

無線マルチホップネットワークの故障中継無線ノードを検出する協調ウォッチドッグ手法

岡瀬 力[†] 梶垣 博章[†]

東京電機大学大学院 未来科学研究科 ロボット・メカトロニクス学専攻[†]

1 はじめに

無線マルチホップネットワークでは、送信元ノードから送信先ノードまで、中継ノードを介してデータメッセージを配送する。このとき、中継ノードが故障することで、データの配送が不可能となる。本論文では、ウォッチドッグ手法を拡張した協調ウォッチドッグ手法を用いて故障ノードを検出する。この手法では、中継ノードとデータ配送に参加しない監視ノードが中継ノードの送信データを傍受することで故障ノードを検出する。この協調により故障ノードの誤検出をも検出することを可能にする。そして、故障通知メッセージを送信元ノード配送し、故障ノードを除外した新しい配送経路を構築する。さらに、監視ノードの一部を接続ノードで置き換えることによって経路検出率を向上する拡張協調ウォッチドッグ手法を提案する。

2 関連研究

故障中継無線ノードを検出するための手法としてウォッチドッグ手法がある [1]。ウォッチドッグ手法では中継ノードが次ホップノードが送信するデータを傍受することを活用して、監視し、その故障を検出する。故障ノードを検出した場合には、故障通知メッセージ $Fnot$ を送信元ノードに配送する。図 1 に示すように、ノード N_i は次ホップノード N_{i+1} が N_{i+2} に送信するデータメッセージ m' を傍受する。 N_i は自身が送信したデータメッセージ m と m' を比較することで N_{i+1} の故障を検出する。また、 N_i は N_{i+1} が一定時間内に m を転送しないことによっても N_{i+1} の故障を検出する。 N_{i+1} の故障を検出した N_i は $Fnot(N_{i+1})$ を N_{i-1} へ送信する。しかし、この手法では $Fnot(N_{i+1})$ の誤送信を検出

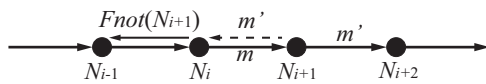


図 1: ウォッチドッグ手法。

することができない問題がある。図 2 では、 N_{i+1} は、 N_i から受信した m を N_{i+2} に送信している。しかし、故障した N_i が誤って $Fnot(N_{i+1})$ を N_{i-1} に送信しても、この誤りを N_{i-1} が検出することができない。この場合、 N_{i-1} は N_i から受信した $Mdet(N_{i+1})$ が正し

いか確かめることができない。その結果、送信元まで $Mdet(N_{i+1})$ が送信されるため、 N_i の故障動作を検出することができない。

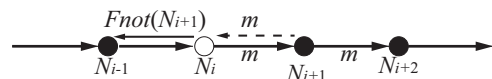


図 2: 誤った故障動作通知検出不可。

3 提案手法

3.1 協調ウォッチドッグ手法

故障中継ノードが前ホップ中継ノードから受信したメッセージとは異なるメッセージを次ホップ中継ノードに送信したことを検出することに加えて、故障ノードが誤った故障通知メッセージを送信したことを検出することを可能にする協調ウォッチドッグ手法を提案する。協調ウォッチドッグ手法では、各中継ノード N_i に対して監視ノード O_i を定める。このとき、 O_i は N_{i-1} 、 N_i 、 N_{i+1} と隣接することが必要である (図 3)。

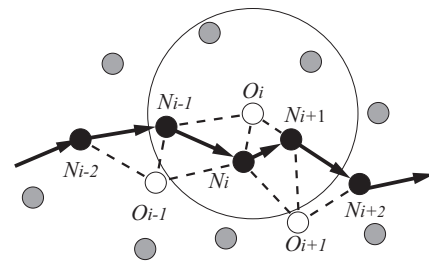


図 3: 協調ウォッチドッグ手法。

N_i と O_i は、 N_{i+1} のメッセージ転送を監視する。 N_i と O_i は、 N_{i+1} が N_{i+2} へ送信したメッセージを傍受することができる。そのため、 N_i は自身が N_{i+1} へ送信したメッセージと比較することによって、 O_i は自身が傍受した N_i が N_{i+1} へ送信したメッセージと比較することによって、 N_{i+1} が N_i から受信したものと異なるメッセージを N_{i+2} へ送信したことを検出する。 N_i と O_i は、 N_{i+1} の故障通知メッセージ $Fnot(N_{i+1})$ を N_{i-1} へ送信する。

