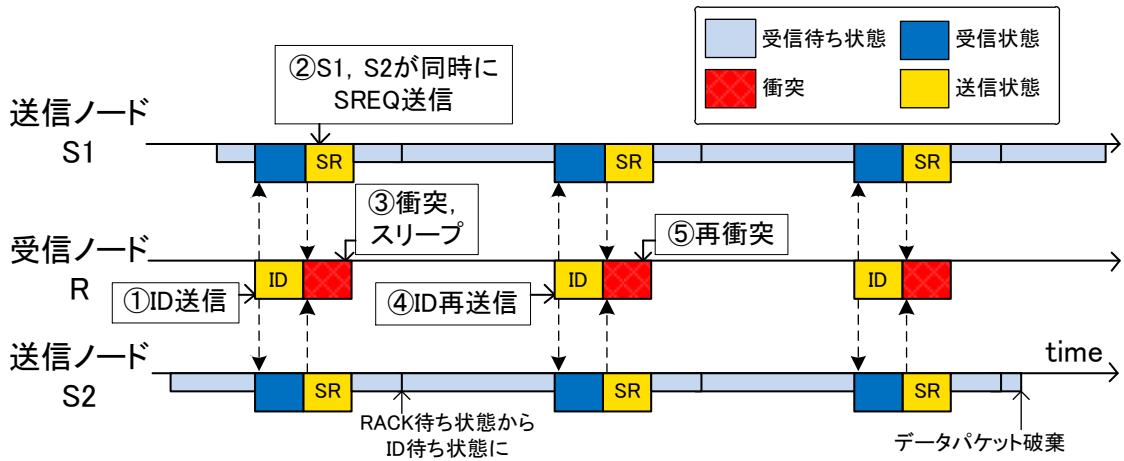
(b) ノード A が保持する構成情報テーブル

図 3: ネットワーク構成情報の例

どちらかが別の前向きノードにデータを送るなど、データを保持するノードの数が減らない限りは、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 におけるデータパケット収集率の低下やノードの消費電力の増大などの原因となっている。そこで本報告では、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。



(a) 衝突の様子



(b) 連続衝突の動作シーケンス

図 4: SREQ パケットの連続衝突問題

### 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、エラーを含むなど、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て適用することが困難であるため、それぞれ独立に評価を行い、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 5 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。自分宛の RACK を受信することができなかったノードは、再度 ID の受信待ち状態に戻る。

バックオフ機能では、送信される SREQ のうち少なくとも 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。スロットタイムの倍数を  $S$ 、競合するノード数を  $n$  とすると、 $P_{link}$  は以下の (1) 式で表され、その変化を図 6(a) に示す。

$$P_{link} = \begin{cases} 1 - \left(\frac{1}{S}\right)^{n-1} & (n = 2, 3) \\ 1 - \left\{ \left(\frac{1}{S}\right)^{n-1} + \frac{{}_S C_2 \times {}_n C_2}{S^n} \right\} & (n = 4) \end{cases} \quad (1)$$

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、(2) 式のように表せる。

$$\Delta T = T_{slot} \times S + T_{cycle} \times P_{link} \quad (2)$$

(2) 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 6(b) である．SREQ の衝突は主に 2, 3 ノードの競合によって発生すると考えて，図 6 から，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 7 に示す．この機能では SREQ の衝突発生後，衝突を検知したノードが 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．

これにより，競合しているノード数を  $n$  とすると，パケットを送信するノード数が二項分布  $B(n, p)$  に従うので，確率  $p$  による制御を行なった場合に，1 回の再送で通信に成功する確率  $P_{once}$  は式 (4) で与えられる．

$$P_{once} = n \times p \times (1 - p)^{n-1} \quad (3)$$

したがって， $x$  回までの ID の再送で通信に成功する確率  $P_{link}$  は以下ようになる．

$$P_{link} = P_{once} \sum_{i=1}^x (1 - P_{once})^{i-1} \quad (4)$$

ID を再送する回数，競合するノード数ごとの  $P_{link}$  の値は，図 8 のとおりである．1 回，3 回，10 回と ID を再送した場合，競合するノード数が 2, 3, 4 ノードの場合の組み合わせ SREQ の衝突は主に 2, 3 ノード，特に 2 ノードの競合によって発生すると考えて，確率  $p$  の初期値は 0.5，連続衝突は 10 回まで ID を再送信する．

### 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 9 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番号を選択し，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，受信した ID で，自身の ID 番号が SREQ 送信を許可を受けており，かつ受信した 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送信休止機能について説明する（図 10）．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，自身宛でない RACK を受信した場合に行われる．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，既存の IRDT 方式や前節で説明した 3 つの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 11 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．シミュレーションは，C++ 言語を用いて，IRDT の動作を記述したシミュレーションプログラムによって行った．シミュレーションの設定値は表 1 の通りで，データの発生は，シミュレーション開始時に，パケットの発生率とシミュレーション時間から 1 ノードあたりで発生するデータの個数を算出して，それをシミュレーション時間内にランダムに配置したデータ発生シナリオを生成し，そのシナリオに従ってデータを発生させている．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），さらにそれぞれに送信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パケットの収集率，データパケットの遅延時間，ノードの消費電力，SREQ パケットの衝突回数を以下のように定義し，図 12 に示す．また，それぞれの結果は 50 回のシミュレーションの平均であり，信頼区間は 95% である．

