Wireless Pers Commun (2017) 94:17–29
DOI 10.1007/s11277-015-3154-z

An Energy-Efficient Balancing Scheme in Wireless Sensor Networks

Hye-Young Kim[1] **· Jinsul Kim**[2]

Published online: 24 December 2015

Abstract A typical wireless sensor network is conceived as bring a very large collection of low-powered, homogeneous nodes that remain stativ post-deployment and forward sensed data to a single sink via multi-hop communication. During the recent years, many energy-efficient load balancing protocols have been proposed for wireless sensor networks. Because a wireless networks consists of a large number of nodes with limited resources, the load balancing protocol is one of the key issues which can be solve the tradeoff between the service capacity and energy efficience. Load balancing protocols typically employ only a network capacity oriented approach in the next hop node is selected on adjacent or network information. This approach draw into a large overhead when the accurate adjacent information is needed for efficient and reliable routing. When an application service is caused large interaction between the adjacent nodes, the previous load balancing protocols without considering this issue were re-allocated the adjacent nodes and the other adjacent is re-allocated another region. This is not efficient for network performance because the previous protocols are generated the large overhead by increased routing and overhead. So, we propose a user-oriented load balancing scheme for an energy-efficient load balancing in wireless networks which is based on allocate load on wireless sensor nodes proportionally to each of the agent's capacity and user-oriented approach. This proposed scheme is combined dynamic provisioning algorithm based on greedy graph and user oriented load balancing scheme for maintain of the performance and stability of distributed system in wireless sensor networks. We address the key functions for our proposed scheme and simulate the efficiency of our proposed scheme using mathematical analyze.

✉ Jinsul Kim
jsworld@chonnam.ac.kr

Hye-Young Kim
hykim@hongik.ac.kr

[1] School of Game/Game Software, Hongik University, 2639 Sejong-ro, Jochiwon-eup, Chungcheongnam-do 339-701, Korea

[2] Department of Electronics and Computer Engineering, Chonnam National University, Gwangju, Korea

Keywords Load balancing · User oriented · Energy-efficient · Wireless sensor networks · Greedy algorithm

1 Introduction

An early research into Wireless Sensor Networks (WSNs) resulted in a definition as a large-scale wireless, ad hoc, multi-hop, un-partitioned network of homogeneous, tiny nodes, mostly immobile sensor node that would be randomly deployed in the area of internet [1]. A large number of tiny nodes have limited computation, memory storage, communication range, and battery power. They are utilized in a wide range of applications, including military applications and the monitoring of oceans and hard-control environment. Wireless sensor nodes have maximum utility when they can be used "anywhere at any time" [2]. A recent market research report claims that the Industrial WSNs market will be worth \$3.795 billion by 2017, experiencing a Compound Annual Growth Rate(CAGR) of 15.58 % [3] while a third report on wireless sensor devices predicts a 43.1 % CAGR leading to a market worth \$4.7 billion by 2016 [4].

Without load balancing, a coordinator with excessive number of wireless sensor nodes may lead to more energy consumption and experiences data reception delay as compared to the other coordinators. Energy-efficient is a crucial issue in WSNs due to the limited capacity of batteries of the sensor node. This could be because the network consists of so many nodes that replacing all depleted batteries is not feasible or because the network is located in a remote or hostile environment. In a WSNs, frequent changes in large scale network topology lead to a challenge for delay sensitive data reception. When nodes that are associated with the same coordinator send data at the same time, there will be a delay or a failure of data received by the coordinator. A lot of work on the WSNs field has been carried out resulting in the development of the WSNs on a wide variety of application and system with vastly varying requirements and characteristics. At the same time, various energy-efficient load balancing protocols have been designed and developed for WSNs in order to support network performance.

In the WSNs, the wireless sensor nodes can interact with each other in the same world at the same time [1, 5], so it is the large number of nodes interacting simultaneously, reaching the number of tens of thousands [6]. In some instances, a large number of sensor nodes' results in bandwidth demand that is a square function to the number of the applications. In other situations, there may be liner relationship to WSNs. Another important consideration is the overhead of the distribution. As the wireless sensor nodes need to communicate with one another, there must be a way to minimize this traffic. Reducing the waste of resources of the agent system for wireless sensor nodes must prevent the presence of hotspots from degrading the quality of the application beyond a tolerable limit. There is an important issues when consider a load balancing scheme in WSNs. It must be established within a short time to adapt with the change in wireless sensor nodes locations, and the parameter that triggers the action must relate to the changes in the topology.