$Fnot(N_{i+1})$ の N_0 への無線マルチホップ配送は、次のように実現される。 N_j ($0 < j < i$) は、 N_{j+1} と O_{j+1} から $Fnot(N_{i+1})$ を受信する。 $j = i - 1$ のときには、これらの受信順序は定まらないが、後述する誤故障通知メッセージの転送を回避するため、後から受信した

Cooperative Watch-Dog for Failure Node Detection in Wireless Multihop Network

[†]Riki Okase [†]Hiroaki Higaki

[†]Department of Robotics and Mechatronics, Tokyo Denki University

$Fnot$ を N_{j-1} へ送信する。 $O_j (0 < j < i)$ は、 N_{j+1} が N_j へ送信した $Fnot(N_{i+1})$ と N_j が N_{j-1} へ送信した $Fnot(N_{i+1})$ とをこの順序で受信する。ここでは、 O_j は同一の $Fnot$ を受信するが、後述する誤故障通知メッセージの転送を回避するため、後から受信した $Fnot$ を N_{j-1} へ送信する。

次に、 N_{i+1} が故障していないにもかかわらず N_i が $Fnot(N_{i+1})$ を N_{i-1} へ送信する場合を考える。このとき、 O_i は N_i が N_{i-1} へ送信した $Fnot(N_{i+1})$ を傍受するが、監視している N_{i+1} が故障していないことも分かる。すなわち、 O_i は N_i が故障していることを検出する。そこで、 O_i は $Fnot(N_i)$ を N_{i-1} へ送信する。この結果、 N_{i-1} は、 N_i が送信した $Fnot(N_{i+1})$ と O_i が送信した $Fnot(N_i)$ とをこの順序に受信する。 N_{i-1} は、後から受信した $Fnot$ を N_{i-2} へ送信することから、 $Fnot(N_i)$ を送信する。一方、 O_{i-1} は、 N_i が送信した $Fnot(N_{i+1})$ と N_{i-1} が送信した $Fnot(N_i)$ とをこの順序に受信する。 O_{i-1} は、後から受信した $Fnot$ を N_{i-2} へ送信することから、 $Fnot(N_i)$ を送信する。 N_j と $O_j (0 < j < i-1)$ は、 N_{j+1} と O_{j+1} から $Fnot(N_i)$ を受信することから、 $Fnot(N_i)$ が N_0 に無線マルチホップ配送される。 N_{i+1} が故障していないにもかかわらず O_i が $Fnot(N_{i+1})$ を N_{i-1} へ送信する場合も同様に、 N_i が O_i の故障を検出し、 $Fnot(O_i)$ が N_0 に無線マルチホップ配送される。

最後に、 N_{i+1} から $Fnot(N_k) (k > i+1)$ を受信していないにもかかわらず N_i が $Fnot(N_k)$ を N_{i-1} へ送信する場合を考える。このとき、 O_i は N_i が N_{i-1} へ送信した $Fnot(N_k)$ を傍受するが、それ以前に N_{i+1} が N_i に送信した $Fnot(N_k)$ を傍受していないことから、 N_i が故障していることを検出する。そこで、 O_i は $Fnot(N_i)$ を N_{i-1} へ送信する。この結果、 N_{i-1} は、 N_i が送信した $Fnot(N_k)$ と O_i が送信した $Fnot(N_i)$ とをこの順序に受信する。 N_{i-1} は、後から受信した $Fnot$ を N_{i-2} へ送信することから、 $Fnot(N_i)$ を送信する。一方、 O_{i-1} は、 N_i が送信した $Fnot(N_k)$ と N_{i-1} が送信した $Fnot(N_i)$ とをこの順序に受信する。 O_{i-1} は、後から受信した $Fnot$ を N_{i-2} へ送信することから、 $Fnot(N_i)$ を送信する。 N_j と $O_j (0 < j < i-1)$ は、 N_{j+1} と O_{j+1} から $Fnot(N_i)$ を受信することから、 $Fnot(N_i)$ が N_0 に無線マルチホップ配送される。 N_{i+1} が $Fnot(N_k)$ を送信していないにもかかわらず O_i が $Fnot(N_k)$ を N_{i-1} へ送信する場合も同様に、 N_i が O_i の故障を検出し、 $Fnot(O_i)$ が N_0 に無線マルチホップ配送される。

