

低消費電力MACプロトコルに対応した 無線センサネットワーク管理システムの一検討

浦田 佑貴^{1,a)} 吉廣 卓哉^{2,b)} 川橋 裕^{3,c)}

概要：無線センサネットワークを長寿命化するために、低消費電力 MAC プロトコルが盛んに研究されている。これらのプロトコルは、できるだけ長時間スリープを行い、ウェイク時に通信を行う。送受信ノード間でウェイクのタイミングを上手く同期することで、僅かなウェイク時間で一定時間毎にシンクにデータを収集できる。しかし、この種の無線センサネットワークにおいては、ノードとの直接的な通信接続が困難であるため、ノード故障や通信性能の低下を効率的に発見し、ネットワークの機能を長期的に維持するための管理が課題となる。本研究では、低消費電力 MAC プロトコルが動作する無線センサネットワークにおいて、シンクノードに収集されるパケットを用いて受動的にネットワークを管理するシステムを提案する。提案システムでは、シンクノードに集約されたパケットからネットワークの状態を可視化し、管理者が管理に必要なネットワークの状態を読み取る。評価を通じて、提案システムがネットワーク管理に役立つことを示す

1. はじめに

無線センサネットワーク (Wireless Sensor Networks; WSN) は IoT (Internet of Things) を実現する技術として注目されており、ネットワーク接続された各種センサを農業・医療・環境モニタリング等に活用することが期待されている。実用的な WSN を実現するためには、低い消費電力で信頼性の高いデータ通信を実現することが求められる。このために、様々な無線通信方式や無線センサ端末が研究開発されている。

WSN のための代表的な通信規格として IEEE802.15.4[1] が標準化され、この規格に対応した複数のハードウェアが販売されている。IEEE802.15.4 は様々な場面に対応できる通信機能を集約し多機能化したため、結果として消費電力性能はそれほど高いとは言えない。このため、センサデータを集約するためのさらに低消費電力な MAC プロトコルが多数提案されている。これらの MAC プロトコルでは、各ノードはできるだけ長い時間スリープし、送信ノードと受信ノードがタイミングを合わせてウェイクしてデータを

送受信する。B-MAC[2] や X-MAC[3] は、送信ノードが主導してタイミングを合わせる送信ノード主導型 MAC プロトコルであり、例えば B-MAC は、ウェイク周期よりも長いプリアンブルを送出することにより、データ送信時には受信ノードが確実に起きることを可能にする。RI-MAC[4] や RC-MAC[5] は、受信ノードが受信可能なタイミングを知らせるために定期的にビーコンを送出する受信ノード主導型 MAC プロトコルである。我々の研究グループでは、RI-MAC を拡張した MAC プロトコルに経路制御プロトコルを組み合わせることで、従来よりも大幅に消費電力を抑える手法を提案している [6]。

一方で、実際に WSN を運用するためには、ネットワーク上で発生する問題を的確・迅速に検知し、これらに対して有効な対策を講じることが必要である。このために、WSN の状況を把握し、必要に応じた対応を可能にする情報を提供するネットワーク管理システムが必要である。実際に、WSN の管理を目的として、インターネットの管理に用いられる SNMP[7] と同様に、各センサに対する問い合わせを用いた方法が提案されている。しかし、先述の省電力 MAC を用いる場合には、各センサが可能な限り長時間のスリープを行うため、マルチホップ通信による問い合わせは応答が非常に遅くなり、実用的ではない。このため、管理に必要な情報を常にシンクに送信し蓄積しておき、これを管理に活用する手法が提案されている。Ramanathanらは、シンクの情報から決定木を構築し、問題の根本原因

¹ 和歌山大学大学院システム工学研究科
930 Sakaedani, Wakayama, 640-8510, Japan
² 和歌山大学システム工学部
930 Sakaedani, Wakayama, 640-8510, Japan
³ 和歌山大学システム情報学センター
930 Sakaedani, Wakayama, 640-8510, Japan
a) s171009@center.wakayama-u.ac.jp
b) tac@sys.wakayama-u.ac.jp
c) yutaka@center.wakayama-u.ac.jp

