

應用於無線感測網路之節能分群演算法設計

施再繁
朝陽科技大學資訊工程系
助理教授
tfshih@mail.cyut.edu.tw

黃義恒
朝陽科技大學資訊工程系
研究生
i_am_basspaul@yahoo.com.tw

摘要

在科技日益發達的時代中，無線通訊已成為日常生活中不可獲缺的一部分，其中無線感測網路能即為一新興的熱門應用，如何使無線感測網路有效節能達到長時間穩定的通訊，是目前最急需要解決的問題。本研究將採用分群技術來解決耗能問題，此技術中最重要的部分是叢集頭(Cluster Head, CH)的選擇，然後再進行有效的分群，以增加整體網路的穩定度，延長網路的生命週期。我們利用各節點的剩餘能量(residual energy)、鄰近節點數量、節點與鄰近節點的距離為計算因子，設計出有效的叢集頭競爭積分(Score)計算公式，只需做單次積分比較即可找出最合適的節點做為叢集頭。在我們的方法中，更利用備用叢集頭的概念來減少重新尋找叢集頭的次數及耗能，亦可減少在重選叢集頭時所造成網路通訊的負載，提高網路生命週期。由電腦模擬的結果可得知我們的方法比之前已被提出的方法有更佳的效能。

關鍵詞：無線感測網路、分群演算法、叢集頭

1. 前言

無線感測網路是由多個無線感測器所組成的網路，通常應用於觀測環境變化的資料傳輸，除此之外也常應用於軍事、醫療、災害的偵測上，一般情況下，感測節點常常佈建於惡劣的環境，電池的電力是其能量的主要來源，一旦電池的電力耗盡，因能源補充不易，節點即無法繼續運作及進行資料的傳輸。無線感測

網路的主要特性為僅能利用有限的計算能力和電力來做資料的傳輸，故在此特性下要如何使無線感測網路能穩定且可長時間運作，為本研究所要探討的主要目標。

在本研究中，我們以 LEACH 為基礎，設計新的叢集頭選取方式，提出一更有效率的節能分群演算法 HEEC (An High Energy-Efficient Clustering Algorithm)，此演算法是利用節點與鄰近節點的距離、節點的剩餘能量及鄰近節點的數量為依據，訂出一個叢集頭競爭積分公式，據以選出最合適的叢集頭，同時，為了避免多次重新選擇叢集頭時花費太多的能量，在選取出叢集頭的同時也一併找出叢集頭的備用節點，讓網路不會因其中一個叢集頭失效後，使得整個網路要花費能量再重新選取叢集頭，如此不但節省能量，增加網路的生命週期，亦可讓網路更加的穩定。

本論文主要架構如下：第二節探討過去提出之相關研究，第三節敘述本研究所提出之方法及架構，第四節為模擬結果之分析，第五節為結論。

2. 相關研究

本節將探討已被提出之各種分群演算法。

2.1 LEACH(Low-Energy Adaptive Clustering Hierarchy Algorithm)

LEACH 是最具代表性的階層式分群演算法，其作法為先選出叢集頭(CH)，並將所有

節點分成許多叢集，然後收集感測節點資料，整理集結後再傳送給基地台(BS)，如此不用每個節點直接傳送資料給 BS，可節省能量消耗。叢集頭的選擇是利用隨機產生 0 到 1 之間的數值，檢查是否有大於門檻值，有則成為該叢集之叢集頭，門檻值的產生是由當前回合數和建議 CH 個數比率所計算出來。各叢集選定 CH 後即開始設定其時槽(Time Division Multiple Access, TDMA)來讓各節點依序傳送資料，使各叢集資料傳輸時不會相互碰撞。由 CH 的選定到資料傳輸為一回合，每回合結束後會再重新選取 CH，如此可讓每個節點都有機會成為 CH，以平衡各叢集選出 CH 時所消耗的能量，使得整個網路可以存活更久的時間 [1]。

2.2 HEED (Hybrid Energy-Efficient Distributed Clustering)

HEED也是一種叢集架構的方法，它是以節點剩餘能源與通訊成本來計算出當選叢集頭的機率，並與通訊範圍內其他節點先比較其當選機率，以決定可否被選為叢集頭。其中剩餘能量高者當選機率高，若發生當選機率相同時，則比較通訊成本來決定叢集頭，通訊成本是節點與鄰節點通訊所需消耗能量的平均值。剩餘能量最多，且與鄰節點通訊耗費能量最少者將會被選為叢集頭。各成員節點會週期性的與叢集頭比較當選機率，當某個節點機率高於叢集頭時，則選定此節點為新的叢集頭，此方法能夠均勻地選擇能源最多的節點來當叢集頭，但是在節點能源相等或相近時，可能因經常更換叢集頭而增加傳送控制封包所造成的能源消耗[2]。

