

REVIEW

Open Access



Understanding people with human activities and social interactions for human-centered computing

Sangil Choi*

*Correspondence:
sgilchoi@gmail.com
Department of Computer
Science, Swaziland Christian
University, Mbabane,
Kingdom of Swaziland

Abstract

An ultimate goal of human-centered computing is making human beings the center of computing technologies. To make people a core component of the technology, first of all, we need to understand people and society in which people live. In this paper, we propose two important factors in order to understand people and their social interactions. Proposed human activity recognition and neighbor discovery schemes help us comprehend human activities and their social behaviors. In addition, combination of these two mechanisms provide us with an opportunity for better understanding of people in the near future. Finally, it might be worth analyzing correlation between human activities and social interactions for the group of people using our proposed schemes.

Human-centered computing

Computing technology has changed a lot of things in the world quickly. It is still affecting the way we think, the way we communicate with each other, the way we study, the way we work, and so on. Furthermore, computer hardware and software keep evolving every day and new paradigms and mechanisms continuously emerge into the world. There is no doubt that the main purpose of these technologies make human beings and our daily life convenient. In other word, the center of computing is all mankind.

However, the main purpose of computing technology may not be human beings. Artificial intelligence (AI) could be one of the promising computing technologies in the future. Many AI researchers hope that this technology will help and assist people effectively and efficiently. In addition, they believe that AI dramatically change our life in the near future. Unfortunately, all of the change are not always positive. Rapid and frequent changing of computing technologies could be difficult to use especially for elderly people. Furthermore, it is well known that innovative AI technology could give people a great fear. Some famous movies dealing with AI issues reflected this phenomenon. In addition, there was the fight of the century to play game Go between an AI Go player (AlphaGo) and a professional human Go player in March, 2016. AlphaGo finally defeated the human Go player at that game. It is definitely possible that the result of the game surprised and frightened all of the people in the world.

As we discussed before, one of the fundamental problems in current computing technologies is that there is a gap between human and computing technologies. Most current computing technologies concentrate on improving existing technologies or solving specific technological problems. This approach may not take care of human issues carefully. Originally, a computer was made for better human life, but now experts probably keep developing computing technologies for the enhanced performance of computer itself. In order to minimize the existing gaps, human-centered computing (HCC) has emerged. HCC encompasses all of the studies and efforts regarding making human beings the heart of computing technologies. Because of this reason, HCC could incorporate different types of disciplines such as computer science, sociology, psychology, cognitive science, graphic design, and anthropology. HCC research expects revolutionary movements to happen by dealing with human abilities and needs.

In [1], authors proposed three important factors that form the core of HCC system: First, the HCC system should take into account personal human abilities and limitations. Second, it should consider social and cultural environments. Finally, the system could be properly adjustable to diverse individuals and specific environments. In addition, the authors addressed the following research directions:

- New algorithms and development of diverse HCC systems;
- Integration of multiple sensors, media, and human sciences that make human beings the central point;
- New interdisciplinary academic and industrial programs;
- Discussions on the impact of computing technology that include the social, economic, and cultural contexts in which such technology might be deployed;
- Data collection from culturally diverse social situations;
- Evaluation metrics for theories, design processes, implementations, and systems from a human-centered perspective; and
- Methodologies for privacy protection and the consideration of ethical and cultural issues.

The papers [1, 2] mentioned that a key challenge of developing HCC systems is encompassing diverse disciplines that we have already discussed in computational environments. Especially, in this paper, we propose a potential computational model that focuses on understanding individual and social activities toward future HCC systems.

Human activity monitoring system

Human beings have different types of physical activities in their daily life. They could sit on a chair, walk along a street, or sometimes run or jog inside or outside a building. Although there are some different levels or degrees of activity based on their occupations, most people have their own activity styles everyday. In addition, many people are getting interested in their health so that different kinds of activity tracking devices have emerged these days. With the activity trackers, users can keep recording their own daily activities and realize the pattern of physical activities numerically. Furthermore, these activity patterns may indicate their current health status.

