

PSOを用いた無線センサネットワークにおける シンクノードの配置手法の検討

M2015SC011 三福健太

指導教員 : 河野浩之

1 はじめに

WSN(Wireless Sensor Networks)は大規模数のセンサノードを利用することで広範囲をセンシングすることが可能である.WSNは農園や工場などの環境の監視や災害救助支援など、多種多様なシステムへの利用が期待されている.このWSNを実現するための課題としてシンクノードの最適配置問題がある.WSNでは大規模数のセンサノードを利用してセンシングを実行し、各々のセンサノードは自分が所得したデータをシンクノードに向けて送信する.シンクノード付近に配置されたセンサノードは転送回数が過多になり、通信負が集中してしまう.このため、シンクノードを適切な位置に配置すればネットワーク全体の長期的運用が可能になる[1][4].

本研究では群知能の一種である粒子群最適化(PSO:Particle Swarm Optimization)を利用することにより、シンクノード配置の最適化を実行することでネットワーク内のセンサ全体の電池寿命を可能な限り長くすることを目的とする.本研究におけるネットワーク寿命の定義は無線端末が1つでも通信不可能となった場合をネットワーク寿命とする.また、解が早期に収束することを避けるためにPSOアルゴリズムの改良を提案する.

第2章ではPSOの基本動作と既存の研究を紹介する.第3章では粒子群最適化を改良したものを解説する.第4章ではネットワークシミュレータであるns2を使用し、提案したアルゴリズムを使用してシンクノードの最適配置を実行する.また、ベンチマーク関数を利用してアルゴリズムの性能を評価する.第5章では提案したPSOの性能を評価する.

2 Particle Swarm Optimization(PSO)

2.1節ではPSOの概略と粒子の速度と位置を求める計算式、また粒子の振る舞いについて述べ、2.2節ではPSOの課題点について説明する.

2.1 PSOの概要

PSOは1995年にJ.KennedyとR.Eberhartによって提案されたアルゴリズムである[3].PSOは、解空間に多数存在する粒子が解情報を保持しながら、解空間を飛び回ることにより目的関数の解の探索を行う手法である.

鳥や魚の群れの振る舞いを解探索に応用したものであり、各個体を粒子に見立て、目的関数のより良い評価値が得られる位置を探索する.このとき粒子は他の粒子と影響を及ぼし合うため、個々は非常に簡単なルールにより動作しているが、群れ全体の振る舞いは非常に複雑になる.

PSOは、各粒子が並列動作できるので、効率的に解探索を行える.また、PSOにおける群れの中の粒子は位置と速度の情報を各々持っている.

各粒子の速度の更新を式(1)で位置の更新を式(2)に示す.式(1)の右辺第一項は、慣性と呼ばれ、今までの速度と同じ方向へエージェントが継続して移動する様子をモデル化している.第二項は自己認識と呼ばれ、粒子自身の最良の位置であるpbestに向かう乱数係数で変化する線形の誘引力である.第三項は、社会認識と呼ばれ、すべての粒子によって発見された最良の位置であるgbestに向かう線形の誘引力であり、これも乱数の係数により変化する.

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (pbest_i^t - x_i^t) + c_2 r_2 (gbest^t - x_i^t) \quad (1)$$
$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (2)$$

各々の粒子は自己が過去に得た目的関数の評価値が最高となる位置情報 $pbest_i^t$ を保持している.これとは別に各々の粒子は解空間において目的関数の評価値が最高となる位置情報 $gbest^t$ を全体で共有する. x_i^t は粒子 i の時間 t における位置ベクトルを示し、 v_i^t は粒子 i の時間 t における速度ベクトルを示す.また、 ω は慣性係数、 c_1 、 c_2 は加速度係数、 r_1 、 r_2 は一様乱数を意味する.PSOにおける粒子の探索位置決定の概念図を図1に示す.図中の x_i^{t-1} は一世代前の粒子の位置、 x_i^t は現在の位置、 x_i^{t+1} は一世代後の位置である.