2.3 EESH (Energy Efficient Strong Head Clustering Algorithm)

EESH 也是一種叢集架構的演算法，其作法為利用平均能量來對所有節點進行叢集頭篩選，大於平均能量者可成為叢集頭的候選節點，每個候選節點會利用與鄰節點距離和鄰節點的能量來計算一個 Cost 值，鄰節點的能量越小或與鄰節點的距離越近者其 Cost 值會越大，Cost 值最大者會被選為叢集頭，之後重覆此動作直到所有節點都完成分群，於每回合結束後，將會再重新選取叢集頭。此方法可以順利找出能量最大的節點當叢集頭，但選出之節點不一定位於最適的位置來服務最多的節點，也可能因而選出過多的叢集頭來傳送資料，增加能源的消耗[3]。

2.4 EECF (Energy-Efficient Cluster Formation Protocol)

EECF 在叢集頭選取開始前，每個節點先用鄰節點數和剩餘能量做分數(Score)的計算並廣播出去，各節點會收到鄰節點的 Score 值並和自己的 Score 比較，據此將鄰節點分成兩集合，一群為 Score 大於等於自己之節點所成的集合(N^+)，另一群為 Score 小於自己之節點所成的集合(N^-)， N^+ 集合中各鄰節點利用 Score 大小排名做進一步計算得出各鄰節點成為叢集頭的機率 RCRA，回傳此數值給各 N^+ 群的鄰節點，同時該節點也會收到 N^- 群鄰節點傳來的 RCRA，匯整 N^- 群的 RCRA 後可得知該節點成為 N^- 群鄰節點之叢集頭的機率 RCNA，之後再回傳 RCNA 給 N^- 群各節點，同時該節點會收到 N^+ 群中各節點傳送的 RCNA，最後選擇 RCNA 最大者成為叢集首，並於固定的回合時間後重新選擇叢集首。此方法可以確保每個節點一定都有相對的叢集，但花費太多的時間於傳送訊息才得以選出叢集頭，整體效益不是最好的[4]。

2.5 EEE LEACH(Energy Efficient Extended LEACH)

EEE LEACH 為 LEACH 之變形方法，其作法主要是盡可能保持最短的通訊距離來提高能源的使用率，為了要減少通訊距離，因此採用多階層的分群方法，在基地台和叢集頭間多加一個分層，當進行資料傳輸時，各傳送節點會將資料傳送至所屬的叢集頭，該叢集頭再將資料傳給主叢集頭(Master Cluster Heads, MCH)，主叢集頭收集資料後，會將資料進行壓縮再傳送到基地台，以此來最小化所有節點和基地台之間的通訊距離[5]。

3. 研究方法和架構

本節將詳細介紹我們所提出的演算法。鄰節點數、節點剩餘能量及與鄰節點間的距離等均為影響叢集頭的選擇的關鍵因素，有鑑於過去的相關研究中並無相關演算法於叢集頭選擇時同時將上述因素全部納入考量，故本研究試圖將這些影響因素同時考量，提出一更有效率的節能分群演算法 HEEC (An High Energy-Efficient Clustering Algorithm)，HEEC 演算法會在做叢集頭競爭積分(Score)計算時同時考量上述所有因子，且只做單次積分比較即找出最合適的節點做為叢集頭，並利用備用叢集首來增加整體網路的穩定度。

HEEC 演算法可分為三個階段，分別為評估階段、叢集設立階段、資料傳輸階段。以下我們將對各階段的流程做詳細的說明。

3.1 評估階段

第一階段為評估階段，此階段我們先利用 Hello Message 取得鄰節點的相關訊息如：鄰節點數及與鄰節點距離，然後再以此數據計算出節點之叢集頭競爭積分(Score)。以節點 i 為

例，說明其流程如下：

- (1) 節點 i 傳送 Hello Message 給鄰節點 j 。
- (2) 接收鄰節點 j 訊息，並計算節點 i 與 j 的距離 d_{ij} 。
- (3) 計算 $Score(i)$ 並廣播給節點 i 之所有鄰節點 j 。