Therefore, we propose a user-oriented load balancing scheme for an energy-efficient load balancing in wireless networks which is based on allocate load on wireless sensor nodes proportionally to each of the agent's capacity and user-oriented approach in WSNs.

The rest of the paper is structures as follows. Section 2 introduces related works on load balancing schemes on WSNs. In Sect. 3, we describe our proposed a load balancing scheme for energy-efficiency in WSNs. Also in Sect. 4, we address the efficiency of our proposed scheme using analytical models and compare the results with the previous researches. The final section, we constitute a summary of our proposal and suggest a further study directions.

2 Related Works

A lot of protocols have been proposed to efficiently manage the consumption of energy in WSNs. The load distributing problem in conventional distributed systems has been studied for many years. It can be classified into static and adaptive algorithms, where adaptive algorithms can be regarded as special class of dynamic algorithm. Dynamic load distributing algorithms can be further classified into load sharing algorithms and load balancing algorithms [7–10].

In WSNs, communication efficiency and power consumption are usually the main obstacles to improving service availability and energy efficiency. The power consumption of node and link is highly related with the corresponding traffic volume. On method to solve the tradeoff between communication efficiency and power consumption is to distribute the traffic uniformly across the network. To deal with this issue, a number of load balancing approaches have been previously proposed.

There have been many efforts in the literature toward developing user association rules considering interference avoidance and/or cell load balancing [11–13]. To avoid interference when frequency is universally reused and inter-cell interference is severe, centralized approaches have been considered [14, 15]. The basic idea is to schedule users across cells so that they do not severely interfere with each other. This is called inter-cell coordinated scheduling. Earlier work on load balancing also mostly assumed a centralized controller that governs the BSs and the MTs with access to all the necessary information [16]. However, centralized approaches, for either interference avoidance and (or) load balancing, may require excessive computational complexity and message overhead, which increase exponentially in the size of the network.

Recently, much work has been done applying load balancing to WSNs [17] and, more specifically, directed diffusion. In [18], multiple paths are constructed between the sink and the sources. Data are spread and distributed along different paths according to several energy-related parameters, and, as such, the load across different nodes tends to balance. One of the problems of this and other multiple path approaches are that an insufficient amount of paths are generated due to the multiple distinct routes merging at some point in the network. While the merging of distinct routes can be seen as an advantage in terms of increasing the data aggregation capability, the nodes in the merged path will consume more energy than others; consequently, the network will be unbalanced and die prematurely. To overcome this problem, [19] proposed to force inactive nodes to take part in a routing path according to a local greedy algorithm. Similar to this work, the work in [20] proposed a multipath routing scheme, which creates several paths between source and sink. However, in this algorithm, it is possible that distant nodes can be selected even though a more energy-efficient path to the source may exist. In this case, the nodes along the longer path would expend more energy, and thus, this would lead to premature network death. The work proposed in the present paper is similar to [21], but it differs by not choosing to select nodes based on their relative distance from the sender.

Other approaches include a load balance solution derived by formulizing the network lifetime maximization problem into a nonlinear programming [22], a method to achieve load balance among all clusters each of which is composed of several nodes in wireless networks [23, 24]. This least energy constraint node in a cluster acts as the cluster head and performs as gateway to forward data to the base station. However, this method achieves load balance in an offline way that is based on the static traffic demand knowledge and topology information.

A load shedding method is one in which an overloaded server attempts to shed its load to its neighbors [25]. After finding a lightly loaded neighbor server, it transfers some of its boundary microcells to that neighbor. To form the group of microcells to transfer, a microcell from the border is chosen, and others are added in a breadth first search (BFS) order. A very similar work, which also uses BFS, is described in [26].

A dynamic geographic hash table for data centric storage is proposed in [27, 28]. A temporal-based hashing function redefines the original basic operation of the geographic hash table and combines the timeslot. For each timeslot, each event is mapped into a grid cell. For the next timeslot, the next event will be mapped into another grid cell and thus, achieve load balancing. The concept of node contribution potential is used to avoid situations where events are mapped into locations in which the surrounding nodes do not have enough resources to service the networks [29–31]. However, the contribution potential value and the coordinate of the node with the highest potential in a cell must relay back to the sink. The sink selects a set of cells with potentials above a certain threshold to be the location set. Hence, it may result to a higher overhead. In addition, all nodes must be time synchronized and hence, will need extra overhead.