を推定するシステム Sympathy を提案した [8]。Liu らは、シンクの情報から確率的な因果ダイアグラムを作成し、根本原因を推定する手法 PAD を提案した [9]。

これらの手法は、予め想定した事象を対象として、それらの因果関係を推定することで、問題の原因を特定しようとしている。しかし、インターネットにおけるネットワーク管理と同様に、WSN の管理においても想定外の問題が発生し得ると考えられる。このため、根本原因を特定するだけではなく、常にそれ以外の可能性も考慮して注意深くネットワークを監視する必要がある。また、問題が顕在化する前に隠れた予兆を検知し、これに対処することで未然に問題を防ぐことも重要である。ネットワークの調子がなんとなく悪い、というような状況を解消しておくことは、ネットワーク性能の維持という観点からも望ましい状況である。既存研究を見渡すと、上記のような WSN の監視を実現するための WSN の可視化法もいくつか提案されている [10]。しかし、ネットワーク管理の観点からの設計がなされておらず、管理のための有用性評価も不十分である。

本研究では、ネットワーク管理における管理要求のうち、構成管理、障害管理、および性能管理の 3 要求に着目し、これらを実現するための実用的な WSN 管理システムを設計・構築する。その上で、シミュレーションデータを用いて提案システムを評価し、上記 3 要求を満たす WSN 管理が可能であることを明らかにする。

本論文の構成は以下の通りである。第 2 章では、WSN 管理における要求について議論したうえで、提案システムの設計方針を説明する。第 3 章では、提案する無線センサネットワーク管理システムの詳細を述べる。第 4 章では、WSN 管理の観点から提案システムの評価を行い、最後に第 5 章でまとめる。

2. 要求定義と設計

2.1 管理システムへの要求

本研究では、何らかの省電力 MAC プロトコルが用いられており、SNMP のような各ノードへの問い合わせが実行できない WSN を想定する。この WSN を管理するために、各ノードがセンサで測定したデータを格納したパケットに、管理に必要な情報を付加してシンクに集めることで、シンクノードに集まった情報から適切な管理が行えるようにする。

WSN の管理においては、ネットワークが常に正常に動作しているかを監視する必要がある。このためには、発生した問題を迅速に検知できるだけでなく、通信性能の低下や経路の変更等、その兆候を捉えることが重要である。シンクに集められたデータを可視化することで、これらを迅速かつ的確に把握できるようにすることが、WSN の管理において重要である。

従来の（インターネットにおける）ネットワーク管理に

おいては、構成管理、セキュリティ管理、資源管理、性能管理、障害管理の 5 つの管理項目が重要とされている [11]。これらの管理項目の具体的な内容を以下に示す。本研究では、WSN においては特に構成管理、性能管理、障害管理の 3 つが重要であると考え、これらを実現する WSN 管理システムを設計する。

構成管理

構成管理とは、WSN を構成する物理的要素と論理的要素を個々に管理し、加えてノード同士の繋がりにあるリンクの状態を管理することである。このとき管理される情報がすべての管理項目の基礎となるため、常に最新の状態に保つことが重要である。WSN を構成する物理的要素とは、センサノードやシンクノードであり、論理的要素とは、それらの物理的要素が持っているノード ID や通信速度、データ生成間隔やスリープ時間といった設定情報である。

障害管理

障害管理とは、WSN で起きるどのような事象を障害と見なすのかを定義し、定義された障害ごとに検出や対策、または予防の方法を検討して実施することである。加えて、各センサノードでのログ情報を管理することも障害管理の 1 つである。障害検知の方法としては各機器に状態の問い合わせをおこなう能動型と、各機器からの報告を待つ受動型が存在する。