3.1.1 叢集頭競爭積分 Score 之計算

網路起始時，節點利用 Hello Message 訊息來查詢鄰節點數、鄰節點距離，同時查詢節點之剩餘能量，將這三種值分別乘上各別的權重後加總可得到一個叢集頭競爭積分公式 Score，分數越高者越有機會被選為叢集頭，其公式表示如下：

$$Score(i) = W_1 * \frac{degree(i)}{N-1} + W_2 * \frac{Er(i)}{E_0} + W_3 * \frac{1}{\sum_{j \in N_i} d_{ij}^2}$$

其中

$Score(i)$ ：為節點 i 之叢集頭競爭積分

W_i ：為各影響因子之權重

$degree(i)$ ：為節點 i 之所有鄰節點個數

N ：為網路所有節點個數

$Er(i)$ ：為節點 i 之剩餘能源

E_0 ：為各節點之初始能源

d_{ij} ：為節點 i 與鄰節點 j 之距離

N_i ：為節點 i 之所有鄰節點所成的集合

3.1.2 各影響因子之權重值 W_i 的計算

各影響因子之權重值 W_i 是以各權重影響整體網路存活時間之程度來決定，我們利用多次模擬各種影響因素的權重值組合，找出最合適之 W_1, W_2, W_3 的權重值分別為 0.2, 0.2, 0.6，在後面的模擬實驗中，我們將採用下列公式來計算各節點的 Score 值：

$$Score(i) = 0.2 * \frac{degree(i)}{N-1} + 0.2 * \frac{Er(i)}{E_0} + 0.6 * \frac{1}{\sum_{j \in N_i} d_{ij}^2}$$

3.2 叢集設立階段

節點 i 在收到所有鄰近節點的 Score 後，會將其所有鄰近節點的 Score 做排序即可找出節點 i 之所有鄰近節點中的最大分數為 $\text{MaxScore}_i = \max\{\text{Score}(j) | j \in N_i\}$ 和節點 i 之所有鄰近節點中的最小分數為 $\text{MinScore}_i = \min\{\text{Score}(j) | j \in N_i\}$ ，再與節點 i 的 Score 做比較後，節點 i 即可決定自己應該成為叢集頭節點(CH)或成為叢集成員節點(Cluster member, CM)，當節點成為 CH 後，隨即於鄰近節點中找出剩餘能量最大的兩個備用 CH 節點，其目的在當 CH 的能量低於門檻值即可直接以備用節點當新的 CH 而不需全部進行重分群，故可有效減少整體網路之重分群次數，減少整體能量之消耗，增加網路的存活時間及資料傳輸的穩定性，提高網路的效能。其運作流程如下：

- (1) 等待並接收所有鄰近節點 j 的叢集頭競爭分數 $\text{Score}(j)$, $j \in N_i$ ，找出所有鄰近節點之最大分數 MaxScore_i 和最小分數 MinScore_i 。
- (2) 如果節點 i 若非 CH，且 $\text{Score}(i) > \text{MaxScore}_i$ ，則節點 i 立即成為 CH；如果 $\text{Score}(i) = \text{MaxScore}_i$ ，則選擇 ID 較大者成為 CH，隨後發送 CH 訊息。
- (3) 若節點 i 非 CH，將等待所有 $\text{Score}(i) < \text{Score}(j)$, $j \in N_i$ 鄰近節點 j 的 CH 訊息，若沒有接收到任何鄰近節點 j 的 CH 訊息，則立即宣告自己為 CH。
- (4) 若節點 i 非 CH，並且 $\text{Score}(i) \leq \text{MinScore}_i$ 則立即宣告自己為 CM 節點並等待 CH 訊息，若沒有收到任何 CH 訊息，則成為獨立節點，會將訊息直接傳送給 BS 節點。
- (5) 若節點 i 成為 CH，會等待 CM 的 join 訊息，建立 TDMA 排程，並依據剩餘能量大小選定兩個能量僅次於 CH 的

node 為備用 CH。

- (6) 若節點 i 成為 CM，則會選擇擁有最大 Score 的鄰近節點當作 CH，並向其發送 join 訊息及等待接收 TDMA 訊息。

3.3 資料傳輸階段

當叢集設立階段完成分群之後，即可開始進行資料的傳輸，持續傳輸直到 CH 能量低於起始傳輸時的 30%，便會將備用 CH 節點更換成新的 CH 以便持續傳輸資料，當全部的 CH 有 30% 是由備用節點來替補為新 CH 時，網路將會重新進行分群。整個流程說明如下：