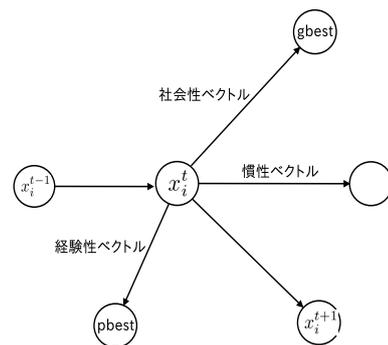


図1 PSOにおける粒子の探索位置決定の概念

2.2 PSOの課題点

PSOは局所解を多数持つ大域的最適化問題に対する効率的なメタヒューリスティクス解法として知られているが、その収束の早さから、広範囲で解探索に利用する場合には、局所解への落ち込みが発生し、十分な解探索が行えないことがある.この問題は、パラメータ設定や、群

れのネットワーク構造と関連しており、群れの振る舞いを制御できないことが原因であると考えられる。

3 広範囲の探索を考慮した PSO

3.1 節では密集の回避方法と移動モデルについて、3.2 節では情報伝達モデルについて述べ、3.3 節で提案手法の振る舞いを説明する。3.4 節ではベンチマーク関数を使用し、提案手法と既存手法の性能評価をする。

3.1 密集回避と移動モデル

PSO では各粒子が影響しあうため、複数の粒子が密集してしまうことが多々ある。このような状況に陥った場合、同範囲を探索してしまい探索効率の低下が予想される。この状況を打開するためにここでは密集回避行動を追加する。領域を持つ粒子のイメージを図 2 に示す。ここで粒子 i_1 と i_2 が密集した場合を考える。各粒子は自己を中心とする半径 d の円である領域持ち、この領域に他の粒子が t 世代以上存在した場合に密集と判断する。密集と判断した場合に番号の高い順、ここでは粒子 i_2 をランダムウォークする粒子へと t 世代変更した後に、 $pbest$ を初期化させ PSO の振る舞いを実行する。

ランダムウォークを実現するため、RWM(Random Waypoint Model)[2] を利用する。RWM はアドホックネットワークなどのネットワークシミュレーションに利用されており、個々のノードの移動を非常に簡単に記述するモデルである。各ノードは与えられた通過点から次の通過点に向かって無造作に移動を行う。ここで、通過点は与えられた移動可能範囲内に均一に分布しており、ここでの通過点と次の通過点を結び直線を慣用的に”leg”と呼び、ノードは必ず leg 上を移動する。目的通過点に到着してもただちに次の移動を開始する必要はなく、Pause Time を設けることができる。最後に目的通過点に到着したらエリア内に新しい次の通過点を設定し、新たな leg を設ける。

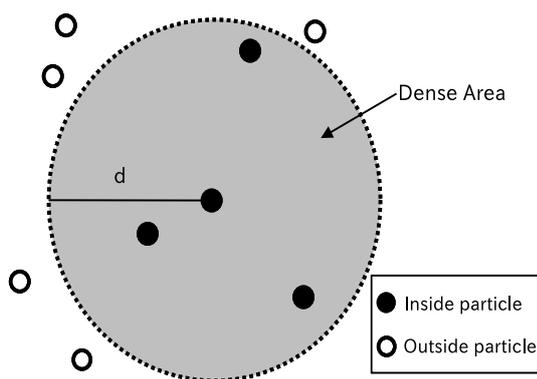


図 2 密集を判断する領域

3.2 情報伝達モデル

PSO では一般的には粒子全体の最良解である $gbest$ の情報を領域内全体で共有するモデルを採用している。しかし、最良解に向かうベクトルを強くすると、収束速度が速くなる一方で大域的探索の可能性は失われてしまう

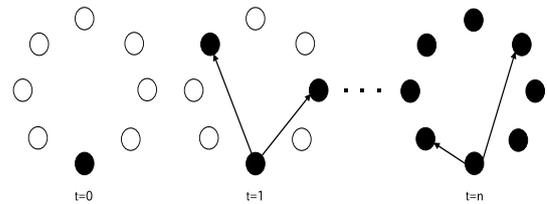


図 3 情報伝達の概略図

ことがある。そのために一度収束した解が局所解であった場合、そこから抜け出すのは困難である。そこで本研究では、最良解の情報伝播経路をメッシュ型のモデルを採用し $gbest$ を使用し、一回で全ての粒子に情報を伝達していたものを複数の期間に分けて伝達し、時間を掛けることで、大域的探索の可能性を大きく広げる。