性能管理

性能管理とは、ネットワークの性能を一定のレベルに維持することを目的とした管理である。ネットワークの性能としては、パケットロス率、トラフィック量、輻輳回数、CPU やバッファの使用率などがある。これらの値に閾値を設けて監視をおこない、閾値を上回った際の対処についても決めておく。加えて、これらの値を定期的に測定し、結果を蓄積していくことでネットワークの動向を把握することも重要である。

資源管理

資源管理とは、ネットワーク機器とは別に、ネットワークケーブルの配線や停電時の電源管理、冷却のための空調などの設備に関する管理をおこなう。

セキュリティ管理

セキュリティ管理とは、ウィルスや不正アクセスからネットワーク内の資産を保護し、それぞれの脅威に対する対策を実施することである。

本研究において、資源管理とセキュリティ管理を考慮しない理由を述べておく。資源管理については、無線で構築される WSN においてケーブルや電源の管理は必要なく、消費電力も小さいため空調の管理も省略できると考える。セキュリティ管理については、各センサノードの通信を第 3 者に盗聴されないようにする管理が考えられるが、これは、通信の暗号化によって対策できると考えられる。

2.2 管理システムの設計

構成管理、障害管理、および性能管理を行うために、必要な情報を収集し、それらを適切な形で表示する必要がある。管理に必要な情報の収集は、各ノードが自身で観測できる値のうち、WSN 管理上有用であるものを、データパケットに付加してシンクノードまで届ける。通信負荷は付加した値の分だけ増大するが、情報収集に複数ノードの協調を必要としないため、制御が複雑になることはない。

表示（ユーザインタフェース）では、まず、WSN の全体像を把握するために、パケットがシンクに収集される経路木をトップ画面に配置する。各ノードを正しい位置に配置し、その時点で通信に用いられるリンクが表示されるため、現在のネットワークの構成を直感的に把握できる。また、経路木に過去の経路を重ね合わせ表示することで、時系列における構成管理を行えるようにする。

障害管理としては、ノードの故障を対象とする。長時間データが到着しない場合に故障とみなすことができる。これは、データ未到着時間が大きいことを自動検出し、警告表に表示することで、管理者に知らせる。また、管理者が障害の状況を正確に把握できるように、関連するデータの詳細を一覧表および折れ線グラフにより確認できるようにする。

最後に性能管理は、電力残量、パケットロス率、遅延時間等の通信性能指標を監視することで行う。これらの指標があるしきい値を超えて悪化した場合に警告表に表示することで、管理者は性能の監視を行える。また、状況を正確に把握するために、折れ線グラフやデータ一覧表を用いて実際の値を確認することが可能である。

WSN を管理するための本提案システムの特徴として、下記に示す機能を実装した。

- (1) 変更された経路の表示ある時点の経路木に、その時点より前と後に変更された経路を重ね合わせて表示する。そうすることで、過去のいつ頃にどのような経路であったか、もしくは後のいつ頃にどのような経路になるのかが一目でわかり、ノードの故障による経路変更が見つけやすくなる。
- (2) ネットワーク性能の低下の通知ネットワークの性能を示すデータ項目に対して、閾値を超えていた場合には警告表に表示することで管理者に知らせる。なお、時間が経つほどデータは蓄積され、過去の時点では判断できなかった警告も検出できるようになる。
- (3) 管理効率を上げるための直感的操作各ノードが持つ情報を、経路木表示に反映させて可視化をおこなうことで管理者を支援する。加えて、ネットワークトポロジ内のノードに、カーソルを合わせたリクリックすることで情報を表示し、より直感的な操作をおこなうことができる。