We proposed a human activity monitoring system using a wearable sensor in our previous paper [3]. The system consists of two primary components: activity data collection and physical activity recognition. Our system focuses on classifying and monitoring human daily activity. In this section, we briefly introduce our activity monitoring system.

Shimmer: wireless wearable sensor

For collecting and recognizing different human physical activities, we used a tiny and lightweight wireless wearable sensor called Shimmer [4]. It is widely used for studying a lot of phenomenon, implementing diverse algorithms, and making some experiments in wireless sensor networks. The Shimmer device consists of a number of integrated and extended sensors, a central processing unit, wireless communication modules, and storage devices. It can be considered as a tiny personal computer equipped with wireless communication and sensing capabilities. Especially, Shimmer has a triaxial acceleration sensor. By reading and analyzing a series of acceleration data with this sensor, we can recognize a real-time human physical activity. Shimmer has three key features: First, it has a very low power or low energy consumption capability. Second, its size is very small ($53 \times 32 \times 15$ mm enclosure). Lastly, its weight is only 15 g. Due to these features, it is currently widely used for wearable sensor applications. In addition, Shimmer is applicable to a variety of areas such as healthcare, rehabilitation, remote patient monitoring and assistive technology, environmental sensing solutions, and biomechanics.

Metrics for activity recognition

There are two typical physical activities in our daily life: static and dynamic activities. Representative static activities consist of standing, sitting, and reclining. Additionally, famous dynamic activities are composed of walking, running, and jumping. With acceleration readings from Shimmer, we compute an inclination angle and a standard deviation with vertical and anterior-posterior accelerations to recognize real-time human physical activities. The inclination angle has been calculated to classify the static activity in previous research [5–9]. After obtaining each x- and y-axis of acceleration values, we calculate the inclination angle (Φ) with Eq. (1), where A_x and A_y represent accelerations of the x- and y-axis, respectively. In our system, as mentioned before, the inclination angle is used for recognizing standing, sitting, or reclining activities.

$$\Phi = \arctan \frac{A_x}{A_y} \quad (1)$$

Most dynamic activities are closely related to the variation of accelerations in vertical direction. We decided that a standard deviation could be one of the best measurements to distinguish between static and dynamic activities. If the standard deviation value is constantly changing, we can conclude that the activity is dynamic. To get the standard deviation of the y-axis, we collect 10 samples of acceleration values of the y-axis and calculate the standard deviation (σ) with Eq. (2), where n is the number of samples, y_i is the acceleration of the y-axis at time i , and \bar{y} represents the average of the acceleration of the y-axis over n samples, respectively.

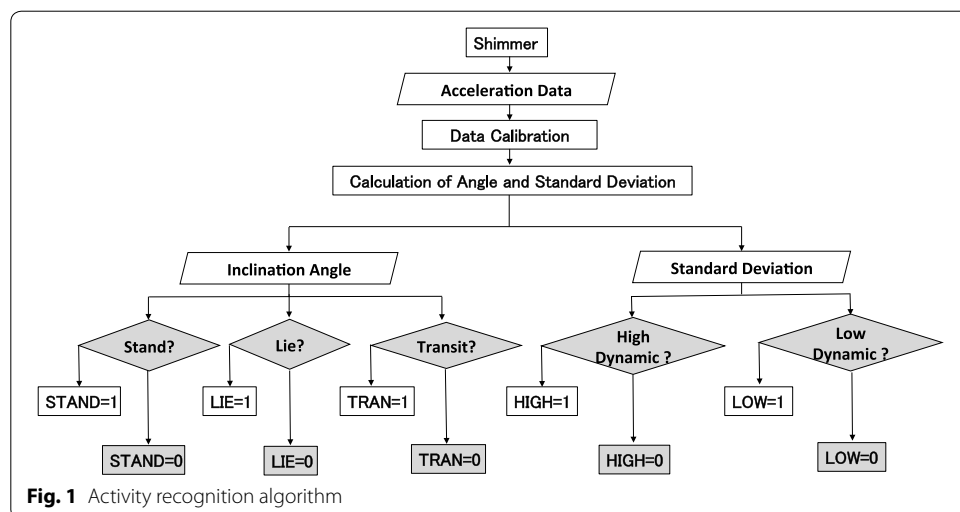
$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (y_i - \bar{y})^2} \quad (2)$$