- (1) 若節點 i 成為 CH，開始接收資料，當收到所有 CM 節點之訊息後，傳送整合資料給 BS 節點，之後一直重覆接收資料及傳送資料給 BS 的動作。
- (2) 若節點 i 成為 CM，開始傳送資料，之後不斷重覆傳送資料的動作，直到整體網路重新分群選擇新的 CH 或節點 i 能量耗盡。
- (3) 當 CH 能量低於 30% 時，選取其能量較大之備用 CH 節點成為新的 CH，該備用 CH 節點要比原 CH 節點能量大二倍，才可成為該叢集之新 CH。若備用 CH 能量皆無達到較原 CH 節點能量大二倍，則不作 CH 變更，直到網路重新分群。
- (4) 當網路中有 30% 的叢集都選用備用 CH 或網路當中任一個 CH 節點能量耗盡，整體網路就會重新分群選擇新的 CH，即完成一個回合的網路運作。

4. 演算法效能模擬

為了驗證我們所提出之方法的效能，我們針對以下項目作效能分析：網路存活時間 (lifetime)、網路耗能、網路重分群頻率。

網路存活時間，是從網路開始運作後，到最後一個節點死亡所經過的時間。網路耗能，是指網路開始運作後，某一時刻點所觀測到之網路整體消耗能量的情況，在實驗中我們每隔 10 秒鐘記錄一次耗能情況。網路重分群頻率，即為單位時間內的網路重分群的次數。

4.1 實驗環境

本研究之模擬環境，採用 NS2 軟體平台開發模擬程式，以 LEACH 之預設模擬參數為基礎進行模擬，模擬環境之各項假設如下：

- (1) 感測網路範圍固定。
- (2) 基地台和所有感測節點位置固定。
- (3) 所有感測節點初始電量皆相同。
- (4) 基地台電量無限且知道所有感測節點電量。
- (5) 所有感測節點感測範圍相同，且可直接跟基地台通訊。
- (6) 每個感測節點，在每回合都會上傳資料至 CH，CH 收集後再轉傳至 BS。
- (7) 每回合感測節點，上傳資料會包含剩餘電量。

模擬參數設定如下：101 個靜態感測節點 (100 個感測器節點和 1 個基地台)，分佈在 100 m x 100 m 感測範圍內，基地台位於 (50, 50) 位置，傳送封包大小，設定為 25 bytes，採用與 LEACH 相同電力消耗模型，節點初始能量 2 J，資料處理所耗費的電量為 5 nJ/bit/signal，傳送與接收能量消耗為 50 nJ/bit，放大器電量消耗為 10 J/it/m²。

模擬環境參數設定，如下表所示：

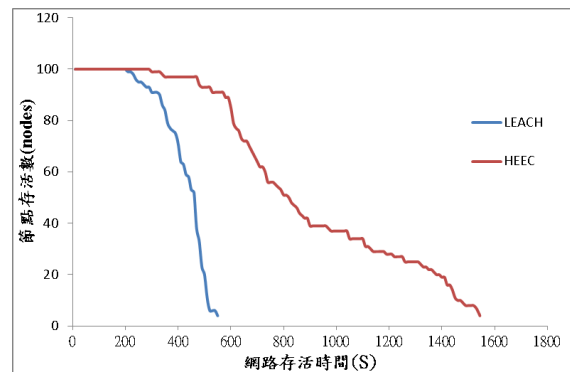
參數名稱	模擬參數值
感測範圍	100m x 100m
基地台位置	(50, 50)
最適叢集數	$k = 5$

實驗節點數(node)	101(含 BS)
節點初始電量	2J
傳送封包電量消耗 ($E_{Tx-elec}$)	50nJ/bit
接收封包電量消耗 ($E_{Rx-elec}$)	50nJ/bit
放大器電量消耗 (ϵ_{amp})	10pJ/bit/m ²
資料處理電量消耗	5nJ/bit/signal
傳送封包大小	25 bytes

4.2 實驗結果分析

(1) 網路存活時間

如圖一所示 HEEC 比 LEACH 多了近三倍的存活時間，而在第一個節點死亡的表現上也較 LEACH 多了近 100 秒的時間，之後節點的存活數也是漸近的死亡，整體來看，HEEC 較 LEACH 穩定且有更長的網路運作時間，因此可得知 HEEC 在 CH 的選擇方式較 LEACH 改善許多。