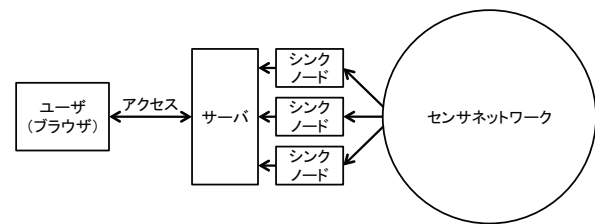


図 1 システム構成

3. 提案する WSN 管理システム

3.1 システム構成

本システムは、シンクノードが収集した情報を入力として、ネットワークの状態を可視化する機能を持つ。システム構成を図 1 に示す。シンクノードは、WSN から送信されるデータをリアルタイムにサーバに集約する。シンクノードは一般に複数存在すると考えられるが、単一である場合には、シンクノードはサーバを兼ねても良い。サーバは Web サーバ機能を持ち、管理者が Web ブラウザを用いてサーバにアクセスすることで、WSN の情報が可視化された Web ページ（アプリケーション）を利用する。

3.2 パケットへの管理情報の付加

WSN を適切に管理する上で必要と思われるデータ項目の値を、各ノードがセンシング時にパケットに格納することで、シンクに伝える。必要であると考えられる値として、次の 18 項目を挙げた。

- (a) シンクノード到着時の受信時刻
- (b) シンクノード ID
- (c) 送信元ノード ID
- (d) シーケンス番号
- (e) 親ノード ID
- (f) センシングデータ送信回数
- (g) センシングデータ受信回数（オーバヒア含む）
- (h) ACK 送信回数
- (i) ACK 受信回数（オーバヒア含む）
- (j) 制御メッセージ送信回数
- (k) 制御メッセージ受信回数
- (l) センシングデータ受信回数（オーバヒア含まない）
- (m) ACK 受信回数（オーバヒア含まない）
- (n) センシングデータ生成時刻
- (o) ウェイク時間
- (p) スリープ時間
- (q) センサ位置情報（センサが GPS 等を備えている場合）
- (r) 電力残量

ここで、オーバヒアとは、自分が受信対象ノードではないのに、電波が届くことで受信されてしまう現象である。オーバヒアにより受信されたパケットの情報は、場合に

よって有効利用できる可能性がある。

各データ項目を説明する。(d)は、センシングデータに一意に付けられ、そのノードの何番目のセンシングデータかを表す番号である。(f)(g)はセンサノードが送信・受信したセンシングデータの回数の累計を記録したものである。(h)(i)はデータを受信したことを知らせる確認応答ACKを送信・受信した回数の累計である。(j)(k)はネットワークの経路制御プロトコルが配送木を構築するために送受信する制御メッセージ数の累計である。なお、ここで説明した(f)(h)(j)の送信回数とは、自身のデータを送信した回数だけでなく、周囲のノードのデータを受け渡した際の回数も含まれる。(o)(p)はノードがデータを送受信するために起きていた時間と、何もせずに動作を休止していた時間のそれぞれの合計である。

センサの位置情報は、センサ自身がGPS等により位置取得機能を持っている場合には、パケットに付加して収集しても良い。しかし一般には、消費電力を節約するために、位置取得機能を持たないセンサ端末は多い。この場合には、センサ設置時に各センサの位置を記録しておき、テキストデータとして入力することで、システムの画面にセンサの位置情報を反映させることができる。

上記の各項目から何らかの計算をすることにより有用な値となるものもある。以下の項目は、項目(a)-(r)の値から計算によって算出される。

- (s) パケットロス率
- (t) 遅延時間
- (u) 前回の受信からのインターバル

(s)は、自身の送信したデータの回数とACKの受信回数を比較することで、そのノードが送信したデータのロス率を算出する。よって計算式は、 $s = \{((f)-(m))/(f)\} \times 100[\%]$ となる。(t)は、データを生成してからシンクノードに到着するまでにかかった時間である。計算式は $t = (a)-(n)$ となる。(u)は、ひとつ前のセンシングデータと今回のセンシングデータの(a)の値から算出でき、n回目(第n回)に到着したセンシングデータの(a)を a_n と表すと、 $u = a_n - a_{n-1}$ となる。