- データパケットの収集率

（シンクノードが受信したデータパケット数） / （発生したデータパケット数）



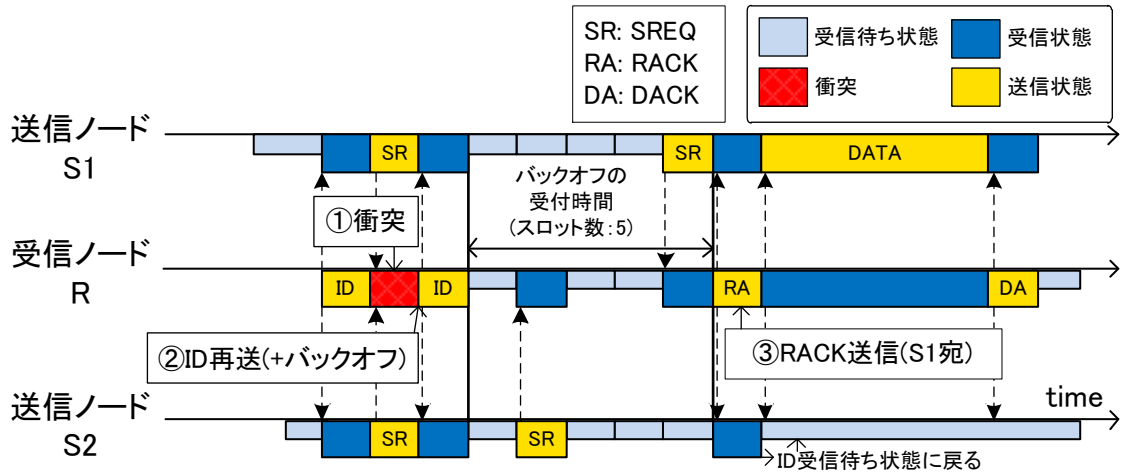


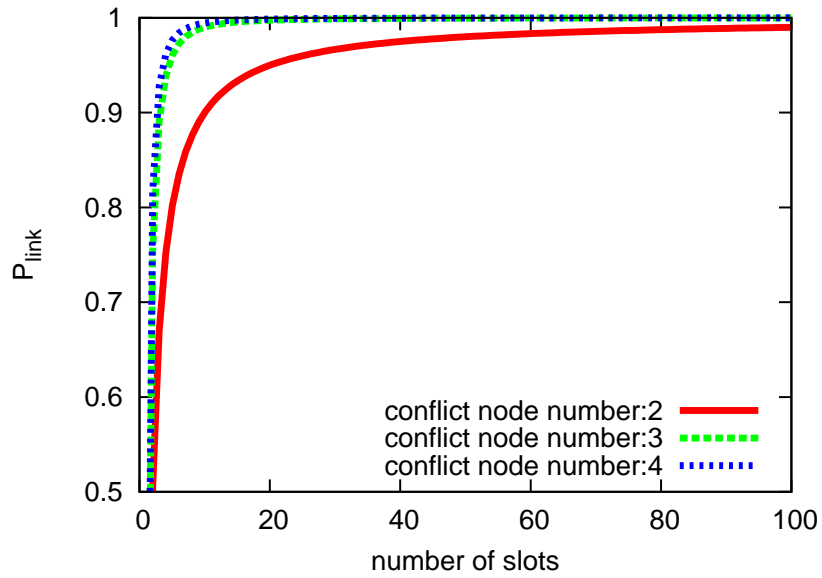
図 5: バックオフ機能

表 1: シミュレーションの設定値

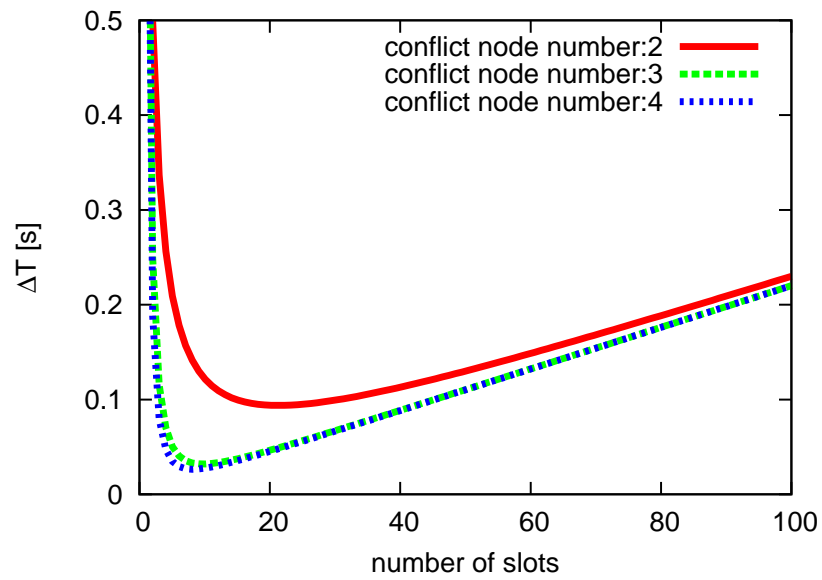
		パラメータ名	値
		間欠周期	1 秒
データ	値	データ保持期限	間欠周期 * 5 秒
試行時間	10000 秒	通信可能範囲	100 m
試行回数	50 回	最大中継回数	最小ホップ + 5 回
信頼区間	95 %	送信時電流	20 mA
スロットの倍数	40	受信時電流	25 mA
確率 $p$	0.5	スリープ時電流	0 mA
		データサイズ	128 byte
		無線速度	100 kbps