図 3 に情報伝達の概要を示す。図中の黒丸は情報を得ている粒子であり、白丸は情報をまだ得ていない粒子を意味している。また、図中の左は探索初期である $t = 0$ を表している。探索初期にあたる $t = 0$ において $gbest$ を発見した粒子は $t = 1$ の時に領域内からランダムに選んだ粒子に対し情報伝達を実行する。最終的にはすべての粒子が情報を共有することで、局所解に陥る可能性を排除しながら局所的探索へ移行する。いくつかの粒子の群れが局所解に収束してしまっても、その情報を共有していない群れが存在するため、局所解に捕らわれている群れに先導されて全体が局所解に陥ってしまう危険を回避し、最適解発見の可性を高めることができる。

3.3 解の探索手順

PSO が持つ課題を回避し、探索の後半においても広範囲での最適解の探索をするために、本研究では $pbest$ 、 $gbest$ に影響されず常にランダムウォークする粒子群の追加を提案する。これにより、粒子の過密による探索の停滞を防ぎ、より効率的な探索の実現を可能にする。本研究の改良型の解探索手順を以下に示す。

Step1. 準備

探索回数 T_{max} 、粒子の総数 N 、式 (1) と式 (2) で用いるパラメータ ω 、 c_1 、 c_2 を任意の数に設定。

Step2. 初期化

更新回数カウンタ $k = 0$ とし、粒子それぞれの初期位置ベクトル x_i^0 と初期速度ベクトル v_i^0 をランダムに生成する。初期条件における $pbest_i^0$ と $gbest^0$ を決定する。 $pbest_i^0$ は各粒子の初期位置であるので $pbest_i^0 = x_i^0$ となる。 $gbest^0$ は $pbest_i^0$ の粒子の中の最良解と等しくなるので $ib = \operatorname{argmin}_i (pbest_i^0)$ であるため $gbest^0 = pbest_{ib}^0$ となる。

Step3. 速度と位置の更新

式 (1) を利用し速度ベクトルを、式 (2) を利用し位置ベクトルを更新する。

Step4. 最良解の更新

ここでは Pbest と Gbest の更新し、もし $f(x_i^{k+1}) < f(Pbest_i^k)$ ならば更新式は $Pbest_i^{k+1} = x_i^{k+1}$ となる。
 $ib = \operatorname{argmin}_i f(pbest_{ib}^{k+1})$ であるので $Gbest^{k+1} = Pbest_{ib}^{k+1}$ となる。

Step5. 密集の判断 粒子 P から距離 d 以内に別の粒子が存在する場合、粒子 P をランダムウォークする粒子に変更する。

Step6. 終了判定

$k = T_{max}$ であれば探索を止め値を $f(Gbest^{k+1})$ として終了し、そうでなければ $k = k + 1$ として Step3 へ戻る。

3.4 PSO アルゴリズムの評価

ここでは提案した PSO アルゴリズムの性能評価をベンチマーク関数を利用して調べる。ベンチマーク関数は先に説明した、単峰性、多峰性、関数の不連続などを特徴に持ち、探索の範囲が設定され、最小値、最大値が既に分かっている関数である。最適化アルゴリズムの評価はこれらベンチマーク関数を評価関数として最適解を探させることで性能の良し悪しを比較する。評価には Rastrigin 関数を利用する。

Rastrigin 関数は、式 (4) で表され、最適解の周辺に格子状に最適値に近い値を持つ局所的最適解を持つ多峰性関数である。設計変数間に依存関係はないことが知られている。表 4 は、Rastrigin 関数での結果のグラフである。40 次元では、提案型 PSO が 141 世代あたりで最適解に辿り着いたのに対して、既存型の PSO は 237 世代になっても最適解に到達することができないという結果になった。

$$F_{Rastrigin}(x) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i)) \quad (3)$$