上記(a)-(u)の値を、それぞれに設定したしきい値と比較することによって異常かどうかを判断し、システムの画面や警告表に反映させることで、ユーザの気づきを促す。

3.3 ユーザインタフェース

3.3.1 画面遷移

提案システムの画面遷移を図2に示す。トップ画面は、経路木表示領域、データ一覧表、警告一覧表を含む。経路木表示領域には、シンクに集められた情報から推定されるネットワークの経路木(データが集約される経路を合成した木)が表示される。データ一覧表には、シンクが受信したデータと、これに付加された各種データが一覧表として

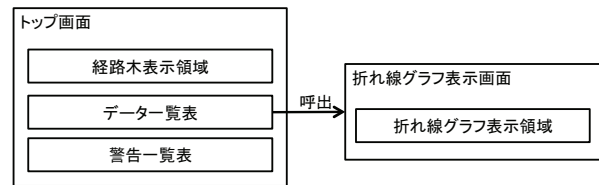


図2 ユーザインタフェース

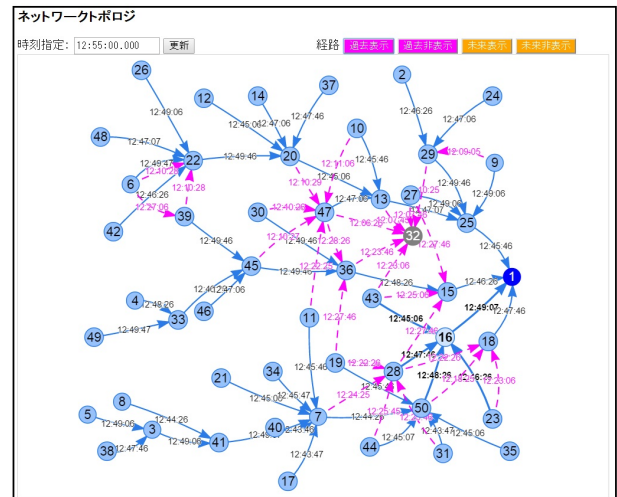


図3 経路木表示(過去の経路を重ねて表示した場合)

表示される。警告一覧には、管理者の気づきを補助するために、ネットワーク内で発生した重要事項を自動的に推定してリスト化する。トップ画面のデータ一覧表の各項目にはボタンがあり、これを押下(クリック)することにより、データを折れ線グラフとして表示するグラフ表示画面(別窓)を開く。ネットワークの管理者は、これらの画面を用いて種々のグラフを閲覧する操作を行うことで、ネットワーク管理に必要な情報を得る。

3.3.2 経路木表示領域

経路木表示領域には、現時点での最新の経路木を表示する。これにより、テキストデータを見るだけでは把握が困難なデータの配送経路を、管理者が把握するのを助ける。最新の経路木だけでなく、過去の時刻を指定することによって、その時刻の経路木を表示することが可能であり、時間経過の中で経路がどのように変化したかを表示できる。図3に、過去と現在の経路木を重ねて表示した例を示す。各リンクに表示された時刻は、そのリンクの親ノードからのパケットが最後に届いた時刻を表し、パケットの遅延状況を直感的に把握できる。また、各リンクのパケットロス率が悪化した場合にはリンクの色を変えて表示することで、変化が直感的に把握できる。

3.3.3 警告一覧表

管理者が状況の変化に気づけるように、注意すべき状況を自動判別して警告として表示する。異常とみなす閾値を超えた項目を、該当するノードIDや受信時刻とともに警告という形で表示する。警告の対象となる項目は、3.3.2項