(a) 主な設定値

(b) データの発生モデル



(a) リンク確立成功率:  $P_{link}$



(b) 遅延時間の変位:  $\Delta T$

図 6: スロットタイムの倍数の検討

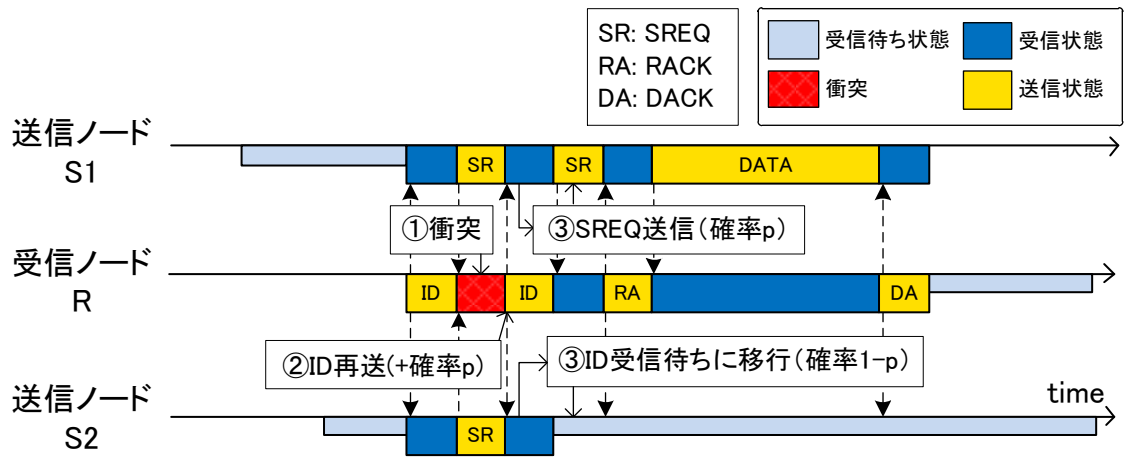


図 7: 確率的に SREQ を再送させる機能

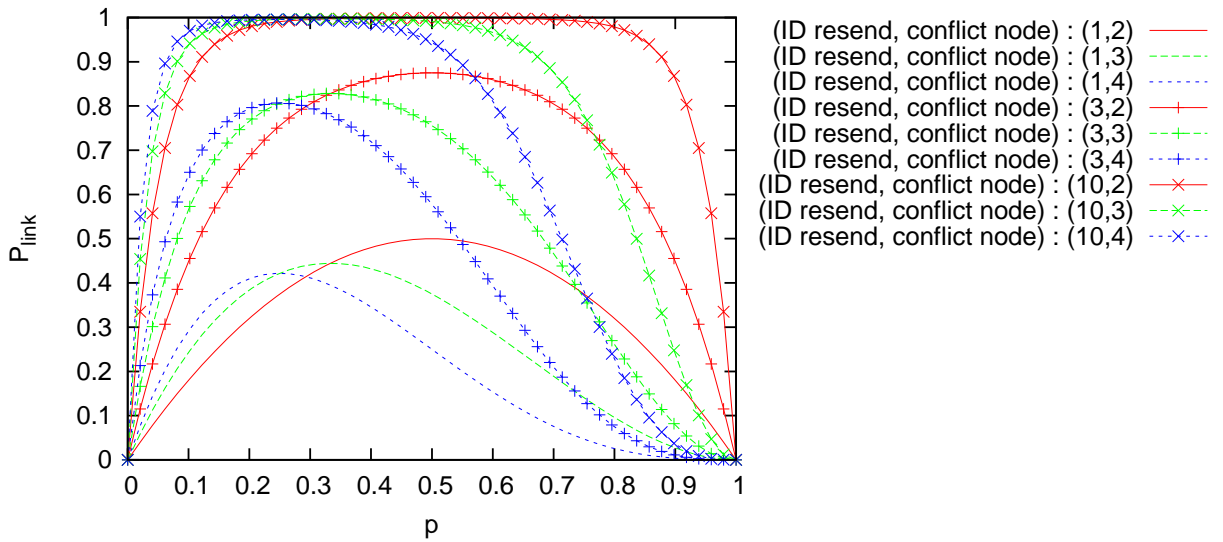


図 8: 確率  $p$  の値の検討

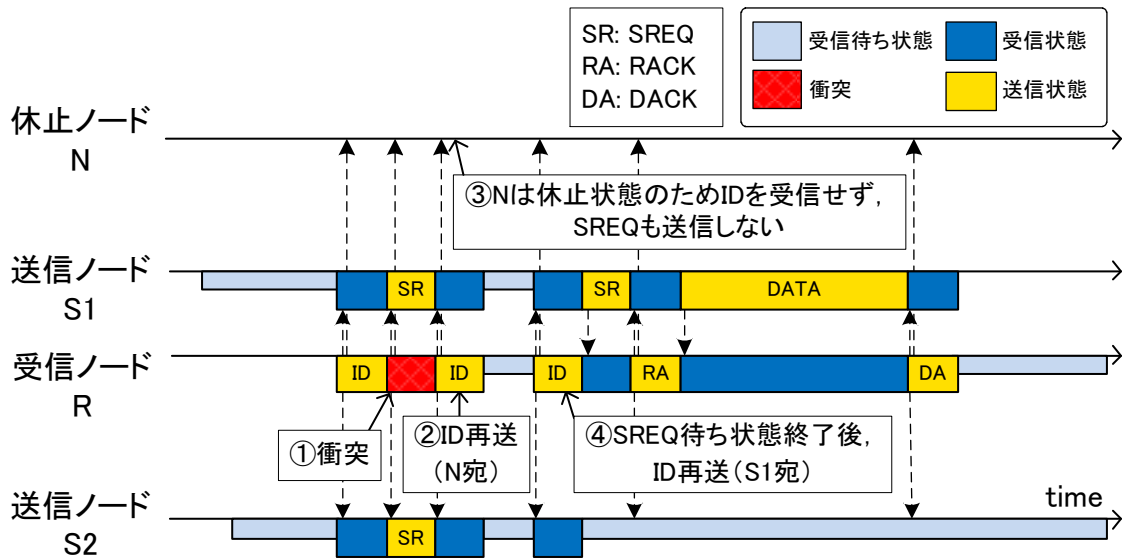
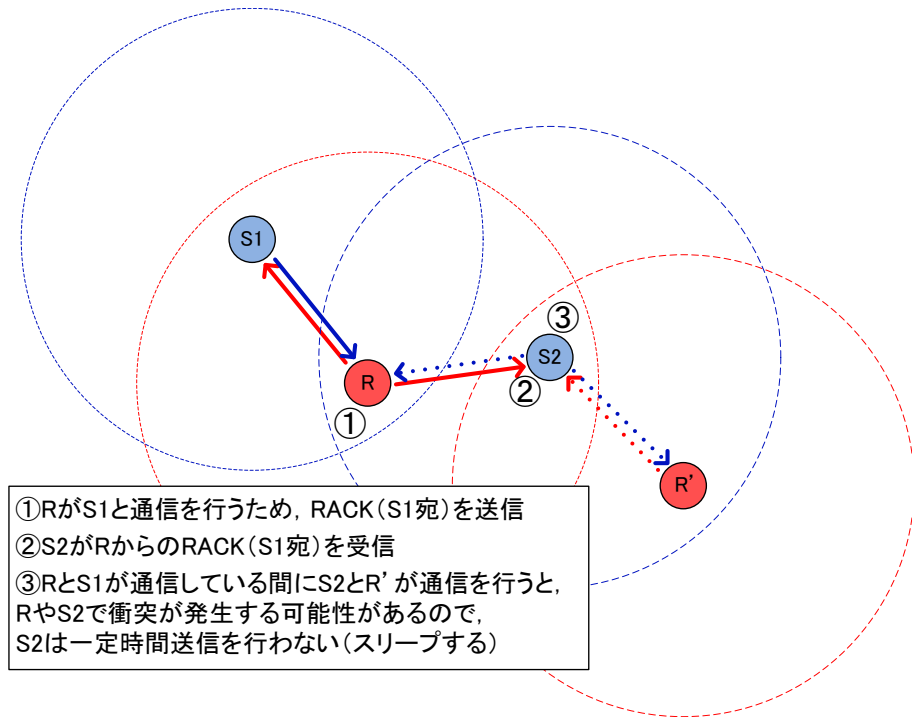


