

天井照明の光度制御と無線センサノードを用いた通信手法の検討

村上 広記

Hiroki MURAKAMI

1 はじめに

無線センサネットワークは、無線装置を内蔵した多数のセンサが相互に連携することで、実空間の情報の収集を可能とする技術である¹⁾。無線センサネットワークを構成する全てのノードにデータを送信するためには、一般的にフラッシングを用いる方法がある。しかしながら、フラッシングは無線センサノードの限られた電力資源を消費するとともに転送遅延により同時にデータを取得することはできない。また、照明の光度制御を用いるデータ通信手法としては可視光通信システム²⁾があるが、高速通信を行うために専用の照明設備と受光端末が必要となる。そのため本研究では、調光可能な照明器具および一般的な照度センサのみを用いてデータ通信を実現する。また、提案手法における実用性を向上させるため、通信速度の高速化方法について検証する。天井照明の光度制御と無線センサノードを用いた通信手法を Data Transmission based Lighting Control とし、以後 DTLC と表記する。

2 Data Transmission based Lighting Control

2.1 Data Transmission based Lighting Control 概要

DTLC は、照明の明るさ（光度）を変化させることで、照度センサが感知する明るさ（照度）を変化させ、照度変化量からデータ通信を実現する。まず本手法のアルゴリズムを提案するために、光度変化時の照度変化の推移を調査する。

2.2 無線センサノードを用いた照度推移実験

現在照度に変化を与えた場合に、無線センサノードに搭載された照度センサが得る照度値がどのように推移するかを検証する。本実験では、無線センサノードとして Crossbow 社の MOTE MICAz を使用した。MOTE MICAz に汎用外部センサ基盤である MDA088 を設置し、リードタイプの NaPiCa 照度センサを組み込むことで、照度値を取得する。本実験は同志社大学香知館の知的システム創造環境実験室にて行い、シャープ製フルカラー LED28 灯と NaPiCa 照度センサを搭載した無線センサノード 1 台、シンクノード 1 台を用いた。照度センサの照度取得間隔を 100 ミリ秒に設定した。実験環境の俯瞰図を Fig. 1 に示す。なお、照明と無線センサノードの距離は照明の垂直直下に無線センサノードを置いた場合、1.9 m である。無線センサノードは、Fig. 1 に示す通り照明の直下に配置した。

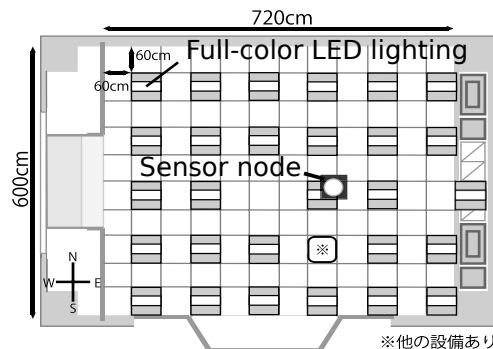


Fig.1 照度取得実験環境

机上面照度が 500 lx となる環境下で、照度を 500 lx から 15 lx 程度上昇させ、その 1 秒後に元の照度値に戻した場合の照度の推移を取得した。なお、外光など照明以外の光が照度センサに影響を与えない環境下で行った。Fig. 2 に照度取得間隔が 100 ミリ秒の場合のある時間における照度推移の履歴を示す。

Fig. 2 からわかる通り、光度変化を感知したとき、照度が急速に変化していることがわかる。この実験結果から、天井照明の光度制御を用いたデータ通信アルゴリズムを提案する。なお、本手法では外光などの照明以外の光が照度センサに影響を与えない環境を想定する。

3 DTLC アルゴリズム

3.1 DTLC Basic Algorithm

Fig. 2 から、光度変化を感知し照度が急速に変化するとともに、同じ光度で照明が点灯している場合にも、照度値に誤差が発生し、照度値に揺らぎが発生している。したがって、光度変化した際のみの照度変化をセンサが正しく光度変化時の照度変化であると感知する必要がある。そこで DTLC-BA では取得した照度値を前回取得した照度値を用いて微分し、その傾きの大きさを光度変化時の照度変化か否かを判断する。

次に、具体的なアルゴリズムを述べる。データ送信間隔はセンサノードが既知であるという条件で、アルゴリズムを提案する。なお、データ送信間隔を T 秒とする。照明側では送信したいデータを 2 進数のビット列に変換し、1 ビットずつ送信する。本手法では、傾きが閾値未満であれば 0、閾値以上であれば 1 とする。また、連続して同じビットを送信する場合は同じ光度で照明が点灯するため、照度値は変化しない。したがって、開始ビット受信時、あるいは光度が変化した T 秒後に現在照度値を用いて 1 か 0 かを判断する。