シンクノードに届いた、ノード16の全生成データ情報

recvtime	seq	dest	データ送信回数	ACK送信回数	制御送信回数	データ受信回数	ACK受信回数	データ送信時間(μs)	ACK受信時間(μs)	データ受信時間(μs)	ACK受信時間(μs)	電力消費量[mJ]	ロス率	送信までにかかる時間(μs)		
08分25秒00	0	1	0	0	0	0	0	64	0	0	0	82分17秒	59分28秒	2690.13	0%	3秒
13分43秒15	1	1	1	0	0	1	0	65	0	1	13分11秒	59分21秒	2690.02	0%	28秒	
18分22秒01	2	1	2	0	0	2	0	66	0	2	18分17秒	59分22秒	1259409	0%	3秒	
23分08秒37	3	1	3	0	0	3	0	67	0	3	23分17秒	59分22秒	1759289	0%	19分49秒	
28分27秒15	4	1	4	0	0	4	0	115	0	4	28分17秒	59分22秒	1852089	0%	10秒	
33分46秒07	5	1	30	25	25	30	22	164	25	30	33分17秒	149分32秒	1959289	0%	29秒	
38分27秒58	6	1	50	46	46	50	30	195	46	48	38分17秒	175分29秒	2059489	0%	10秒	
43分46秒59	7	1	72	63	63	70	37	193	63	70	43分17秒	175分29秒	2059489	2.2%	29秒	
48分07秒00	8	1	90	84	84	88	45	199	84	88	48分17秒	185分39秒	2059589	2.2%	30秒	

図 4 データ一覧表

警告表

受信時刻	生成時刻	送信ノード	親ノード	ステータス	補足
08分25秒700	08分5秒	32	1	長時間データ未到着	51分38秒 インターバル
10分29秒071	9分29秒	20	47	リンク不安定	20% ロス率
14分26秒909	12分50秒	41	7	リンク不安定	12.5% ロス率
26分28秒914	25分18秒	25	1	リンク不安定	11.1% ロス率
29分08秒517	24分29秒	20	13	リンク不安定	11.5% ロス率
30分26秒334	30分18秒	25	1	リンク不安定	10% ロス率
35分06秒376	34分14秒	29	25	リンク不安定	10% ロス率
45分06秒654	44分29秒	20	13	電池残量	26968.6[J] 電池残量

図 5 警告一覧表

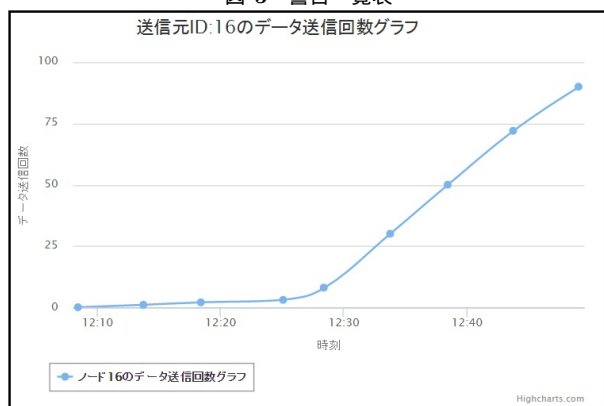


図 6 折れ線グラフ

の「リンクの状態」で述べた項目と同じ (r)(s)(t)(u) である。警告一覧表の例を図 5 に示す。

3.3.4 データ一覧表

シンクノードに到達したパケットを一覧表示することで、管理者は得られた個々の値を確認することができる。経路木表示領域のノードをクリックすることで、ノード毎のデータ表示が可能である。データ一覧表の例を図 4 に示す。

3.3.5 折れ線グラフ

データ一覧に表示された各項目を折れ線グラフで表示することにより、データ量が膨大である場合でも値の傾向を掴みやすくなり、時間とともに移り変わる微妙な変化を捉えやすくなる。データ一覧表の項目毎に用意されたボタンを押下することで、別のウィンドウに折れ線グラフが表示される。折れ線グラフの例を図 6 に示す。

4. 評価

4.1 実装

提案システムは、サーバ上で動作する可視化アプリケーションとして実装した。Webサーバとして Apache ver.2.4.6[12] を用い、アプリケーション内では JavaScript ラ

表 1 WSN の環境

実験開始時刻	12時 00分
実験期間	60分
配置領域	200m × 200m
ノード数	50個
通信可能範囲	半径 50m の円内
制御メッセージ間隔	40秒
センシングデータ生成間隔	5分
ノードの総電力容量	27000[J]

