ETRI: A Dynamic Packet Scheduling Algorithm for Wireless Sensor Networks

Shu Lei, S.Y. Lee, Yang Jie
Department of Computer Engineering
Kyung Hee University, Korea
{sl8132, sylee, yangjie}@oslab.khu.ac.kr

Abstract

How to efficiently use energy, enhance information quality, and improve transmission performance are three key issues of today's wireless sensor networks. Packet scheduling for wireless communication subsystem is one of the most important methods to achieve these issues. To our best knowledge, our paper is the first one to present the Two Tiers Butter model (Buffer of sensor networks and Buffer in sensor node) as the platform for the ***ETRI (Energy, Time, Reward, and Interest)*** *packet scheduling algorithm to filter the coming packets. By using ETRI packet scheduling principles, we can dynamically combine these four constraints to provide diverse scheduling versions for different sensor nodes' working environments and purposes, which can substantially improve the information quality and average performance of whole sensor networks.*

1. Introduction

Recently, with advances in low-power analog and digital electronic, compact and inexpensive battery-operated sensor units are emergently equipped with wireless communication and computation abilities. Due to the continuous declining of cost, a large amount of sensor nodes are possible to be deployed as sensor networks and used for scientific researches, such as battlefield monitoring and weather forecast. Since most of sensor nodes are battery-operated, the lifetime of sensor networks turns to be a critical factor considered by the system designers. Conventional researches on processor's scheduling algorithms, such as Dynamic Voltage Scaling have been fully utilized in all kinds of handheld devices and embedded systems. But researches on communication subsystems which have not been fully analyzed can still provide plentiful enhancement in terms of reducing wireless communication energy consumption. In addition to the task scheduling in real time operating systems which are based on the CPU's processing, extending the concept of DVS into the communication subsystems to provide a better packet scheduling can substantially reduce the energy consumption, and also improve the information quality and the performance of whole sensor networks.

Uniformed packet scheduling algorithm is obviously not suitable for heterogeneous sensor networks. In many sensor networks, heterogeneous sensor nodes are deployed in different working environments with different purposes and are equipped with different power resources, such as battery or solar cell. Different sensor nodes also have different physical topology positions in sensor networks, which lead to different logical positions in the hierarchical network structure, such as data source, cluster head, and base station. Sequentially, the performance requirement of different sensor nodes must be very different. For example, the performance of cluster head may be several times higher than that of data source. Otherwise, it may cause the bottleneck problem due to many packets are waiting for the relaying of cluster head. Therefore, each sensor node should have its corresponded packet-scheduling algorithm to achieve its special working purpose and meet its working requirement and constraints, so that the whole sensor network's performance can be guaranteed.

Providing high quality information to users is another important issue of sensor networks. Generally a huge amount of raw data can be created by a large sensor network and sent back to base station. However, in most of the time only the raw data of some sensor nodes that relate to the user's purposes are really valuable. In other words, only those data that users are interested in are useful. In order to improve the quality of requested information, **Interest** is introduced as a threshold to reduce the unnecessary data as well as improve the information quality.

Since large numbers of researches have proved that the communication of wireless channel is the main consumer of battery energy compared with that of the computation of processor. If we can reduce the receiving and transmitting times by refusing or discarding the unnecessary data, then we can substantially reduce the energy consumption.

Naturally, among a set of tasks of real time applications, some of them are more valuable than the others. Sometimes, instead of processing several unimportant tasks that just consume a small amount of energy, it is more meaningful to process one valuable task that will consume more energy. **Reward** is introduced for the first time to denote how important one task is by Cosmic Ruse, *et al.* in [1, 2, and 3]. Beside the Energy and Time constraints, Reward is considered as the third constraint for each task. In this paper, we simply name their algorithm as ETR (Energy, Time, and Reward) scheduling algorithm. The purpose of ETR scheduling algorithm is to maximize the system value (reward) while satisfying the time and energy constraints. In other words, whenever a new task is accepted, it must have the highest reward value among these tasks, and its energy consumption should not exceed the remaining energy.