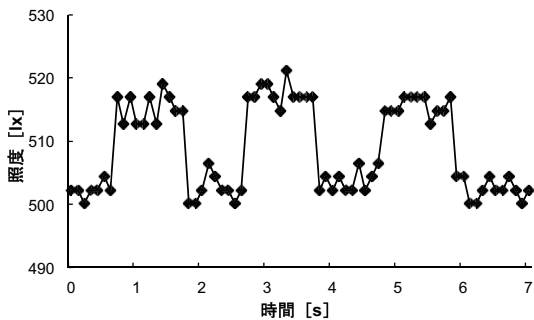


Fig.2 100 ms 間隔で取得した照度履歴

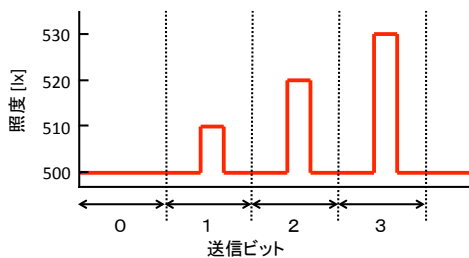


Fig.3 階調変化例

- (1) 照度センサが現在照度を取得する
- (2) 照明が現在光度から $x\%$ (ただし, $x < 7\%$) だけ光度値を上昇させる
- (3) 照度センサが照度変化を感知した場合, 照度変化前後の傾きを計算する
- (4) 傾きが閾値以上であれば, データ通信を開始する
- (5) 照明が光度値を光度変化前に戻す
- (6) 照度取得間隔毎に傾きを計算し, その値が閾値以上なら 1 とする
- (7) 開始ビット受信時から T 秒間傾きが閾値未満の場合, 現在照度が照度変化後の照度値域であれば 1, 変化前の照度値域であれば 0 とする

3.2 DTLC Multi-phased Threshold Algorithm

DTLC-MTA は照明側で送信したい 2 進数のビット列を 2 bit ずつ 4 進数のビット列に変換し, 1 bit ずつ送信する。照明は, 照度を Fig. 3 のように 4 段階に階調変化させることでビットを送信する。照明は送信間隔 T 秒間の間に, 初期光度を変化させ, 再び初期光度に戻す。センサノードは取得間隔毎に照度の傾きを計算し, その傾きを 3 つの閾値によって 4 段階に判別する。Fig. 3 の場合, 500 lx から T 秒間照度変化がないときは傾き 0, 500 lx から 510 lx に変化したときは傾き 5 で 1 のように, 傾きを計算し 3 まで受信する。しかし, Fig. 2 から分かる様に実際の照度値には揺らぎがあるため, 閾値を設定することで誤感知を防ぐ必要がある。

4 DTLC を用いたデータ通信実験

4.1 DTLC を用いたデータ通信実験概要

前章で示したアルゴリズムを無線センサノードに実装し, データ通信時における通信速度と通信誤りの検証実験を行った。前章の実験と同様の実験環境および使用機器を用いた。また, 外光が入らないよう窓際には白色のパーティションを設置した。なお, 本実験では, あらかじめランダムに生成した 2 進数 1000 bit のビット列を用いた。

4.2 DTLC Basic Algorithm を用いた実験

送信間隔 1 秒, 0.5 秒で実験を行った結果, 誤り率は 0 % となり誤りなく全てのビット列を送信することができた。したがって, さらに送信間隔を短くすることで通信速度を向上できることがわかった。

4.3 DTLC Multi-phased Threshold Algorithm を用いた実験

本実験では, 用意した 2 進数 1000bit のビット列を 2 bit ずつ 4 進数に変換を行い送信間隔 1 秒で通信を行った。そのため, 変換後のビット長は 500 bit となった。送信間隔は 1.0 秒とした。

実験の結果, 受信したビット列は 7 箇所誤りがあった。1 箇所の誤りは, 正しい 1 bit の代わりに誤った 1 bit が挿入された。残り 6 箇所の誤りでは, 正しい 1 bit ではなく誤った 2 bit が挿入されていた。

5 DTLC Algorithm の考察と今後の展望

5.1 DTLC Basic Algorithm の考察

DTLC-BA では送信間隔を 1 秒から 0.5 秒に変更することで 2 倍速く通信できている。また, 誤りなく送信できているため, 送信間隔 0.5 秒までは高速化可能であると考えられる。また, 送信間隔をさらに短くし, 送信間隔を NaPiCa 照度センサの最小照度取得間隔である 8.5 ミリ秒まで短縮可能と考えられる。

5.2 DTLC Multi-phased Threshold Algorithm の考察

1 bit の誤りは, 0 を受信する際に, 変化前照度の揺らぎを傾きがあるものとして誤感知したため 0 を 1 と判定していた。これは最適な閾値を設定することや, 照度の変化幅を大きくすることで解消できると考えられる。

6 箇所の 2 bit の誤りビットが挿入された原因は傾きの誤認識と推測する。光度変化が照度取得点に跨っており, 2 つの傾きを得たために, 2 つの受信ビットと判定したことが原因だと考えられる。この解決方法として複数ステップにおける傾きを計算することで対策できると考えられる。

参考文献

- 1) YICK, J.: Wireless sensor network survey, Comput. Netw., vol.52, pp.2292-2330(2008)
- 2) 春山真一郎: 可視光通信, 電子情報通信学会論文誌. A, 基礎・境界, Vol.86, pp.1284-1291(2003)