Activity recognition algorithm

We used two measurements, inclination angle and standard deviation, for activity recognition in our system. The computation needed for these measurements is not complicated. Therefore, our activity recognition algorithm is simple and compact. This is one of the contributions of activity monitoring. The main reason is that a wireless sensor device is a resource-constrained and battery-dependent hardware, and therefore, we need to minimize the battery of sensors for computation in order to decrease energy consumption. If sensor applications consume a lot of energy and require heavy computation, this system might not be useful to collect and monitor daily physical activity in humans.

Figure 1 represents the flow chart of our entire physical activity recognition. First, we collect original acceleration data from the accelerometer in the Shimmer. Second, we perform our own data calibration process to get accurate acceleration data. Then, we calculate the inclination angle and standard deviation with the calibrated acceleration value. Finally, we compare the angle or standard deviation value with some threshold values that we computed to produce a boolean indicator. As a result, we can finally obtain a series of boolean indicators representing '0' or '1'.

How can we get the final activity data from the series of binary numbers consisting of '0' and '1' in the proposed activity recognition algorithm? Here is a concrete example. Let us consider one specific activity like *standing*. We may think that *standing* is a static posture. There is no active movement in this activity, which means that both HIGH and LOW dynamic values from the standard deviation are around zero. From the previous experimental study [3], we found that the inclination angle of *standing* is different from that of *sitting* or *lying*. Therefore, the LIE value in Fig. 1 is also zero. The TRAN value comes from the big difference of angles, so that the value is also zero because there is no big angle difference in the *standing* position. Consequently, all values are zero except the first one (e.g., 10,000). After getting these binary data, we can map them into a specific decimal number representing the specific activity and conclude that the current activity is a *standing* posture. This is how the activity recognition algorithm works.



Neighbor discovery protocol

Most recent studies of the wireless sensor network (WSN) area focus on energy consumed by wireless sensors [10–15]. Especially, neighbor discovery has been actively studied for over a decade since the research of WSN began. There are two primary reasons why the topic of neighbor discovery became popular: First, if sensors cannot find neighbors within a certain amount of time, they are unattended or useless. Second, if this situation continues, it makes the entire network partitioned or disconnected. A variety of asynchronous neighbor discovery protocols have been developed in order to address a neighbor discovery problem [16–24]. In this section, we present an asynchronous neighbor discovery protocol we proposed called *BAND* in the previous paper [25].

Neighbor discovery problem

Most wireless sensors are characterized as a resource-constrained device because they have limited computing power and their power source comes from battery. For the purpose of energy conservation, almost all sensors have their own power-saving mode. In this mode, sensors turn off their communication channel and do nothing. Periodically, they turn on their radio and try to communicate with their neighbors. Unfortunately, there is no information regarding when their neighbors wake up. Therefore, each sensor in the network should know how many neighbors are located within its communication range and how often they wake up. Traditionally, using some number of control packets, sensors tried to synchronize the meeting time of their neighbors, but it turned out there is a huge overhead of exchanging these control packets. Therefore, an asynchronous neighbor discovery scheme came into the spotlight these days.

Combinatorial block design

The neighbor discovery mechanism we proposed borrowed the concept of combinatorial block designs. Especially, we developed a neighbor discovery schedule using a *Balance Incomplete Block Design* (BIBD) [26–28]. As we mentioned before, there are two distinctive modes in a neighbor discovery schedule called active and power-saving modes. We could represent these two modes with a number zero and one. For example, we have ‘010101’ schedule, which means a sensor is in power-saving mode for the first time and then turns on its radio to communicate with its neighbors and continues this pattern repeatedly. One of the most challenging tasks for constructing neighbor discovery schedules is how two different schedules have the same active mode at the same time without any further information. In [29], according to Theorem 2.1.2, it has been proved that if a given (v, k, λ) -BIBD is a symmetric BIBD then randomly selected two blocks have exactly common λ points. This Theorem represents a fundamental property showing that two arbitrary discovery schedules could have common active time slots when we apply the symmetric BIBD to the asynchronous neighbor discovery problem without control packets.