Although there are excellent researches for WSNs, this paper shows an energy-efficient load balancing protocol in WSNs. Our paper is focus on an energy-efficient load balancing in WSNs which is based on allocate load on wireless sensor nodes proportionally to each of the network's capacity and user-oriented approach in WSNs.

3 Our Proposed Load Balancing Scheme

For an energy-efficient load balancing approach, we propose that the traffic generated by user-oriented wireless sensor nodes is not simply linear, but the circle for each cluster of wireless sensor nodes in WSNs. And another pointer is the overhead, which must be taken into account, no matter which load balancing algorithm is being used.

Our proposed scheme is based on greedy growing algorithm and an user-oriented load balancing scheme, seeks to allocate the heaviest wireless sensor node and nearest wireless sensor nodes with communication to the regions managed by the most powerful agents. It is considered that an initial division of the wireless networks in Fig. 1.

In the Fig. 1, the vertices indicates the wireless sensor nodes and the selection starts with the vertex of weight six and at each step the vertex is connected by the heaviest edge and nearest of the it's region's mean is added to the partition. The selected edges and the vertices belonging to the new partition are highlighted. The proceeding and result of the first execution show in [15]. Also, the second proceed and result of our proposed scheme is addressed in [15]. Result of first execution of proposed scheme is shown in Table 2. In the Table 1, the R means the number of wireless network group's region and F means the fragment of the wireless networks.

When wireless sensor nodes are within visual range of each other, it is best to locate them in one wireless sink node [26, 27]. To manage wireless sensor nodes like the above, both the scope of interactions and probable interactions have to be taken into account and

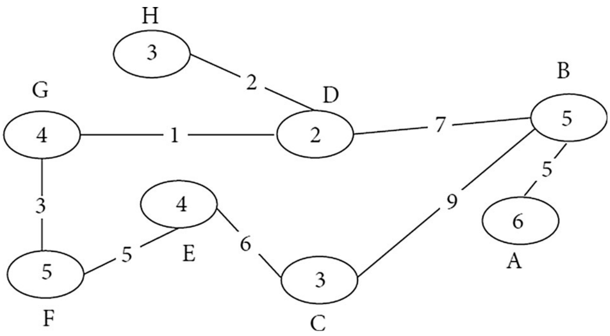

Fig. 1 An example for proposed scheme using greedy algorithm

Table 1 Proceed of first execution of proposed scheme [15]

Region	Vertex	# of vertex	Weight	Center	
				X	Y
R1	A, B	2	17	0	0.5
R2	E, D	2	9	1.5	0.5
R3_F	F, G	2	5	0.3	0.5
R4_F	C	1	4	1	1
R5_F	H	1	3	2	2

R = 2, F = 3 R: wireless network group F: fragment

then set the neighboring wireless sensor node area to about 1.5–2 times the visual range with the wireless sensor node at the center. All wireless sensor node or sink nodes have the same capacity, each one being able to handle the same wireless sensor nodes. When new wireless sensor nodes proceed to log-in, we first check whether the existing wireless sensor nodes are in the nearest sensor node area. If so, then distribute the new wireless sensor nodes to the nearest wireless mobile sensor node areas (Table 2).

For load-balancing, each the center of each wireless sensor node gets updated at the load-balancing agent and this point is equated by the average coordinates of the wireless sensor nodes. Load-balancing agents distribute wireless sensor nodes to the agent which is closest to the center of the agent when new wireless sensor nodes log-in and when all wireless sensor nodes are not included in the nearest section. Figure 2 shows the hop counting energy and the region of the wireless sensor node. In Fig. 2, the vertices and regions are showed the user-oriented load balancing by using Fig. 1.

The proposed scheme is follows.

Table 2 Result of second execution of proposed scheme [15]

Region	Vertex	# Vertex	Weight	Mean	
				X	Y
R1	A, B, D, E	4	20	0.3	0.3
R2	F, G	2	9	3	0.5
R3_F	C	1	3	2	2
R4_F	H	1	3	2	2

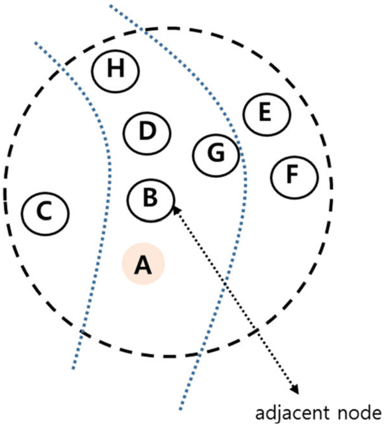

Fig. 2 User-oriented hop count and allocation

1. select the group of local regions: the most of the largest cell allocate to the largest wireless sensor node
2. calculate the weight of each of the cell and represent all of the cells to the graph
3. allocate coordinate of agent to average of cells
4. decision of the cell adjacent of the agent through the coordinate of the cell
5. user-oriented load calculate for balancing each of the wireless sensor nodes
6. balance these regions, assigning to each one a weight which is proportional to the power of its wireless sensor node
7. refine the partitioning, reducing the overhead

Here, we use the parameters listed in Table 3 for simulating of the load balancing in the WSNs. Table 4 is showed the detailed our algorithm.