In this paper we first present the Two Tiers Buffer (Buffer of sensor networks and Buffer in sensor node) model as the platform for energy aware packet scheduling in sensor networks. Then based on this Two Tiers Buffer model we present our new scheduling algorithm, which is named as **ETRI (Energy, Time, Reward,** and **Interest)** packet scheduling algorithm. Within this algorithm each packet has four parameters. They are (1) energy consumption of this packet, (2) deadline of this packet, (3) important level of this packet, and (4) interest level of this packet. Different from the ETR packet scheduling algorithm, we don't always use these four constraints at the same time, instead we dynamically combine these four constraints to filter and schedule packets for heterogeneous sensor nodes and diverse working purposes. By using ETRI packet scheduling algorithm with the Two Tiers Buffer model, we can achieve the following contributions: (1) providing threshold to control the coming packets to reduce the energy consumption; (2) providing diverse packet scheduling algorithms for heterogeneous sensor nodes to improve the whole network's performance; (3) using stricter constraints to provide higher quality information.

In next section, we present the related work. In section 3, we present the Two Tiers Buffer model and six basic scheduling algorithms. In section 4 we present the ETRI scheduling algorithms. We present the simulation work in section 5. Future work and challenge issues are presented in section 6.Finally, we conclude in section 7.

2. Related Work

Since efficient-using energy, enhancing information quality and improving average performance are critical issues of battery-operated wireless sensor networks; many related researches have been done to address these problems by using different approaches. As we have introduced, Cosmic Ruse, *et al.* first time presented the ETR real-time scheduling approach in [1, 2, and 3] to maximize the system value while satisfying the time and energy constraints. But the ETR scheduling algorithm is not able to control the original coming packets. In other words, it is not able to filter packets that users are not interested in. But in our ETRI algorithm, we provide **Interest** as one more constraint. When we use ETRI to process packets, we can choose the packets interested first, and then among these filtered packets, we can schedule them basing on their reward value while still satisfying the time and energy constraints. In [4, 5] Curt Chargers, *et al.* also first time proposed the *Dynamic Modulation Scaling* that should be applied to the communication subsystems as a power management approach, which is the tradeoff between the energy consumption and the transmission delay. If the communication subsystem sends or receives data with a higher frequency, then more energy will be consumed. Therefore, the key idea of this approach is that it always transmits packet with the lowest frequency but still meet the deadline. The drawback of this approach is that it just considers the packets that already existed in the buffer, but does not provide the threshold to reduce the coming packets. In [6, 7, and 8], the date-centric approach is proposed for energy-efficient data routing, gathering and aggregation in wireless sensor networks. The key idea is that whenever users query some data from the sensor networks, they just query the data they are interested in. Moreover, by using the fusion circuit, several packets which have the similar information can be fused into one packet to reduce the packet number. Once the number of packets is reduced, sequentially, the energy consumption will be reduced. In their researches they simply consider all these packets that are of same importance, but actually among these interested packets, some of them may be more important than the others. For example, users are interested in the data of several sensor nodes used to monitor an object. The data created by the sensor nodes which are close to the observed object have more valuable information than the data created by the sensor nodes which are far from the observed object. Therefore, if we can introduce the **Reward** into these interested packets, we are able to select out and process the most important and valuable packet first.

From these related researches, we find that a comprehensive packet scheduling algorithm is really necessary for wireless sensor networks. Before we introduce the ETRI packet scheduling algorithm, we first introduce the Two Tiers Buffer model to readers in the next section.

3. Two Tiers Buffer in sensor network

In wireless sensor networks, generally, a huge amount of data are collected by different sensor nodes and transmitted to the base station through the wireless communication. Different packets that are routed in sensor networks may have significantly different characteristics, such as the packet size, execution time, transmission time, energy consumption, etc. As figure 1 shows that three different sensor nodes A, B, and C are sending different packets to the **Analyzed Sensor Node (ASN)** simultaneously. Then these packets are sequentially forwarded to the cluster head and base station. In terms of the ASN or cluster head, many unprocessed packets are still physically located in different sensor nodes and waiting for the processing of ASN or cluster head. Therefore, in sensor networks, all the sensor nodes which are going to send packets to the cluster head or ASN can be logically considered as a buffer, except the cluster head or ASN. Since all of these packets are waiting for the processing of cluster head or ASN. We regard this buffer as the **First Tier Buffer (FTB)**. Actually the FTB is a logical concept for cluster