Combination of two block designs

Although we could apply BIBD to the construction of an asynchronous neighbor discovery schedule, there is a weakness when simply employing BIBD for creating discovery schedule. The practical challenge of using a block design for asynchronous neighbor

discovery is that there is no unified block construction method. That is why we developed a block combination algorithm. A basic concept under the proposed construction technique is combining two block designs.

Let us explain how to combine two designs in detail. For simplicity, two simple designs are used for demonstration purposes. Note that we created these two designs only for the purpose of illustrating the combination process.

Step 1: choose two block designs

First, we select any two well-known block designs. Assume that there are two block designs, $A = (v_a, k_a, \lambda_a)$ -BIBD and $B = (v_b, k_b, \lambda_b)$ -BIBD. For the sake of simplicity, we use (4, 3, 2) and (3, 2, 1)-designs to demonstrate the proposed idea of block combination. Figure 2 shows (4, 3, 2) and (3, 2, 1)-designs. Here, we call the (4, 3, 2)-design a *base* design and (3, 2, 1)-design a *replacement* design.

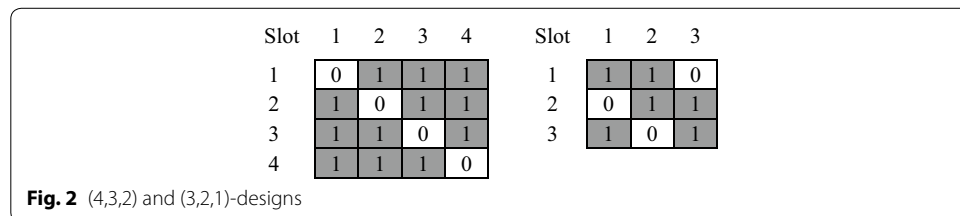
Step 2: replace each active slot of the base design with the entire blocks of the replacement design and every sleep slot of the base design with a sleep schedule of order 3. Here, a sleep schedule is defined as a square matrix of order n with all 0's

In Step 2, for every active slot in the *base* design, we replace it with the entire blocks of the *replacement* design, that is, the (3, 2, 1)-design. In addition, sleep slots are replaced with a *sleep schedule*. Let us assume that we change each active slot from the (4, 3, 2)-design into the entire blocks from the (3, 2, 1)-design.

Step 3: generate a new block design for neighbor discovery

Finally, we can get a $(v_a \times v_b, k_a \times k_b, \lambda_a \times \lambda_b)$ -design by combining A and B in this step. Figure 3 illustrates the completed (12, 6, 2)-design by combining the (3, 2, 1) and (4, 3, 2)-design.

By using the method in Step 2, we can construct a set of new neighbor discovery schedules (NDS) with a different duty cycle. These combined schedules make the neighbor discovery problem easy to solve. We can generate a new NDS by combining two block designs. If there exists a BIBD with a desired duty cycle, we can just use the BIBD for the neighbor discovery. However, due to the lack of a unified algorithm of generating BIBDs, it is not always possible to generate an NDS with a certain duty cycle. Therefore, the proposed block design scheme gives us more options. For instance, the proposed method can virtually construct a new block design with almost any duty cycle by selecting a proper set of previously known block designs. Figure 4 illustrates the algorithm we designed for combining two block designs.



| Slot | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|------|---|---|---|---|---|---|---|---|---|----|----|----|
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 2 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 3 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 4 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 5 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 6 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 8 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 9 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 10 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 11 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 |
| 12 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |

Fig. 3 Combination of (4,3,2) and (3,2,1) designs

Algorithm: Combination of Block Designs

```

1: procedure COMBINING TWO DESIGNS
2:    $\Phi := (v_1, k_1, \lambda_1)$ -BIBD
3:    $\Psi := (v_2, k_2, \lambda_2)$ -BIBD
4:    $\emptyset :=$  a sleep schedule
5:   for all  $p \in \Phi$  do
6:     replace  $p$  by  $\Psi$ 
7:   end for
8:   for all  $q \notin \Phi$  do
9:     replace  $p$  by  $\emptyset$ 
10:  end for
11: end procedure