We propose an analytical model for the load balancing in WSNs. We consider two main types of resources used for load calculation. Let us consider N wireless sensor nodes connected to a distributed regions of a total of R regions. Let C be the total number of wireless sensor nodes in a WSN that has a region R. The node density of the WSN is equal to $\delta = \frac{C}{R}$. Let r be the transmission range of a wireless sensor node. Live candidate nodes, LC, are located within one-third of the whole transmission range in Fig. 2. So, the number of the region, $L(R)$, can be calculated by;

$$L(R) = \frac{1}{3} * \pi * r^2 * \delta \quad \left(\text{i.e.} := \frac{N}{R}\right) \tag{1}$$

They are CPU and network. For modeling the load of one wireless sensor node in a WSN, we consider three basic time activities with one node tick. T_i is the computation of the interaction between pairs of entities. The T_m is reception of event messages from each wireless sensor node. The T_u is the update of entity states received from or send to another wireless sensor node. We modeling the CPU time T_M spend for send and receive messages from agent to each wireless sensor node as [15]

$$T_M = C \cdot t_m \tag{2}$$

Also,

$$T = (N - C) \cdot T_u + (BE - AE) \cdot T_u + AE \cdot T_u \tag{3}$$

Table 3 The parameters of the simulation

Parameters	Descriptions
N	Number of nodes
R	Number of regions
I	Total number of interactions
$L(R)$	Number of Region R's nodes
$P(R)$	Total capacity of region R
$W_i(R)$	Weight of region R
S_i	Node I's coordination
BE	Number of moving node
AE	Number of active entities
C	Number of connected wireless sensor nodes
IC	Number of avatars interacting with any other entities
Ti	Consumed time for each computation
T_u	Update of entity states received from/sent to another nodes
T_m	Reception of event message from/sent to another nodes
P_{ci}	Percentage of CH
P_{ei}	Average number of interactions as a percentage of BE
P_{ui}	Ratio between time necessary for one entity update
B	Bandwidth
D_{update}	Location update of wireless sensor nodes

4 Performance Analyses

In this section, we introduce the perform simulation in WSNs to evaluate the effectiveness and performance of our approach. Then we compare and analyze the result of our scheme and others. MATLAB10 was used in our simulation for a performance analysis.

We compare the performance of load balancing with following two items:

1. Greedy graph algorithm: this is a widely used for conventional load balancing algorithm of WSNs, where a greedy graph is constructed for collection data and re-allocate the load balancing using dynamic load balancing.
2. User-oriented algorithm: in local load balancing protocols, beaconing is required periodically for setting up energy information. The energy information stored in the table does not reflect the actual information, so wireless sensor nodes consume energy over time. So, this algorithm uses its own information, which is always accurate and updated, to evaluate its eligibility to be selected as a next hop wireless sensor node.

We collect the number wireless sensor nodes used to form load balancing under different size of the WSNs as shown Fig. 3. As sown in Fig. 3, the average node number used by our scheme is almost constant.

To explicitly show the energy efficiency performance of our scheme, we measure the average energy consumption of the whole WSNs in each period, as shown Fig. 4. We can observe that our scheme consume less energy on average compared to the previous