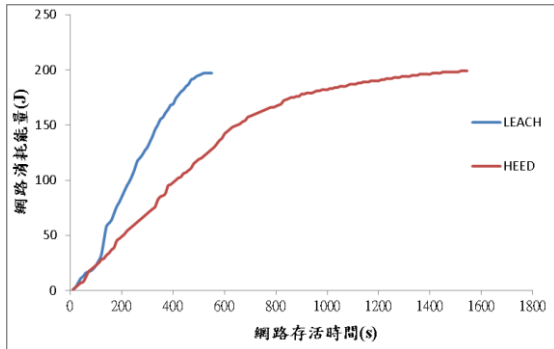


圖一、網路存活時間

(2) 網路平均耗能

如圖二所示 HEEC 比 LEACH 圖形更加的平滑，可知 HEEC 在能量消耗上非常平均，在網路的後期更因叢集變少後，能量的消耗可以更平緩，並能得知在同一時間點，整體網路是

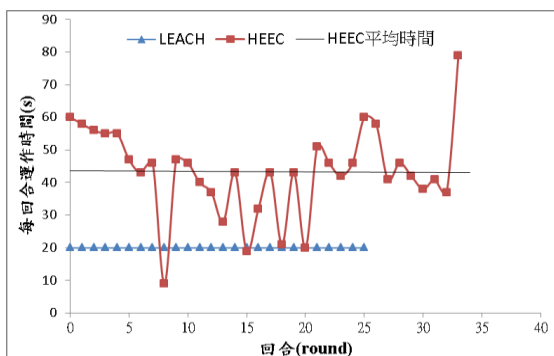
更加的節能。因此可證明選擇 CH 的重要性，當選擇到正確的 CH，不但可以延長網路的運作時間，也可以減少能量消耗，進而達到網路可長時間運作的需求。



圖二、網路耗能

(3)網路重分群頻率

如圖三所示，網路從開始分群到重分群每個回合運作的時間，HEEC 在每個回合運作時間都大於 LEACH，因此重分群之頻率會較 LEACH 低，我們的方法是以節點能量的消耗情況來做重分群之判斷，從圖中可了解 HEED 之分群演算法運用備份節點的方法，重分群的頻率不但比固定回合時間方法低很多，亦可節省重分群時的過多耗能，在圖三中 HEED 的曲線也逼近平均回合運作時間，表示每回合所花費的時間大都相同，即代表網路穩定度不受能量的消耗情況做重分群之判斷所影響。



圖三、網路重分群頻率

5. 結論

在過去相關研究中，可知叢集的形成主要有三大因素影響，分別是鄰節點的距離，鄰節點數及節點的剩餘能量，本研究將這些因素全都一併考量，做單次的叢集頭競爭積分 Score 比較，快速的選出叢集頭進而完成叢集分群，以此來減少能量的消耗增加網路運作時間，同時運用備用 CH 節點的選擇，減少網路重分群的頻率，達到網路運作所需要的穩定度。

從模擬結果中可得知，本方法與 LEACH 方法相較下來可以有效的節省整體能量消耗，增加網路運作時間，在重分群頻率方面，利用備用 CH 節點的方法可以有效的降低，來增加網路的穩定度，讓資料傳輸能更順利。未來可以利用此方法形成穩定的網路來研究增加網路的資料傳輸量，和資料傳輸的正確性，進而增廣其應用面。

6. 參考文獻

- [1] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless micro-sensor networks," **IEEE Transactions on Wireless Communications**, Oct. 2002, vol. 1, no. 4, pp.660-670
- [2] O. Younis and S. Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed clustering approach for Ad Hoc sensor networks," **IEEE Transactions on Mobile Computing**, Oct.-Dec. 2004, vol. 3, no. 4, pp. 366-379
- [3] Zhou W, Chen HM, Zhang XF. "An energy efficient strong head clustering algorithm for wireless sensor networks." **International conference on wireless communications, networking and mobile computing (Wi Com' 07)**, 2007, pp. 2584-2587.

- [4] A. Chamam, S. Pierre “A distributed energy-efficient clustering protocol for wireless sensor networks” **Computers and Electrical Engineering archive** , Mar. 2010, Volume 36 Issue 2, pp. 303-312
- [5] M. Sharma, K. Sharma “An Energy Efficient Extended LEACH (EEE LEACH)” **Communication Systems and Network Technologies (CSNT), 2012 International Conference**, 11-13 May 2012, pp.377-382