($-5.12 \leq x_i < 5.12$)

表 1 各 PSO の性能評価

| メソッド | 到達回数 | 最良解 | 平均 |
|---------|------|--------|---------|
| 既存型 PSO | 237 | 4.1366 | 27.6896 |
| 改良型 PSO | 141 | 0.3774 | 22.2375 |

4 シンクノードの最適配置

ネットワークシミュレータである ns2 を使用しシンクノードの最適配置を PSO を利用して求める。4.1 節では PSO のパラメータなどの実験環境を、4.2 節ではセンサネットワークへの適応するための準備について、4.3 節ではシミュレーション結果についてシンクノードの配置と寿命について既存の PSO と比較して考察する。

4.1 実験環境

実験環境として、数多くの固定されたセンサノード群で構成される大規模無線センサネットワークを用いて、各センサノードのセンシング情報を定期的にシンクノードで収集する状況を想定する。センサノード群の観測領域内への配置は領域内に一様に分布するものとする。また、実際に無線センサネットワークの利用場面ではベースステーションにてこれらのセンサノード群の位置情報を所得してから、シンクノード群の配置候補が探索されることになっている。PSO の各パラメータは表 2 に示す。 c_1 、 c_2 は進行方向を選択する際に gbest に重みを置くか、あるいは pbest に重みを置くかのパラメータである。 ω は粒子の慣性を決定するパラメータであり、値が大きいほど大域的な動作となる。

表 2 PSO 設定

| パラメータ | 設定値 |
|----------|--------|
| 試行回数 | 200[回] |
| 粒子数 | 50[個] |
| c_1 | 2.0 |
| c_2 | 2.0 |
| ω | 0.8 |

4.2 センサネットワークへの適用

本実験においては観測範囲内のネットワーク全体での総合ホップ数を評価する。センサネットワーク環境を準備するにあたり本研究では network simulator2(ns2) を使用する。ns2 での実験環境は表 3 の通りである。シミュレーション時のノードの配置例を図 4 に示す。横軸は x 座標、縦軸は y 座標であり、黒色の四角形がシンクノード、緑色の円がシンクノードであり、番号は各ノード番号を表している。

全てのセンサノードが最寄りのシンクノードをあて先として非同期にデータパケットが転送された場合において、これらのデータパケットのすべてが目的地のシンクノードに到達するまでの総合転送回数である。総パケット数が少ないほど転送回数が少なくなるため全体の消費電力も低下する。評価関数 (式 4) は以下の通りである。

$$f(x) = \sum_{i=1}^s hops(sn_i) \quad (4)$$

sn_i はシンクノードの配置座標を示し、 s は観測領域に配置するシンクノード数、 $hops(Sn_i)$ はシンクノード i を最も近いシンクノードとする全センサノードから送信されたデータパケットが、任意のシンクノード i に到達するまでの総合ホップ数を表す。各ノードの初期電力量は 40[mAh] であり、送信消費電力を 0.02[mA]、受信消費電力を 0.025[mA] とする。

表 3 シミュレーション環境

| | |
|------------|----------|
| シミュレーション時間 | 180[sec] |
| フィールドサイズ | 1000[m] |
| センサノード数 | 500[個] |
| シンクノード数 | 5[個] |
| 通信半径 | 25[m] |
| シミュレーション時間 | 180[s] |