Table 4 Our proposed algorithm

```
cell has a position coordinates(X,Y), Cell(X_m, Y_m)
weight_to_divide ← 0;
free_capacity ← 0;

for each region R in region_list do
      weight_to_divide   <- weight_to_divide + W_r(R)
            free_capacity <- freee_capacity + P(S(R))
            calulate distance between cells
            S(C_x, C_y) ← cell(M)
        // S(C_x, C_y): agent's coordinates, Cell(X_m, Y_m): cell's mean
            cell_list   ← list of cells from R in increasing order of cell`s distance
            while fracCr(R) > fracCp(S(R)) do
              if Cell(X_m, Y_m)is not adjcent of S(C_x, C_y)
                        C ← first element from cell_list
                        remove C from R
                        remove C from cell_list
               endif
            end while

sort region_list in increasing order of U(S(R))

// user-oriented load calculate for balancing
for each region R in region_list do
      weight_share ← weight_to_divide* Pp(s(R)) / Free_capacity
      L(R) = 1/3 * π * r^2 * δ //δ = N/R

while Wr(R) < weight_share do
      D_update (i)(Ni) = S(i) + Si / (L(R) + 1)

if there is any cell from R neighboring a free cell then
          R ← R U {neighbor free cell with the highest Intc(C)
else if there is any free cell then
          for i = 1 to L
             if D (S(i), D_update(i) < min
               min ← D (S(i), D_update(i))
                 R ← R U {cell}
               Endif
          endfor
      else
            R ← R U{heavist free cell}
endif
end while
end for
```

researches. This is because on average they are using fewer wireless sensor nodes to transmit packets in each period.

Our scheme showed that even though the number of wireless sensor nodes increased, the occurrence of reallocation for load-balancing did not increase as much. As shown in Fig. 5, when the number of wireless sensor nodes exceeds 700, our scheme is a difference

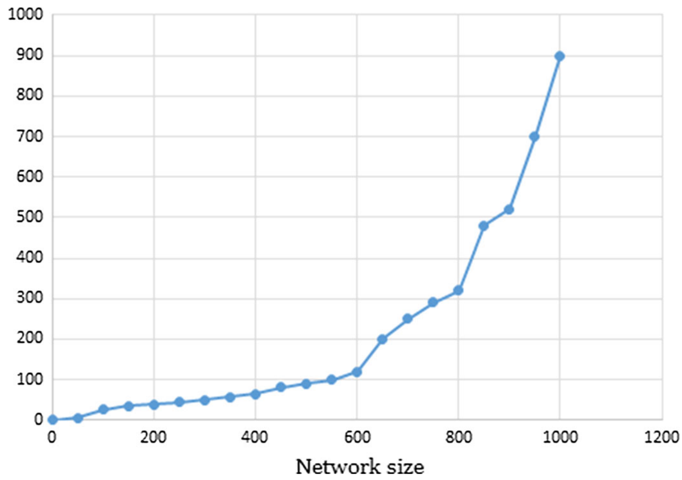

Fig. 3 Time overhead of nodes

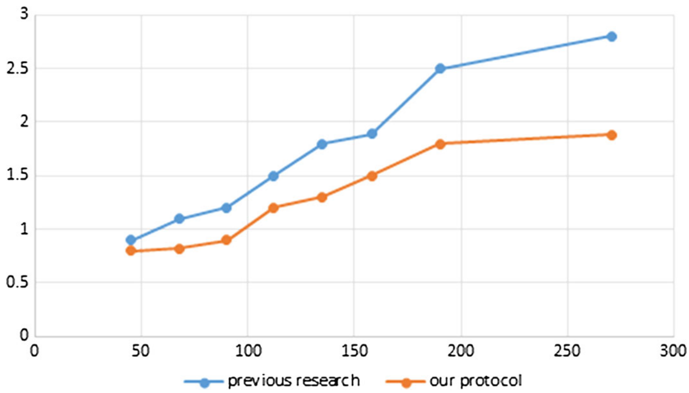

Fig. 4 Average energy consumption

between the previous researches. In addition, we triggered messages between wireless sensor nodes in order to analyze the load balancing of the interaction between wireless sensor nodes in WSNs.

In Fig. 6, we show that even when the number of interaction messages exceeds 1200, the number of reallocation for load balancing does not increase much in our scheme. Therefore, load balancing caused by many interactions between wireless sensor nodes is much more efficient than those of previous researches. In Fig. 7, we illustrate of the load of the wireless sensor nodes using our proposed scheme. We found that the load is kept constant in the wireless sensor nodes on our proposed scheme.