図 9: ポーリング機能

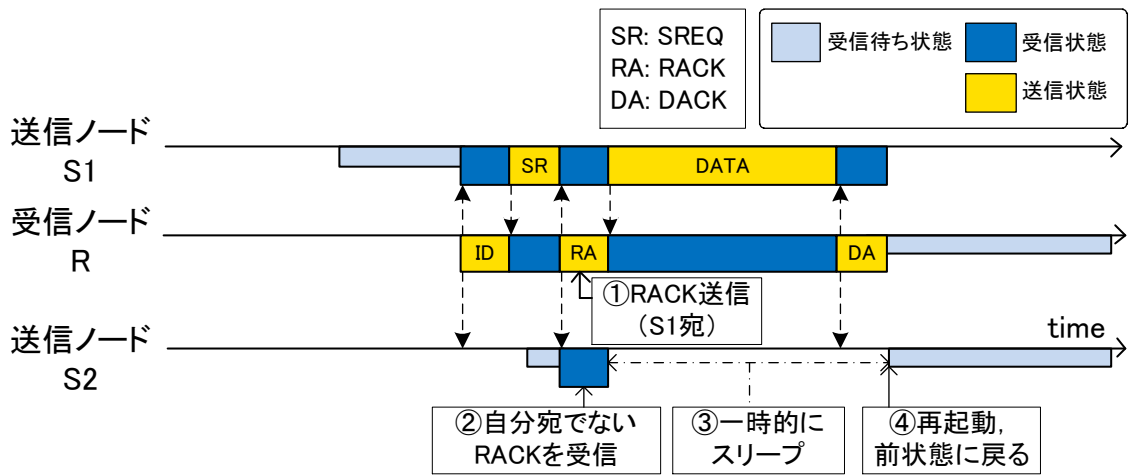
- データパケットの到着遅延時間  
(シンクノードが受信した時間) - (発生した時間) の平均値
- ノードの消費電力  
シミュレーション時間内の全センサノードの消費電力の平均値
- SREQ パケットの衝突回数  
各ノードが SREQ の受信待ち状態のときに衝突を検知した回数の平均値

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, データパケットの収集率, 遅延時間, ノードの消費電力の性能が向上している. データパケットの収集率では, データ発生頻度が  $0.01 \sim 0.15/s \times \text{node}$ , と低い場合では  $0.95$  を超え, データ発生頻度が  $0.15 \sim 0.25/s \times \text{node}$ , 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, 性能を向上させることが示された.

3つの手法を比較すると, ほぼ同じような性能ではあるが, データパケットの収集率, 消費電力, 衝突回数では, ポーリング機能が最も性能が良い. データパケットの到着遅延時間では確率的に SREQ を再送させる機能が優れているが, データパケットの収集率が最も低いので, その影響を考慮すると, 性能が良いとは言い難い. どの手法でも通信するノードを自身が選択できないため, この性能の差はポーリングによる衝突の回避が最も有効に働いて



(a) 送信休止機能の様子



(b) 送信休止機能の動作シーケンス

図 10: 送信休止機能

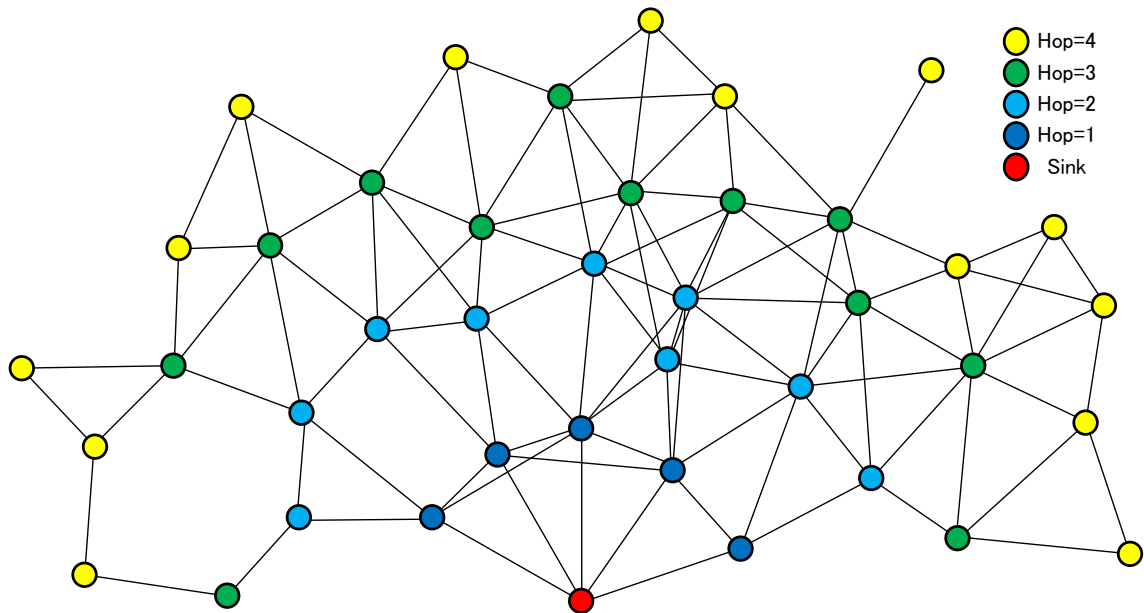
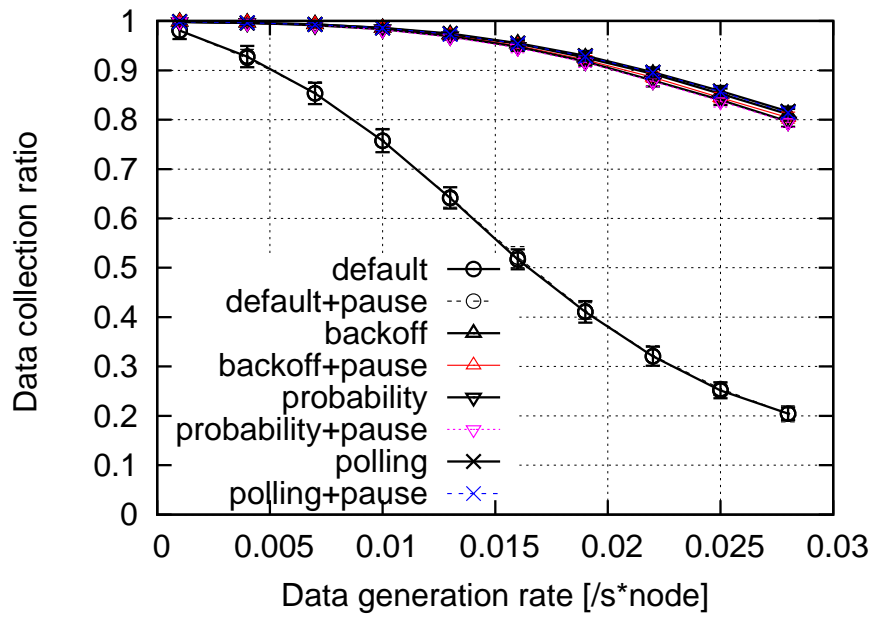