表 2 警告とみなす閾値

電力残量	26970[J]
パケットロス率	10%
前回の受信からのインターバル	30分

イブラリである JavaScript ライブラリである highcharts[13] と vis.js[14] をそれぞれ用いて折れ線グラフとネットワークを表示した。

4.2 評価方法

提案システムに必要な情報収集機能を付与した WSN をシミュレーションにより動作させ、シンクノードに収集されたデータを提案手法に入力することで、提案手法の有効性を評価した。WSN の MAC 及び経路制御プロトコルとしては、受信ノード主導型 MAC プロトコル RI-MAC を拡張し、消費電力をより低減するために我々の研究グループで提案しているプロトコルを用いた [6]。この手法をセンサ端末向け OS である Contiki[15] に実装し、付属のシミュレータ Cooja によりシミュレーションを行った。200[m] × 200[m] の正方形領域に 50 ノードをランダム配置し、60 分間動作をさせた。また、シミュレーション時間中に、故意にノードを 1 つ故障させた。シンクは右辺中央に 1 つ設置し、シミュレーション時間中に収集されたデータを提案システムに入力した。その他のシミュレーションパラメータを図 1 に示す。また、提案手法においては、3.2 節で述べたデータ項目の閾値を表 2 の通りに設定した。これらの値はシミュレーションの結果から、妥当な値を決定したものである。

評価実験は、以下の方法で実施した。まず、上記方法に従ってシミュレーションデータを提案システムに入力する。次に、被験者 1 名に提案システムを利用してもらい、その中で気づいた管理上必要な事項を列挙してもらう。最後に、列挙された内容を精査し、WSN の管理に必要な内容がどれだけ列挙されたかを評価する。ここで WSN の管理に必要な内容とは、2.1 節で述べた、構成管理、障害管理、および性能管理に関する項目のことを指す。つまり、ネットワーク構成や経路の変化、ネットワーク内のノード障害、ネットワーク性能の低下等を検出する。

4.3 評価結果

被験者が提案システムを利用して確認できた事象を確認することで、提案システムの有効性を評価する。

(a) ノード障害 (障害管理): 被験者は、シミュレーション途中で発生させたノード 32 の障害を指摘できた。これは、警告表と経路木表示領域により発見可能である。ある送信元から長時間パケットが届かなかった場合には、警告表に表示されると同時に、経路木発表領域にグレーの色でハイライトされる。これにより、ノード 32 に障害が発生し、不通状態であることがわかる。上記のように、障害 (WSN の場合、障害はノードに限られる) は容易に発見することが可能であり、障害管理が可能であることが示された。

(b) 通信経路の変化 (構成管理): 被験者は、シミュレーション中に経路がどのように変化したかを説明できた。これは、経路木表示領域を用いることで可能である。図 3 のように、経路木表示領域には、現在・過去・未来の経路木を重ねて表示することができる。また、時刻を指定して経路木の状態を表示することも可能であり、時系列を追って経路木の変化を追従できるシステムになっている。本事例では、被験者は、ノード 32 の障害が発生した後、ノード 32 を次ホップとしていた周囲のノードが一斉に経路変更をする様子を把握できた。また、本シミュレーションに用いた MAC・経路制御プロトコル [6] は、経路変更の際に、通常よりも数多くの制御メッセージを交換する。これに対応して、被験者がノード 20 の制御メッセージ受信回数を閲覧したところ、障害発生後に通常よりも多くの制御メッセージが観測されており、この事実も、ノード 32 の障害により周囲のノードが通信経路を変更したことを支持する。以上より、障害に伴う通信経路の変化を詳細に把握することが可能であり、構成管理が可能であることが示された。