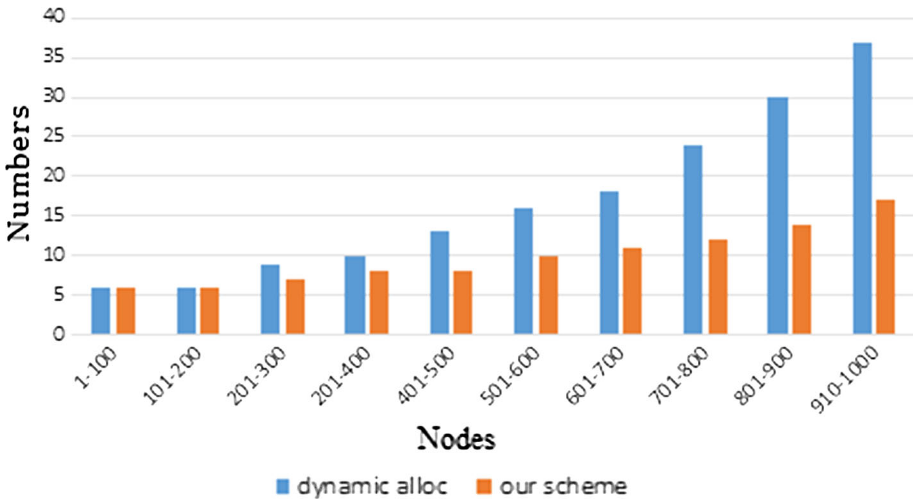

Fig. 5 Reallocation by increasing nodes

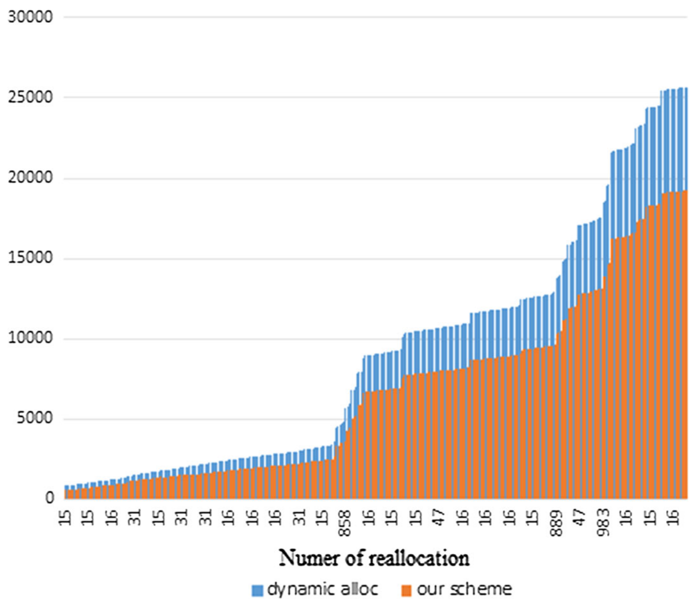

Fig. 6 Reallocation by increasing interactions

5 Conclusion and Future Works

We have proposed a user-oriented load balancing scheme for an energy-efficient load balancing in wireless networks which is based on allocate load on wireless sensor nodes proportionally to each of the agent's capacity and user-oriented approach. This proposed scheme is combined dynamic provisioning algorithm based on greedy graph and user

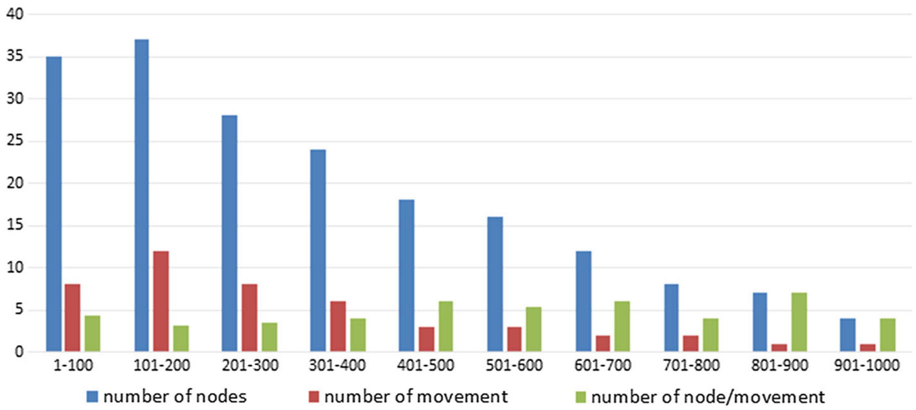

Fig. 7 Load of the wireless sensor nodes

oriented load balancing scheme for maintain of the performance and stability of distributed system in WSNs. We address the key functions for our proposed scheme and simulate the efficiency of our proposed scheme using mathematical analyze. We have presented extensive simulation results to show that load balancing is achieved at the expense of energy for communication. When compared with previous researches, our proposed scheme in Sect. 4 was proven by simulations through mathematical modeling to be more efficient than previous research, even though the wireless sensor nodes increase, the average energy consumption, the time overhead under the different WSNs sizes and the probability of node replacement for load-balancing decreases or constant compared to previous research as shown in Figs. 3, 4 and 5. Despite that the interactions within the node along with the load increases, efficient load-balancing still occurs as seen in Fig. 6. We found through the Fig. 7 that the load is kept constant in the wireless sensor nodes on our proposed scheme. Based on the simulation results, we find that compared to our scheme achieves an energy-efficient load balancing in WSNs.