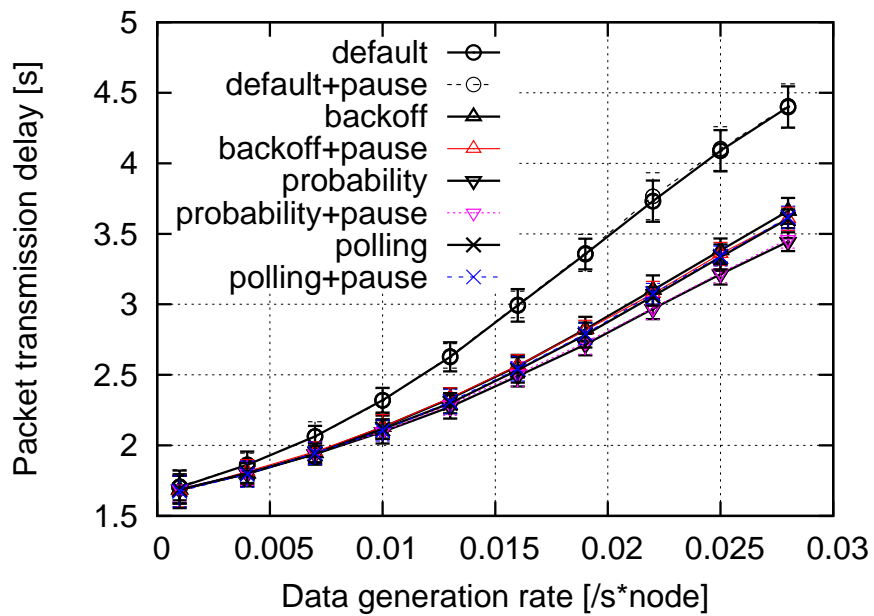
図 11: ネットワークモデル

いるためだと思われる。

また、送信休止機能の有無は、データパケットの収集率、遅延時間、消費電力にほとんど影響がないが、衝突回数がある程度抑制できている。これは、本報告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、データパケットの収集率などには影響しないためと思われる。データパケット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。

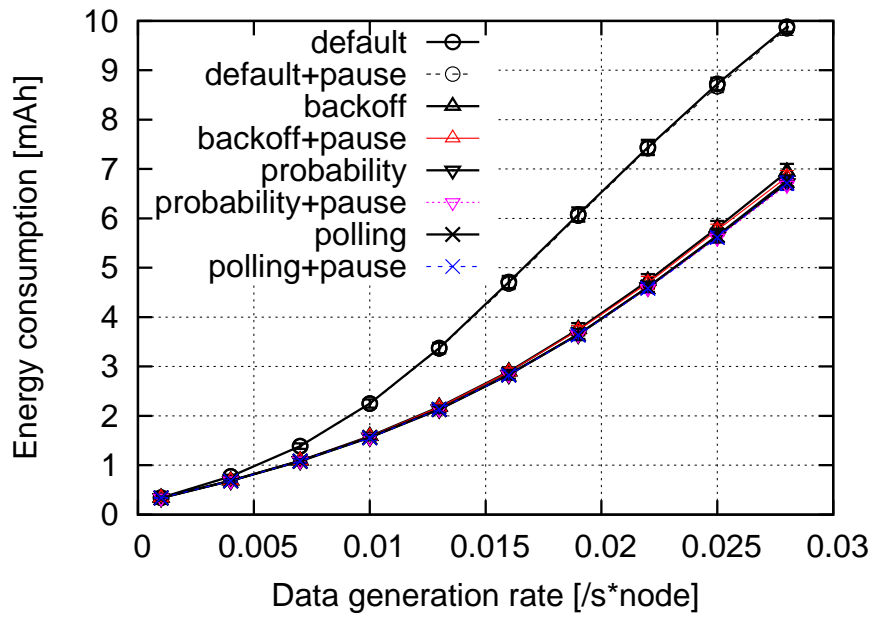


(a) 平均データパケット収集率

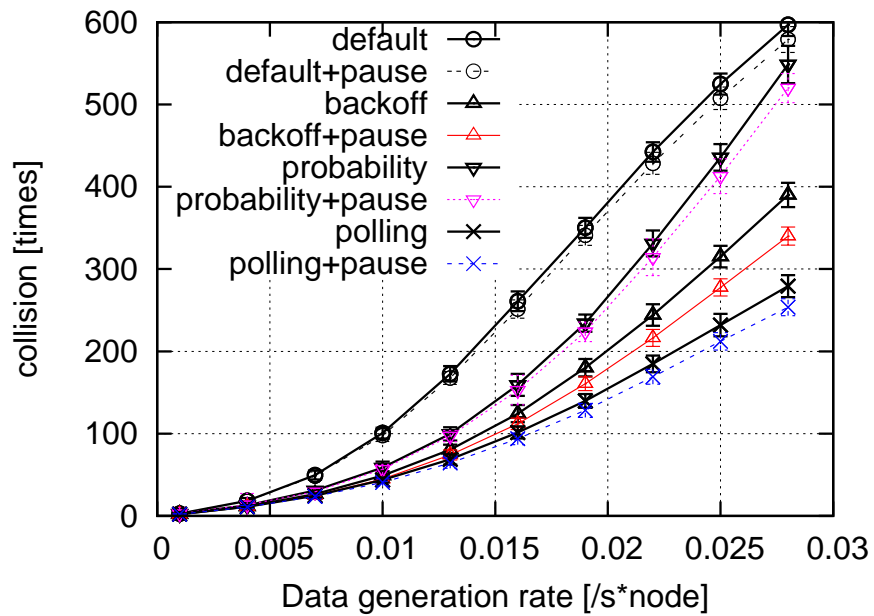


(b) 平均データパケット遅延時間

図 12: 各機能の性能評価



(c) 平均消費電力



(d) 平均衝突回数

図 12: 各機能の性能評価



## 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。また、隣接ノード情報の更新のためにパケットに載せている情報は表 2 のとおりである。本報告では、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。

### 4.1 隣接ノード情報の拡張

新たに追加した情報は、表 3 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に取得し、以降は更新しない。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。

表 2: 既存の IRDT 方式でパケットに載せている隣接ノード情報

パケットの種類	既存の IRDT 方式
ID	残余電力情報
SREQ	なし

表 3: 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	個々のノードが自身のデータを保持していない時間の平均値をとる	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

表 4: バックオフ機能でパケットに載せている隣接ノード情報

パケットの種類	通常時	バックオフ機能実行時
ID	残余電力情報 データ非保持平均時間 バックオフ機能実行フラグ	残余電力情報 データ非保持平均時間 バックオフ機能実行フラグ スロットタイムの倍数
SREQ	データ非保持平均時間	データ非保持平均時間 パケット破棄残り時間