```

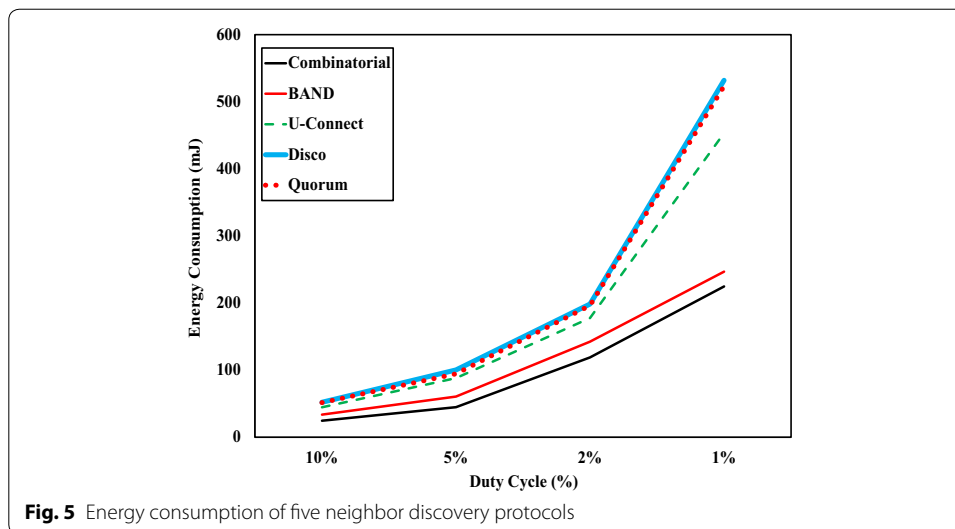
Fig. 4 An algorithm of combining two block designs

In the simulation study, we focused on two major criteria of neighbor discovery problems: energy consumption and discovery latency. Related studies of neighbor discovery also dealt with these metrics in their performance evaluation. We have implemented four representative neighbor discovery protocols in the literature and *BAND*. The performance of the proposed algorithm is much better than other competitive protocols. As shown in Fig. 5, the energy consumption of *BAND* is less than that of three competitive protocols. Note that *Combinatorial* is the optimal solution and the energy consumption of *BAND* is very close to that of *Combinatorial*.

Combination of human activity monitoring and neighbor discovery

We discussed three factors composing the core of HCC system. These factors focus on human abilities and limitations, social and cultural awareness, and adaptability across individuals and specific situations. From a human-centered perspective, one of the most important tasks toward an ideal HCC system is to understand people and society in which people live. In this section, we propose a potential computational model for supporting HCC systems. Based on the research directions that we have discussed in the beginning of this article, we suggest one possible model that can be used for understanding the behavior of individuals and social interactions.

First, we need a data collection mechanism in order to understand and analyze the human's behavior and the relationship with others. According to the purpose of the



system, there are many different types of options we can choose. One of the possible choices is the data of human physical activities. Our proposed human activity monitoring system can be applied to collect physical activity data. The monitoring system only requires a small amount of battery and memory space. It was originally developed for sensor applications, so the system is user-friendly and energy-efficient. Through the monitoring system, we could measure the level of physical daily activity of people.

Additionally, we might also collect the social interactions of people with our neighbor discovery protocol. In a wireless sensor network environment, in the beginning phase of the deployment of sensors, they try to find their neighbors and join the network in their communication range for a certain amount of time. Usually, a graph theory can be used for representing sensor nodes and their connectivity graphically. In the graph, each node illustrates a sensor and each vertex expresses a link between sensors. We could apply this concept to represent the interactions of people in communities. If our proposed neighbor discovery protocol finds neighbors frequently then it means that a specific person actively makes a relationship with others. The frequency of finding different neighbors can be used as a metric for the level of sociability of people.