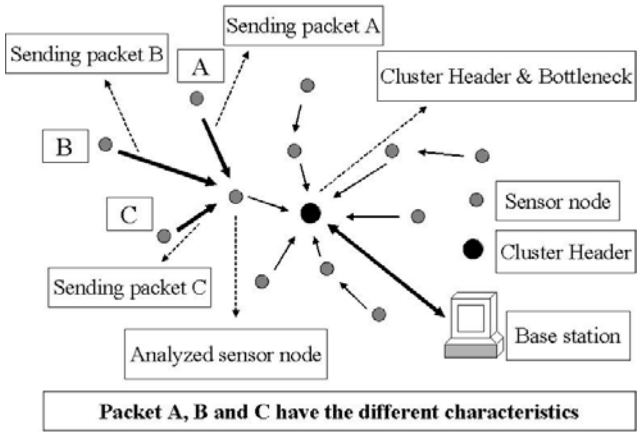

Figure 1. Different sensor nodes send different packet to one sensor node simultaneously

head or ASN. The **Second Tier Buffer** (**STB**) is the buffer that physically exists inside sensor nodes and cluster head. Since many sensor nodes are sending packets to the ASN and cluster head, ASN and cluster head need buffers to store these received packets. Therefore, we propose the Two Tiers Buffer model for wireless sensor network as the figure 2 shows. For the FTB, different packets which are physically located in different sensor nodes and are waiting for the processing of the ASN or cluster head can be logically considered stored inside the FTB. For the STB, several different packets which have been received and stored inside the buffer are waiting for the processing of processor or radio.

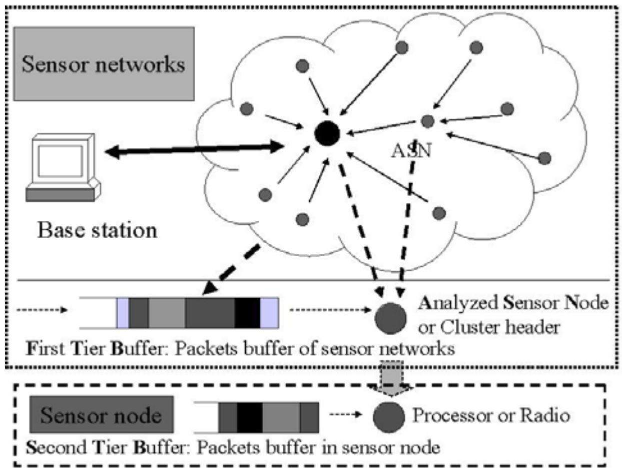

Figure 2. Two tiers buffer

After introducing the Two Tiers Buffer model, we can clearly define the mission of each buffer. In FTB, we base on different purposes and principles to filter and accept packets. And in STB, we base on different purposes and technologies to process or transmit packets.

In FTB, we have the following three algorithms to accept packets or tasks:

1) Maximizing total packet number

Different packets are flying on sensor network with different nature energy consumption characteristics. In order to maximize the total processed packet number for a fixed amount of energy; sensor nodes always accept the packet that has the smallest energy consumption first. This algorithm is suitable for some situation when the total energy consumption of flying packets is much larger than the remaining energy of whole sensor network which often happens in some overload systems.

2) Maximizing reward value

The key idea of this algorithm is that instead of processing two or more unimportant packets which just consume a small amount of energy, we would like to process one important packet which may consume relatively larger amount of energy. Reward value is used to denote the important level of packet. A packet with a larger reward value means that this packet is more important. Therefore, the sensor nodes always accept the packets which have the highest reward value. Thus, we can guarantee that the most important packets can be processed first.

3) Maximizing interest value

In sensor networks' data gathering and aggregation, sensor nodes can distinguish packets based on their content and will not accept those packets that they are not interested in. Interest value is used to denote the interest level of packet. A packet with a larger interest value means that this packet is more interesting. In addition, the packet that has the higher interest value can be accepted first. The more the packets are accepted, the better information quality will be obtained. By reducing the incoming packet number, but still providing enough information for applications, we can substantially reduce the total energy consumption.

In STB, we have the following three algorithms to process packets or tasks:

4) Maximizing system processing performance

By using this algorithm, sensor nodes will process the incoming packets as fast as possible, thus, designers just consider the system execution performance, without worrying about the energy consumption. In other sense, sensor nodes always minimize the processing time of each packet.

5) Maximizing system lifetime [10]

This algorithm's idea is totally opposite to the No.4th algorithm. System designers compromise the energy consumption and execution delay. Whenever sensor node processes the incoming packets, it always considers the energy as the first class resource, and slows down the processing speed in order to just meet the deadline. Typically, Dynamic Modulation Scaling is used for this purpose.

6) Maximizing parallel performance [11, 12, 13]

In terms of rechargeable sensor nodes that equipped with solar batteries, they may transmit several packets simultaneously. The remaining energy of sensor nodes is not enough to support all packets with high transmission rate. But if we slow down the transmitting speed, we can reduce the energy consumption. At the same time, we can have longer time to let the solar battery create more energy. Therefore, the key idea is that sensor nodes slow down the performance while using the saving energy and new created energy to support more packets.