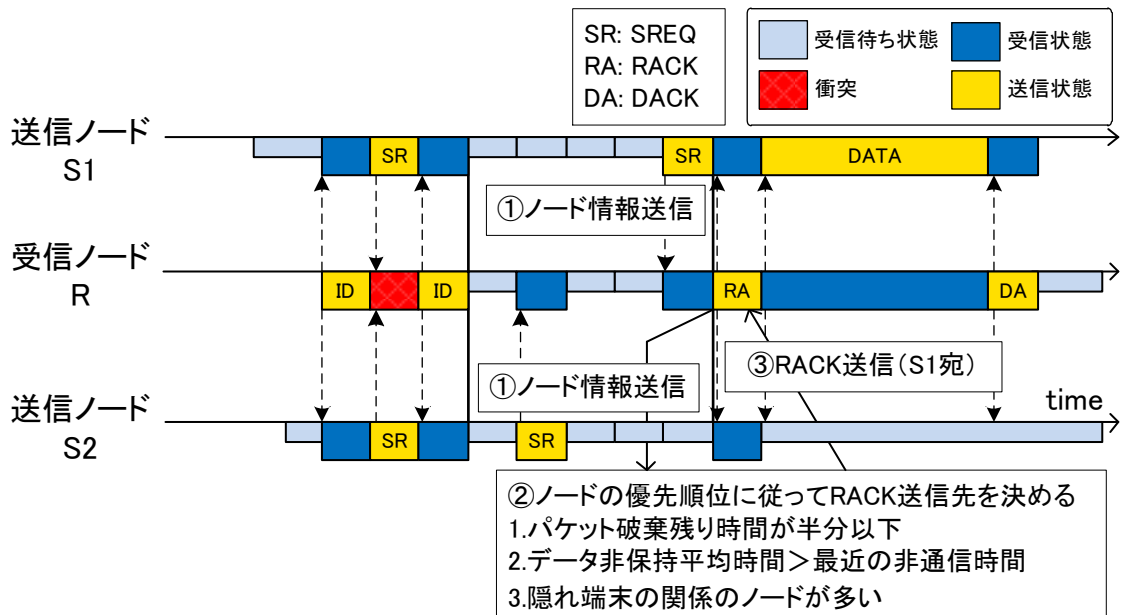


図 13: バックオフ機能での隣接ノード情報の利用

## 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

バックオフ機能で隣接ノード情報の更新のためにパケットに載せている情報を表 4 に示す。バックオフ機能では、図 13 のとおり、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パケット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  
複数存在する場合は、より残り時間が少ないノードを選択
2. 最新通信時刻からの時間 > 非保持平均となるノード  
複数存在する場合は、より開きが大きいノードを選択
3. 隠れ端末の関係にあるノードの数が多いノード
4. 最新通信時刻が古いノード

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、データパケットの収集率を向上させる。また、最後に通信を行ってからの時間がデータ非保持平

表 5: 確率的に SREQ を再送させる機能でパケットに載せている隣接ノード情報

パケットの種類	通常時	確率的な再送機能実行時
ID	残余電力情報 データ非保持平均時間 確率的な再送機能実行フラグ	残余電力情報 データ非保持平均時間 確率的な再送機能実行フラグ 確率 $p$
SREQ	データ非保持平均時間	

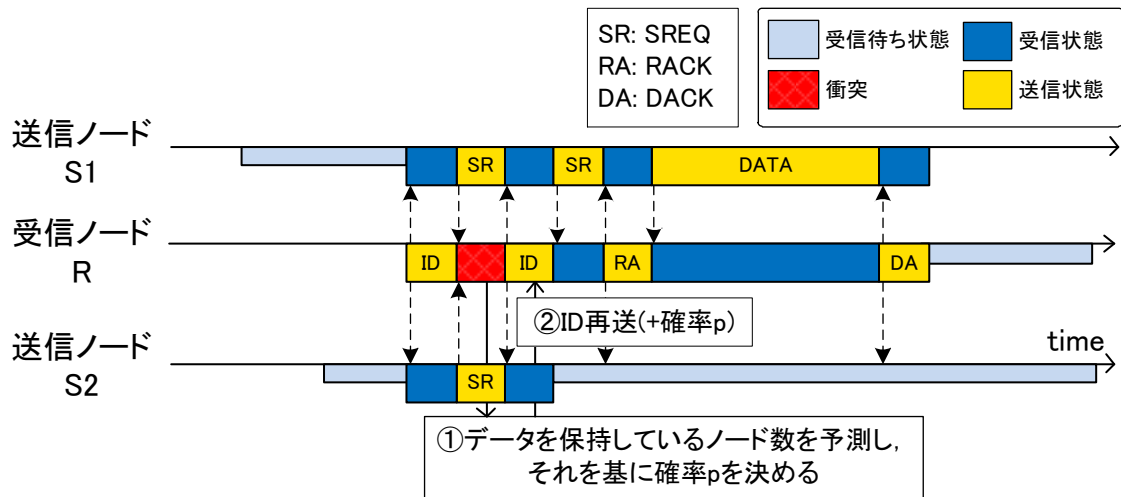


図 14: 確率的に SREQ を再送させる機能での隣接ノード情報の利用

均時間より長い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、データパケットの到着遅延時間や収集率を改善する。どのノードも以上2つに当てはまらない場合は、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て、隠れ端末同士によるパケットの衝突を抑制する。隠れ端末の関係にあるノードの数が多いいノードが複数存在する場合は、最新通信時刻が古いノードを優先する。

#### 4.2.2 確率的に SREQ を再送させる機能

確率的に SREQ を再送させる機能で隣接ノード情報の更新のためにパケットに載せている情報を表 5 に示す。この機能では、図 14 のとおり、最新通信時刻やデータ非保持平均時

表 6: ポーリング機能でパケットに載せている隣接ノード情報

パケットの種類	通常時	ポーリング機能実行時
ID	残余電力情報 データ非保持平均時間 ポーリング機能実行フラグ	残余電力情報 データ非保持平均時間 ポーリング機能実行フラグ SREQ 送信許可ノード番号
SREQ	データ非保持平均時間	

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

ポーリング機能で隣接ノード情報の更新のためにパケットに載せている情報を表 6 に示す。ポーリング機能では、図 15 のとおり、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複数存在する場合は、より開きが大きいノードを選択
2. 隠れ端末の関係にあるノードの数が多いノード
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、データパケットの到着遅延時間や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ノードを優先させることで、隠れ端末同士によるパケットの衝突を抑制する。