Acknowledgments This research was partially supported by the IT R&D program of MSIP(Ministry of Science, ICT and Future Planning)/IITP(Institute for Information & Communications Technology Promotion) [12221-14-1001, Next Generation Network Computing Platform Testbed] and also supported by 'The Cross-Ministry Giga KOREA Project' grant from the Ministry of Science, ICT and Future Planning, Korea. I would like to thank Mr. Joseph Malone for his specific help with the review in this paper.

References

1. Ahmed, D., & Shirmohammadi, S. (2008). A microcell oriented load balancing model for collaborative virtual environments. In Proceeding of the IEEE conference on virtual environments, human computer interfaces and measurement systems, VECIMS, pp. 86–91.
2. Barraca, J., Matos, A., & Aguiar, R. (2011). User centric community cloud. *Wireless Personal Communications, 58*, 31–48.
3. Kim, H.-Y. (2012). An efficient access control scheme for online gaming server, proceeding of the computer science and convergence, lecture notes. *Electrical Engineering, 114*(1), 259–267.
4. White, W. M., Koch, C., Gupta, N., Gehrke, J., & Demers, A. J. (2007). Database research opportunities in computer games. *SIGMOD Record, 36*(3), 7–13.

5. Bezerra, C. E. B., Comba, J. L. D., Geyer, C. F. R. (2009). A fine granularity load balancing technique for MMOG Agents using a KD-tree to partition the space. Brazilian Symposium on 2009, Games and Digital Entertainment (SBGAMES), pp. 17–26.
6. Kim, H.-Y., & Park, H.-J. (2013). An efficient gaming user oriented load balancing scheme for MMORPG. *Wireless Personal Communication, 73*, 289–297.
7. He, T., Stankovic, J. A., Lu, C., Abdelzaher, T. (2013). SPEED, A stateless protocol for real-time communication in sensor networks. In Proceedings of IEEE 23rd international conference on distributed computing Systems.
8. Cheng, S.-T., & Chang, T.-Y. (2012). An adaptive learning scheme for load balancing with zone partition in multi-sink wireless sensor network. *Expert Systems with Appliacations, 39*(10), 9427–9434.
9. Kacimi, R., Dhaou, R., & Beylot, A. L. (2013). Load balancing techniques for lifetime maximizing in wireless sensor networks. *Ad Hoc Networks, 11*, 2172–2186.
10. Gupta, G. & Younis, M. (2003) Load-balanced clustering of wireless sensor networks. In Proceedings of ICC.
11. Liao, W.-H., Shih, K.-P., & Wu, W.-C. (2010). A grid-based dynamic load balancing approach for data-centric storage in wireless sensor networks. *Computer and Electrical Engineering, 36*(1), 19–30.
12. Chen, M., Leung, V., Mao, S., Xiao, Y., & Chlamtac, I. (2009). Hybrid geographical routing for flexible energy-delay trade-offs. *IEEE Transactions on Vehicular Technology, 58*(9), 4976–4988.
13. Lu, Y. M., & Wong, V. W. S. (2007). An energy—Efficient multipath routing protocol for wireless sensor networks. *International Journal of Communication Systems, 20*(7), 747–766.
14. Bezerra, C. E. B., & Geyer, C. F. R. (2009). A load balancing scheme for massively multiplayer online games. *Multimedia Tools and Applications, 45*(1), 263–289.
15. Kim, H.-Y., Park, H.-J., Lee, S. (2014). A hybrid load balancing scheme for games in wireless networks. *International Journal of Distributed Sensor Networks*. doi:10.1155/2014/380318.
16. Nae, V., & Losup, A. (2011). Dynamic resource provisioning in massively multiplayer online games. *Parallel and Distributed Systems, IEEE Transactions on, 22*(3), 380–395.
17. De Grande, R. E., Boukerche, A. (2009). Dynamic partioning of distributed virtual simulations for reducing communication load, haptic audio visual environments and games. IEEE International Workshop pp. 176–181.
18. Quax, P., Cleuren, J., Vanmontfort, W., Lamotte, W. (2011) Empirical evaluation of the efficiency of spatial subdivision schemes and load balancing strategies for networked games. In Proceeding of 20th internatinal conference on computer communications and Networks (ICCN), pp. 1–6.
19. Li, X., Kim, Y. J., Govidan, R., Hong, W. (2003) Multi-dimensional range queries in sensor networks. In Proceedings of the ACM international conference on embedded networked sensor systems (SenSys), November 2003.
20. Li, J., Jannotti, J., DeCounto, D., Karger, D., Morris, R. (2000) A scalable location service for geographic ad-hoc routing. In Proceedings of the annual ACM international conference on mobile computing and networking (Mobicom 2000), Bostonn, MA, USA, August 2000.
21. Tlili, Raja, & Slimeni, Yabha. (2012). A hierarchical dynamic load balancing strategy for distributed data minig. *IJAST, 39*, 21–48.
22. Wang, R., Liu, G., Zheng, C. (2007) A clustering algorithm based on virtual area partition for heterogeneous wireless sensor networks. In Proceedings of the IEEE international conference on mechatronics and automation (ICMA '07), pp. 372–376, August 2007.
23. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: A survey. *Computer Networks, 38*(4), 393–422.
24. Shu, T., & Krunz, M. (2010). Coverage-time optimization for clustered wireless sensor networks: A power-balancing approach. *IEEE/ACM Transactions on Networking, 18*(1), 202–215.
25. Shin, K., Abraham, A., & Han, S. Y. (2006). Self-organizing sensor networks using intelligent clustering. *Lecture Notes in Computer Science, 3983*, 40–49.
26. Tarapata, G., Weremczuk, J., Jachowicz, R., Shan, X. C., & Shi, C. W. P. (2009). Construction of wireless sensor for harsh environment operation. *Procedia Chemistry, 1*(1), 465–468.
27. Raj, J. S., Hridya, K. S., & Vasudevan, V. (2012). Augmenting hierarchical load balancing with intelligence in grid environment. *International Journal of Grid and Distributed Computing, 5*(2), 9–18.
28. Haider, M. B., Imahori, S., & Sugihara, K. (2011). Success guaranteed routing in almost delaunay planar nets for wireless sensor communication. *Sensor Networks, 9*(2), 69–75.
29. Vidhate, D. A., Patil, A. K., Pophale, S. S. (2010) Performance evaluation of low energy adaptive clustering hierarchy protocol for wireless sensor networks. In Proceedings of international conference and workshop on emerging trends in technology (ICWET 2010) TCET, Mumbai, India, 2010, pp. 59–63.