(c) 通信性能の低下 (性能管理): 被験者は、ノード 20 と 41 がパケットを送信する際の転送成功率が低いことを指摘した。これは、警告一覧表と経路木表示領域により認識された。パケットロス率が高ければ警告一覧表に表示され、同時に経路木中の対応リンクの色が赤色に変化する。被験者がノード 20 の周囲の状況を調べると、先述の通り、ノード 20 の周囲のノードの制御メッセージ送信量が増大しており、これにより衝突が発生した結果、パケットロス率が増大したと推定された。ノード 41 は、何らかの原因で一時的に次ホップノードが受信を受け付けられない状態であった期間が比較的長時間続いており (文献 [6] の方法では、経路木の葉ノードは、消費電力削減のため、受信を受け付けられない)、これはプロトコルの設計ミス、またはプログラムのバグであると考えられる。

上記のように、被験者は障害管理、構成管理、性能管理のすべてにおいて、WSN の状態を正確に読み取ることができた。また、本評価で用いたシナリオを著者らが詳細に

調査したところ、これ以外の問題は発見されなかった。以上より、提案システムが WSN の管理において有用であることが示された。

5. おわりに

本研究では、低消費電力 MAC プロトコルを用いた WSN を適切に管理するための WSN 管理システムを提案した。提案システムは、障害管理、構成管理、性能管理の 3 つの管理要素を考慮して設計された。また、WSN シミュレーション結果を用いて提案システムの評価を行い、これらの 3 要素の全てを適切に実行できたことを確認した。これにより、提案システムの有用性が示された。

今後は、シミュレーションではなく実環境で実行した WSN のデータを用いた評価を行うことが課題の一つである。

謝辞

本研究は JSPS 科研費 15H02691 の助成を受けたものである。ここに示して謝意を表す。

参考文献

- [1] IEEE802.15.4, <http://www.ieee802.org/15/pub/TG4.html>
- [2] J. Polastre, J. Hill, and D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks," In Proc. of SenSys'04, pp.95-107, 2004.
- [3] M. Buettner, G. Yee, E. Anderson, and R. Han, "X-MAC: A Short Preamble MAC Protocol for Duty-cycled Wireless Sensor Networks," In Proc. of SenSys'06, 2006.
- [4] Y. Sum, O. Gurewits, and D. B. Johnson, "RI-MAC: A Receiver-initiated Asynchronous Duty Cycle MAC Protocol for Dynamic Traffic Loads in Wireless Sensor Networks," In Proc. of SenSys'08, pp.1-14, 2008.
- [5] P. Huang, C. Wang, L. Xiao, "RC-MAC: A Receiver-Centric MAC Protocol for Event-Driven Wireless Sensor Networks," In Proc. of IWQoS'10, 2010.
- [6] 小島祥平, 吉廣卓哉, 受信ノード主導型 MAC プロトコルに基づいた管理コストが低い無線センサネットワーク, 情報処理学会論文誌, 57(2), 2016.
- [7] SNMP Standard, <http://www.snmp.com/>
- [8] N.Ramanathan, K.Chang, R.Kapur, L.Girod, E.Kohler, D.Estrin "Sympathy for the Sensor Network Debugger," In Proc. Sensys'05, 2005.
- [9] K. Liu, M. Li, and Y. Liu, "Passive Diagnosis for Wireless Sensor Networks," IEEE/ACM Transactions on Networking, Vol.18, Issue 4, pp.1132-1144, 2010.
- [10] B.Parbat, A.K.Dwivedi, O.P.Vyas, "Data Visualization Tools for WSNs: A Glimpse," International Journal of Computer Applications, Vol.2, No.1, pp.14-20, 2010.
- [11] 日本オープンソースソフトウェア (OSS) 推進フォーラム "ネットワーク運用管理に求められる作業" <http://ossforum.jp/node/594>
- [12] "The Apache Software Foundation," <http://www.apache.org/>
- [13] "Highcharts," <http://www.highcharts.com/>
- [14] "vis.js," <http://visjs.org/>
- [15] Contiki, <http://www.contiki-os.org/> (accessed at 1st Feb 2016).