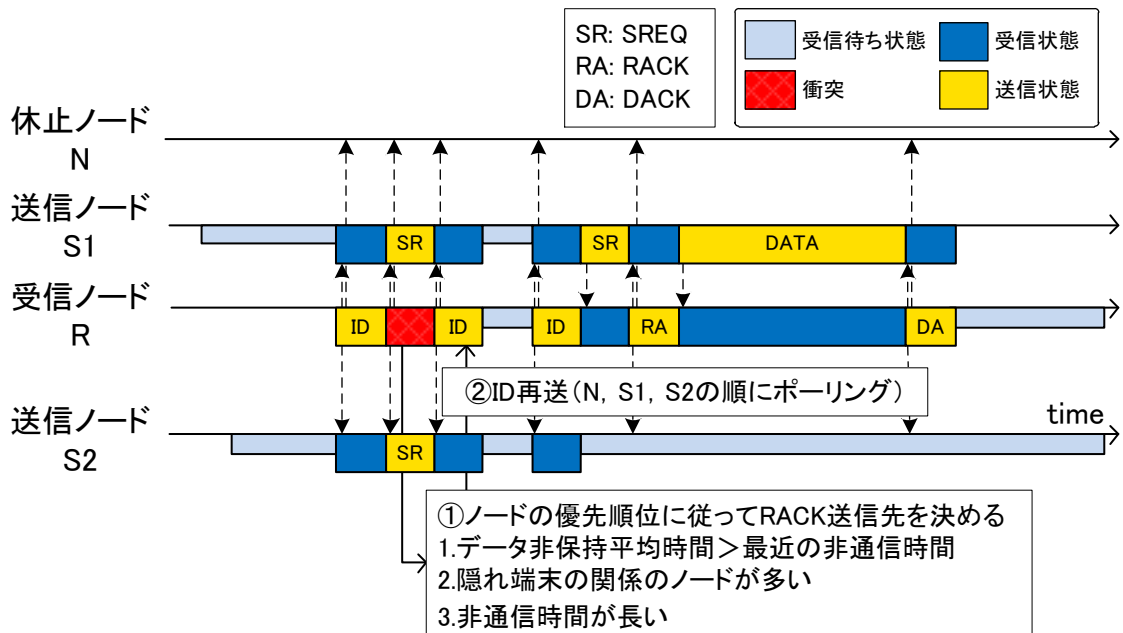


図 15: ポーリング機能での隣接ノード情報の利用

#### 4.3 シミュレーションによる評価

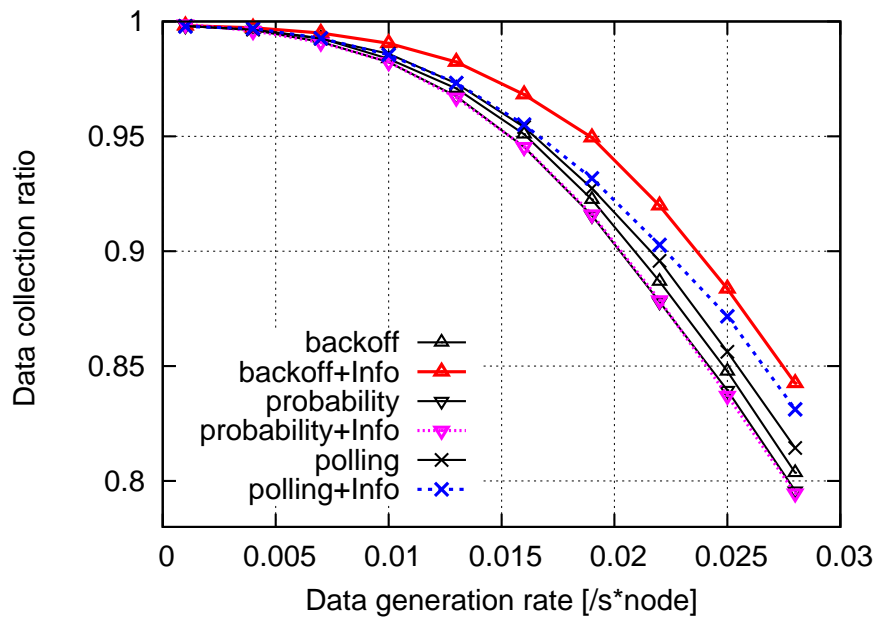
シミュレーションの設定は、3.3 節の設定と同じものを用いる。バックオフ機能，確率的に SREQ を再送させる機能，ポーリング機能それぞれを追加した場合と，何も機能を追加していない場合の比較は 3.3 節で行ったため省略し。図 12 での結果から，分散は小さいと判断して，信頼区間は省略する。また，このシミュレーションでは 3.2 節で説明した送信休止機能は常に適用している。

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パケットの収集率，遅延時間，ノードの消費電力，衝突回数を図 16 に示す。

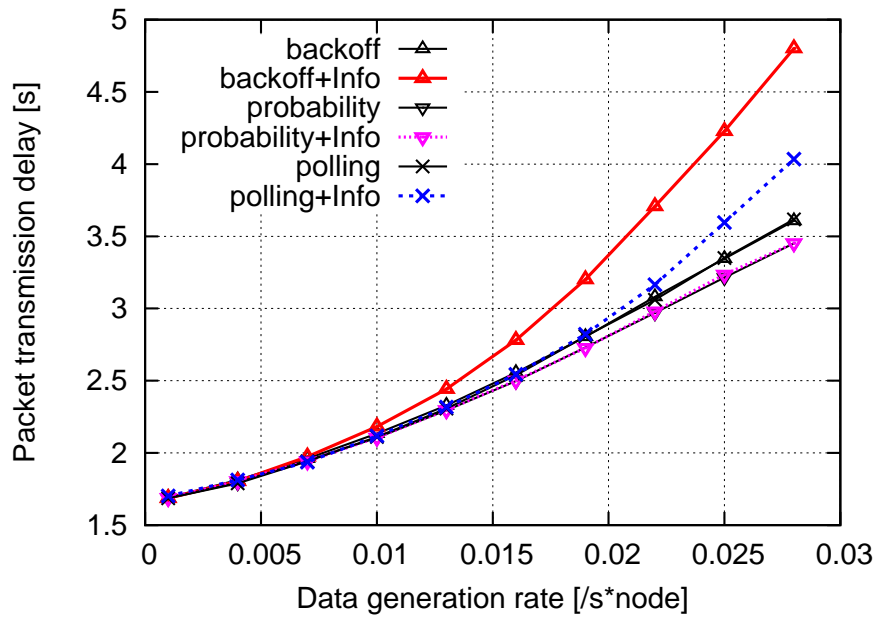
バックオフ機能，ポーリング機能では隣接ノード情報を用いた場合，データパケットの収集率が向上した反面，遅延時間や，わずかに消費電力が上がってしまっている。これはデータパケットの収集率が向上し，ノードがデータを保持する時間が伸びた影響によるものと思われる。データパケットの収集率では，バックオフ機能，ポーリング機能ともに向上しているが，バックオフ機能ではその時点でのパケット破棄までの残り時間がわかり，それを前提とした優先度を定められるので，ポーリング機能よりもパケットの破棄を抑えることがで

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、3.1.2 節の図 8 にもあるように、この機能は ID の再送信を繰り返すことで通信を行っているので、2～4 ノードの競合と予測し、確率  $p$  を 0.5～0.25 と変動させても 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だと思われる。