30. Almazaydeh, L., Abdelfattah, E., Al-Bzoor, M., & Al-Rahayfeh, A. (2010). Performance evaluation of routing protocols in wireless sensor networks. *Computer Science and Information Technology, 2*(2), 64–73.
31. Bokhari, F. (2010). Energy-efficient QoS-based routing protocol for wireless sensor networks. *Parallel and Distributed Computing, Department of Computer Science, Lahore University of Management Sciences, 70*(8), 849–885.

Hye-Young Kim is received the B.Sc. degree and M.S. degree in Computer Science from the SookMyung University in South Korea and Ph.D. degree in Computer Science and Engineering from the Korea University of South Korea in February 2005. During her Ph.D. study, she had focusing on location management scheme, such as mobile IPv6, cellular network and network mobility. Currently, she works at Hongik University of South Korea as Associate Professor. She had developed a network protocol for 9 years (1989–1997) while she was working at Hyundai Electronics as a senior researcher. She had taught in the Department of Computer Game at Chung Kang College as an assistant professor for three years (2001–2004). Her research interests include traffic modeling and location management in mobile network and wireless network for mobile game or multimedia service.

Jinsul Kim received the B.S. Degree in computer science from University of Utah, Salt Lake City, Utah, USA, in 2001, and the M.S. and Ph.D degrees in digital media engineering, department of information and communications from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2005 and 2008. He worked as a researcher in IPTV Infrastructure Technology Research Laboratory, Broadcasting/Telecommunications Convergence Research Division, Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea from 2005 to 2008. He worked as a professor in Korea Nazarene University, Chon-an, Korea from 2009 to 2011. Currently, he is a professor in Chonnam National University, Gwangju, Korea. He has been invited reviewer for IEEE Trans. Multimedia since 2008. He was General Chair of IWICT2013/2014/2015, ICITCS2014, ICISA2015, ICMWT2015 and ICISS2015. His research interests include QoS/QoE, Measurement/Management, IPTV, Mobile IPTV, Smart TV, Multimedia Communication and Smart Space/Works.