この協調ウォッチドッグ手法では、ノードの故障が検出されない場合には追加の制御メッセージ交換を必要としない。

3.2 拡張協調ウォッチドッグ手法

前節で提案した協調ウォッチドッグ手法では、各中継ノード N_i について、 N_{i-1} 、 N_i 、 N_{i+1} と隣接する監視ノード O_i が必要である。この制約条件を緩和するために、図4に示す拡張協調ウォッチドッグ手法を提案する。ここでは、 N_{i-1} 、 N_i 、 N_{i+1} と隣接する監視ノード O_i に換えて、 O_{i-1} 、 N_i 、 O_{i+1} と隣接する接続ノード P_i を導入する(図4)。

O_i が担っていた N_{i+1} によるメッセージ転送の監視は、 N_{i+1} に委譲される。 N_i と O_{i+1} は、 N_{i+1} が N_{i+2} へ送信したメッセージを傍受することができる。その

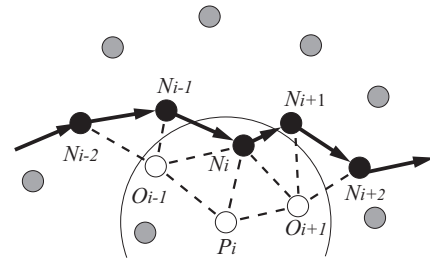


図4: 拡張協調ウォッチドッグ手法。

ため、 N_i は自身が N_{i+1} へ送信したメッセージと比較することによって、 O_{i+1} は自身が傍受した N_i が N_{i+1} へ送信したメッセージと比較することによって、 N_{i+1} が N_i から受信したものと異なるメッセージを N_{i+2} へ送信したことを検出する。 N_i と O_{i+1} は、 $Fnot(N_{i+1})$ を N_{i-1} と P_i へそれぞれ送信する。 P_i は、 O_{i+1} から $Fnot(N_{i+1})$ を受信するとともに N_i が N_{i-1} へ送信した $Fnot(N_{i+1})$ を傍受する。これらの受信順序は定まらないが、後述する誤故障通知メッセージの転送を回避するため、後から受信した $Fnot$ を O_{i-1} へ送信する。 O_{i-1} は、 N_i から N_{i-1} へ送信された $Fnot(N_{i+1})$ を傍受するとともに、 P_i が O_{i-1} へ送信した $Fnot(N_{i+1})$ を受信する。これらはこの順序で受信されるが、後述する誤故障通知メッセージの転送を回避するため、後から受信した $Fnot$ を N_{i-2} へ送信する。 $Fnot(N_{i+1})$ の N_0 への無線マルチホップ配送は、協調ウォッチドッグ手法と同様の方法で行なわれる。

なお、 N_{i+1} から $Fnot(N_k) (k \geq i+1)$ を受信していないにもかかわらず N_i もしくは O_{i+1} が $Fnot(N_k)$ を N_{i-1} もしくは N_i へ送信する場合についても、協調ウォッチドッグ手法と同様にして $Fnot(N_i)$ もしくは $Fnot(O_{i+1})$ を N_0 に配送することができる。

このように、 O_i ではなく P_i が N_i の隣接ノードである場合でも、 O_i である場合と同様に故障ノードを検出し、 $Fnot$ を N_0 に無線マルチホップ配送することができる。ここでは、 P_i は O_{i-1} 、 O_{i+1} と隣接するとしたが、 O_{i-1} ではなく N_{i-1} 、 O_{i+1} ではなく N_{i+1} と隣接するのであっても同様の故障ノードの検出と $Fnot$ の N_0 への無線マルチホップ配送を実現することができる。すなわち、 P_i は、 O_{i-1} または N_{i-1} 、 N_i 、 O_{i+1} または N_{i+1} という3つのノードと隣接することが求められる。

4 まとめ

本論文では、ウォッチドッグ手法を拡張し、誤データ送信に加えて誤故障通知送信を検出する協調ウォッチドッグ手法を提案した。さらに、協調ウォッチドッグ手法で必要となる配送経路の各中継ノードに対応する監視ノードの一部を接続ノードで置き換える拡張協調ウォッチドッグ手法を提案した。それぞれの手法による配送経路検出率を評価することが今後の課題である。

参考文献

- [1] Martl, S., Giuli, T. J., Lai, K. and Baker, M., "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, pp. 255-265 (2000).