結果として、隣接ノードを用いて制御を行った 3 つの手法は、データパケットの収集率の面でデータ発生頻度が高い場合、隣接ノード情報を用いない場合より、バックオフ機能の性能は 4%、ポーリング機能の性能は 2% 向上した。これにより、隣接ノード情報を適切に用いることで、ネットワークの性能をさらに向上できることが示された。



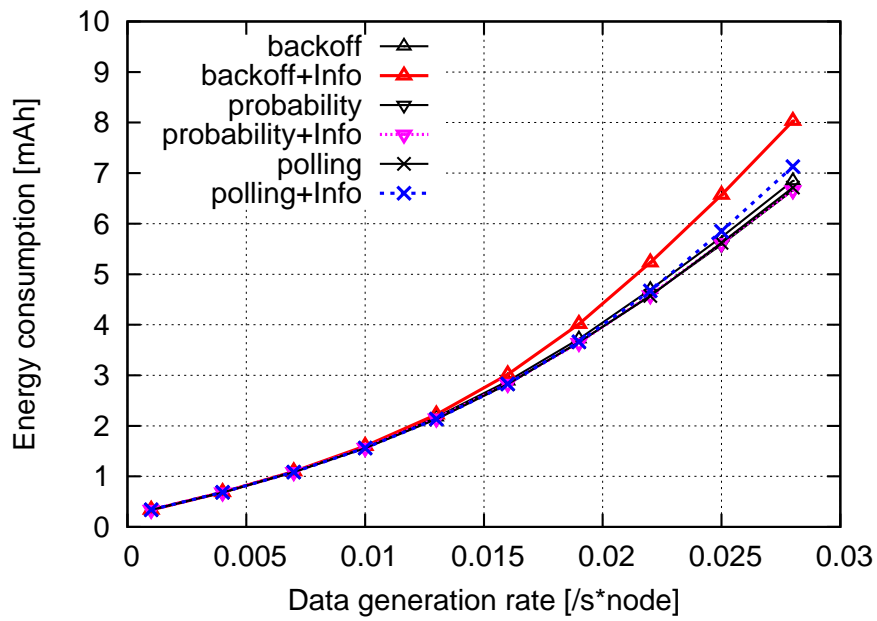
(a) 平均データパケット収集率



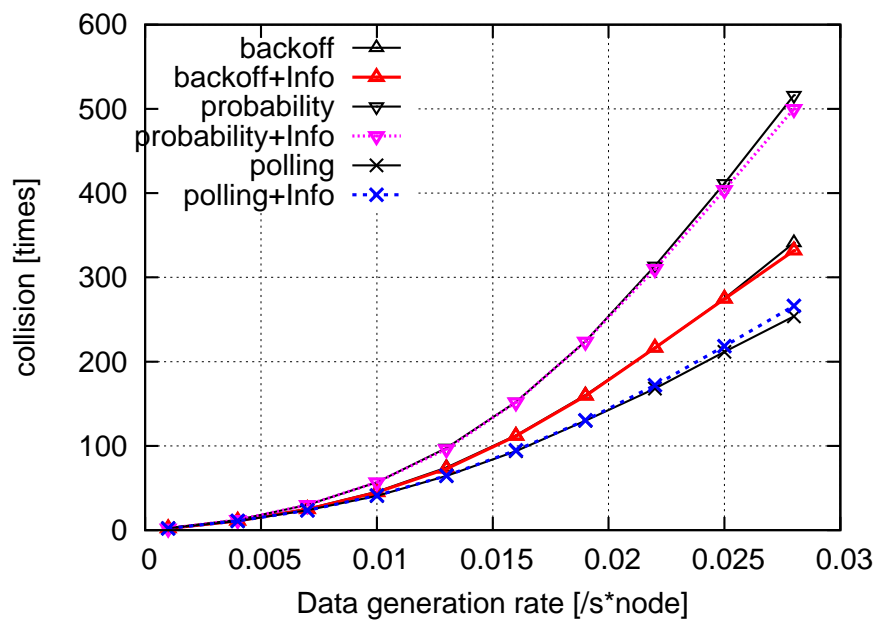
(b) 平均データパケット遅延時間

図 16: 隣接ノード情報を利用した各機能の性能評価





(c) 平均消費電力



(d) 平均衝突回数

図 16: 隣接ノード情報を利用した各機能の性能評価

## 5 おわりに

本報告では、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、再送を工夫することで、既存の IRDT からデータパケットの収集率を、80%前後であった条件下では、95%以上、40%前後であった条件下では、90%以上に向上でき、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、再送を工夫した制御からさらに 5%程度向上できることが明らかになった。

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。また、負荷が変動する場合や、残余電力の偏りに着目して、ネットワーク寿命を延ばせるような再送制御について検討する。

## 謝辞

本報告を終えるにあたり、御指導、御教授をいただきました大阪大学大学院情報科学研究科の村田正幸教授に深く感謝いたします。ならびに、本報告において直接ご指導を頂きました大阪府立大学総合リハビリテーション学部の菅野正嗣教授には、終始様々な相談に乗っていただき、多くの助言をいただきました。心よりお礼申し上げます。また、平素から適切なご助言をいただいた大阪大学大学院情報科学研究科の若宮直紀准教授、ならびに荒川伸一助教、大阪大学大学院経済学研究科の大下裕一助教に厚くお礼申し上げます。最後に、日頃から本報告の作成にあたり様々な質問に答えて頂きました村田研究室の小南大智氏、Chuluunsuren Damdinsuren 氏をはじめとする村田研究室の皆様方に厚くお礼申し上げます。誠にありがとうございました。

## 参考文献

- [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,” in *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,” in *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,” in *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, Mar. 2009.
- [6] Damdinsuren 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 (*AN2010-3*), pp. 11–14, May 2010.
- [7] Z. Liang and D. Zhao, “A priority-based polling scheme for clustering wireless sensor network MAC protocols,” in *Information Science and Engineering (ICISE 2009)*, pp. 2488–2491, Dec. 2009.
- [8] 譜久島亮, 菅野正嗣, 村田正幸, 林隆好, 四蔵達之, “間欠動作に基づく低消費電力アドホック・メッシュネットワークの性能評価,” 信学技報 (*IN2007-219*), pp. 361–366, Mar. 2008.
- [9] 畠内孝明, 福山良和, 石井美里, 四蔵達之, “メッシュネットワークのためのポーリングによる低消費電力型アクセス方式の提案,” 電気学会論文誌 C (電子・情報・システム部門誌), vol. 128, pp. 1761–1766, Dec. 2008.

- [10] 小南大智, 菅野正嗣, 村田正幸, 畠内孝明, 町田潤一, “受信端末始動型マルチホップ無線ネットワークにおける制御パケットの衝突を考慮した性能改善,” 信学技報 (*AN2009-32*), pp. 65–70, Oct. 2009.