Second, the level of physical activity and sociability of individuals can be used for developing evaluation metrics for HCC systems from a human-centered perspective. Our activity monitoring system can recognize typical human daily activities such as walking, running, standing, or sitting. It is possible that some active group of people usually has combination of running and walking activities. We could imagine that most businessmen or salesmen generally have some busy schedules almost every day. Especially, these type of people frequently meet other individuals in their daily life to make a social or business relationship with their partners. Therefore, we could define the degree of activeness of an individual based on analyzing different kind of physical activities through the proposed activity monitoring system. We could also design the degree of dynamics of social interactions using frequent data exchange for neighbor discovery. These two metrics can be employed for evaluating that a HCC system deeply considers the understanding of people.

Finally, let us think about a feasible application of combining the application-level solution and the network protocol-level approach. The proposed activity monitoring system can measure the level of physical daily activity of people. In addition, we could also apply our neighbor discovery protocol to quantify the level of sociability of people by checking how frequently the sensors detect their neighbors or how many neighbors they find within a certain amount of time. Let us make an assumption: the daily activity of a certain person consists of a lot of dynamic activities (walking or running) and the frequency or the number of neighbor discovery is also high then we could say that he or she is a sociable and an active person. The level of physical activity indicates how people live vigorously during their daily life. Frequent data exchange for neighbor discovery shows how actively people interact with others. Therefore, using the combination between physical activity monitoring and neighbor discovery, we can measure both individual health level and the level of socialized relationship with others.

There is one more possibility on the combination of human activity monitoring and neighbor discovery. We can conduct a correlation analysis between the level of physical daily activity and sociability of people. There are some meaningful hypotheses for better understanding of people. For example, we have the following questions?

- Does the people who has mostly dynamic activities show a vigorous relationship with their friends or partners?
- Does the people who actively interact with others have the high degree of activeness of their daily life?
- Is human daily activity closely related to human social interaction?

Conclusion

In conclusion, one of the most significant considerations of designing or implementing HCC systems is understanding people. That is why human beings should be the core component of the system in a HCC environment. In this paper, we propose a computational model quantifying the level of activeness of individuals and the level of social interactions. The combination of human activity monitoring and neighbor discovery could have some promising research works for better comprehension of people in the future. If there is a close relationship between individual physical activities and social interactions through the correlation analysis we suggest, we could adopt this relationship as a meaningful evaluation metric for HCC systems. In addition, by mixing the level of activeness and social interactions, we could characterize a number of groups of people. This clustering approach might give us a better understanding of people when each group indicates its own characteristic. This could be an opportunity of understanding the behavior or characteristic of a group of people.

Competing interests

The authors declare that they have no competing interests.