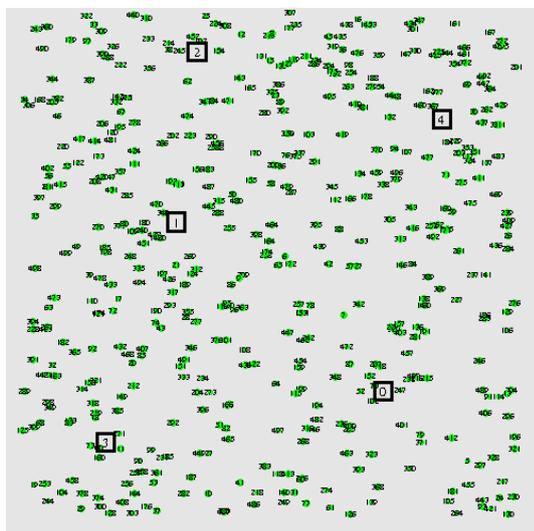


図 4 ノードの配置例

4.3 実験結果と考察

実験は 50 回行い、回毎にセンサノードの位置をランダムに配置した後に提案手法によりシンクノードを配置した。配置後に各センサノードが最寄りのシンクノード一台に向けてデータを送信する。図 5 では既存の PSO と提案した PSO を使用し、シンクノードを配置したときのセンサネットワークの寿命を比較したものである。x 軸は生存時間、y 軸はセンサノード全体の残余電池量の合計を表している 1000 秒毎に各ノードの残余電力量の合計値を算出し、50 回の平均値をグラフにプロットした。

また本実験の寿命は一つ目のセンサノードが電池切れを起こした時間としており、図中の途切れた箇所が寿命である。シミュレーション開始直後に両方に大きな差は無いが時間が経過するにつれて差は広がった。表 4 より提案型の PSO の方が既存の PSO よりネットワークの寿命が 4328 秒長く、つまり既存の手法より 18.8% 寿命が延びた。

表 4 実験結果の比較

| | ネットワーク寿命 | 残余電量 |
|------|----------|-----------|
| 既存手法 | 22495[s] | 5462[mAh] |
| 提案手法 | 26733[s] | 5221[mAh] |

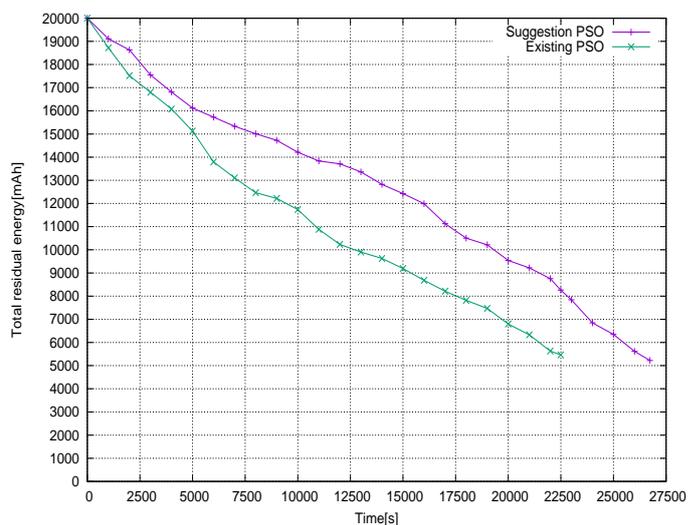


図 5 ネットワーク全体の残余電力量

5 むすび

センサネットワーク環境において PSO を利用しシンクノードを適切な位置に配置することでネットワークの寿命を延ばすことができる。しかし、既存の PSO では粒子同士が共通の解に密集してしまい寿命を延ばす妨げとなった。この解に密集してしまう状況を防ぐために本研究では粒子に密集しているかを判断する領域を持たせ、局所解から脱出する機能を追加した。解から抜け出す粒子を数世代の間にランダムウェイポイントモデルを使用しランダムウォークさせ、探索領域の拡大を促し、ネットワークの寿命を既存手法より 18.8% 延びた。

参考文献

- [1] Heinzelman, W. R., Chandrakasan, A. and Balakrishnan, H.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks, Vol. 33, pp. 8020–8041 (2000).
- [2] Josh, B., David, M., David, J., Yih-Chun, H. and Jorjeta, J.: A Performance Comparison of Multi-hop Wireless Ad Hoc Network Routing Protocols, *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom '98*, New York, NY, USA, ACM, pp. 85–97 (1998).
- [3] Kennedy, J. and Eberhart, R.: Particle Swarm Optimization, *IEEE International Conference on Neural Networks*, pp. 1942–1948 (1995).
- [4] Kulkarni, R. V.: Particle Swarm Optimization in Wireless-Sensor Networks: A Brief Survey, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, Vol. 41 (2011).