Received: 22 April 2016 Accepted: 4 May 2016

Published online: 05 July 2016

References

1. Jaimes A, G-Perez D, Sebe N, Huang TS (2007) Human-centered computing: toward a human revolution. *IEEE Comput* 5:30–34
2. Jaimes A, Sebe N, G-Perez D (2006) Human-centered computing: a multimedia perspective. In Proceedings of the 14th ACM international conference on multimedia. ACM, pp 855–864
3. Choi S, LeMay R, Youn J (2013) On-board Processing of acceleration data for real-time activity classification. In Proceedings of the 10th IEEE consumer Communications and networking conference (CCNC), pp 68–73
4. <http://www.shimmer-research.com>
5. Culhane KM, Lyons GM, Hilton D, Grace PA, Lyons D (2004) Long term mobility monitoring of older adults using accelerometers in a clinical environment. *Clin Rehabil* 18(3):335–343
6. Karantonis DM, Narayanan MR, Mathie M, Lovell NH, Celler BG (2006) Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. *IEEE Trans Inf Technol Biomed* 10(1):156–167
7. Lyons GM, Culhane KM, Hilton D, Grace PA, Lyons D (2005) A description of an accelerometer-based mobility monitoring technique. *Med Eng Phys* 27(6):497–504
8. Najafi B, Aminian K, Paraschiv-Ionescu A, Loew F, Bula CJ, Robert P (2003) Ambulatory system for human motion analysis using a kinematic sensor: monitoring of daily physical activity in the elderly. *IEEE Trans Biomed Eng* 50(6):711–723
9. Veltink PH, Bussmann HBJ, De Vries W, Martens WJL, Van Lummel RC (1996) Detection of static and dynamic activities using uniaxial accelerometers. *IEEE Trans Rehabil Eng* 4(4):375–385
10. Meng T, Wu F, Yang Z, Chen G, Vasilakos AV (2015) Spatial resuability-aware routing in multi-hop wireless networks. *IEEE Trans Comput* 65(1):244–255. doi:10.1109/TC.2015.2417543
11. Chen L, Li Y, Vasilakos AV (2016) Oblivious neighbor discovery for wireless devices with directional antennas. In Proceedings of the IEEE INFOCOM, San Francisco, 10–15 April 2016
12. Yao Y, Cao Q, Vasilakos AV (2015) EDAL: an energy-efficient, delay-aware, and lifetime-balancing data collection protocol for heterogeneous wireless sensor networks. *IEEE ACM Trans Netw* 23(3): 810–823
13. Bhuiyan MZA, Vasilakos AV (2015) Local area prediction-based mobile target tracking in wireless sensor networks. *IEEE Trans Comput* 64(7):1968–1982
14. Zhang XM, Zhang Y, Yan F, Vasilakos AV (2015) Interference-based topology control algorithm for delay-constrained mobile ad hoc networks. *IEEE Trans Mob Comput* 14(4): 742–754
15. Xiao Y, Peng M, Gibson J, Xie GG, Du D-Z, Vasilakos AV (2012) Tight performance bounds of multihop fair access for mac protocols in wireless sensor networks and underwater sensor networks. *IEEE Trans. Mob Comput* 11(10):1538–1554
16. McGlynn MJ, Borbash SA (2001) Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks. In Proceedings of the 2nd ACM international symposium on mobile ad hoc networking and computing, pp 137–145
17. Jiang J, Tseng Y, Hsu C, Lai T (2005) Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks. *Mob Netw Appl* 10(1–2):169–181
18. Tseng Y-C, Hsu C-S, Hsieh T-Y (2002) Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks. In Proceedings of twenty-first annual joint conference of the IEEE computer and communications societies, pp 200–209
19. Dutta P, Culler D (2008) Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications. In Proceedings of the 6th ACM conference on embedded network sensor systems, pp 71–84
20. Kandhalu A, Lakshmanan K, Rajkumar R (2010) U-connect: a low-latency energy-efficient asynchronous neighbor discovery protocol. In Proceedings of the 9th ACM/IEEE international conference on information processing in sensor networks, pp 350–361
21. Zheng R, Hou JC, Sha L (2003) Asynchronous wakeup for ad hoc networks. In Proceedings of the 4th ACM international symposium on mobile ad hoc networking and computing, pp 35–45
22. Bakht M, Trower M, Kravets RH (2012) Searchlight: Won't you be my neighbor. In Proceedings of the 18th annual international conference on mobile computing and networking, Istanbul
23. Meng T, Wu F, Chen G (2014) On designing neighbor discovery protocols: a code-based approach. In Proceedings of the IEEE conference on computer communications, Toronto, 27 April–2 May 2014
24. Chen L, Bian K, Zheng M (2014) Heterogeneous multi-channel neighbor discovery for mobile sensing applications: theoretical foundation and protocol Design. In Proceedings of the 15th ACM international symposium on mobile ad hoc networking and computing, Philadelphia, 11–14 August 2014
25. Choi S, Lee W, Song T, Youn J (2015) Block design-based asynchronous neighbor discovery protocol for wireless sensor networks. *J Sens* 2015:951652. doi:10.1155/2015/951652
26. Anderson I (1988) Combinatorial designs and tournaments. In Chapter 2, Oxford University Press, Oxford
27. Colbourn CJ, Dinitz JH (1996) The CRC handbook of combinatorial designs. CRC Press, Boca Raton
28. Stinson DR (2004) Combinatorial designs: constructions and analysis. Springer, Berlin
29. Godsil C (2010) Combinatorial design theory. <http://www.math.uwaterloo.ca/~kpurbhoo/co634/Designs.pdf>