Another aspect in STB is the packet discarding principle. Corresponded to the foregoing three algorithms in FTB, we have the following three discarding principles respectively: 1) always discard the packet that has the largest energy consumption; 2) always discard the packet that has the smallest reward value first; 3) always discard the packet that has the smallest interest value. In next section we introduce ETRI algorithm based on this Two Tiers Buffer model and six basic algorithms.

4. ETRI packet-scheduling algorithms

As we have proposed the ETRI packet-scheduling algorithm in section one, the principles of ETRI are as following:

(1) Whenever a new packet is accepted, its energy consumption should not exceed the remaining energy;

(2) Whenever a packet is processed, it must meet its deadline;

(3) Every packet can under Energy, Timing, Reward, and Interest constraints simultaneously;

(4) It is not necessary to always under these four constraints at the same time;

(5) We can dynamically combine these constraints to filter and schedule packet for heterogeneous sensor nodes and divers working purposes.

Therefore, we dynamically combine these six foregoing basic algorithms in FTB and STB. By using permutation and combination for FTB, we can provide seven different combinations. And each time we can only choose one approach from the STB. So finally we can have twenty one versions. In the following paragraph, we present eight versions of them as examples. Some of these versions actually have already existed in other researchers' former works, such as the following ETRI version one, two and three. And ETRI version four & five are similar with the REW-Pack algorithm and REW-Unpack algorithm that have been presented by Cosmin Rusu, *et al.* in [1]. The ETRI version seven and eight are new algorithms that are generated by using the principles of our ETRI packet-scheduling algorithm. In figure 3 and 4, we give the flowchart and source code of ETRI version seven as the example.

ETRI version one:
FTB: Empty
STB: Maximizing system processing performance

ETRI V-1: The ASN doesn't check any received packet. It just simply receives packets and relays them. Once it gets a packet, it will process this packet as fast as possible. The aim of ETRI V-1 is to maximize the system performance.

ETRI version two:
FTB: Empty
STB: Maximizing system lifetime

ETRI V-2: The ASN doesn't check any received packet. It just simply receives packets and relays them. Once it gets a packet, it will always process this packet just meet its deadline. The aim of ETRI V-2 is to reduce the energy consumption.

ETRI version three:
FTB: Maximizing interest value
STB: Maximizing system lifetime

ETRI V-3: The ASN always accepts the packet that has the largest interest value among several checked packets. Once it gets a packet, it will always process this packet just meet its deadline. The aim of ETRI V-3 is to maximize the interest value and system lifetime.

ETRI version four:
FTB: Maximizing reward value
STB: Maximizing system lifetime

ETRI V-4: The ASN always accepts the packet that has the largest reward value among several checked packets. Once it gets a packet, it will always process this packet just meet its deadline. The aim of ETRI V-4 is to maximize the reward value and system lifetime.

ETRI version five:
FTB: Maximizing reward value
STB: Maximizing system processing performance

ETRI V-5: The ASN always accepts the packet that has the largest reward value among several checked packets. Once it gets a packet, it will always process this packet as fast as possible. The aim of ETRI V-5 is to maximize the reward value and system performance.

ETRI version six:
FTB: Maximizing interest value
STB: Maximizing system processing performance

ETRI V-6: The ASN always accepts the packet that has the largest interest value among several checked packets. Once it gets a packet, it will always process this packet as fast as possible. The aim of ETRI V-6 is to maximize the interest value and system performance.

ETRI version seven:
FTB: Step one: Pass interest threshold
Step two: Maximizing reward value
STB: Maximizing system lifetime

ETRI V-7: The ASN accepts the packets, of which interest values are over the threshold, among several checked packets. Within these accepted packets, ASN will always choose the packet that has the largest reward value first and process it just meet its deadline. The aim of ETRI V-7 is to guarantee the interest value, maximize the reward value and system lifetime.

ETRI version eight:
FTB: Step one: Pass interest threshold
Step two: Pass reward threshold
Step three: Maximizing total packet number
STB: Maximizing system lifetime

ETRI V-8: The ASN accepts the packets, of which interest value and reward value are over the threshold. Within these accepted packets, ASN will always choose the packet that has the least energy consumption first and process it just meet its deadline. The aim of ETRI V-8 is to guarantee the interest value and reward value, maximize the processed packet number and system lifetime.

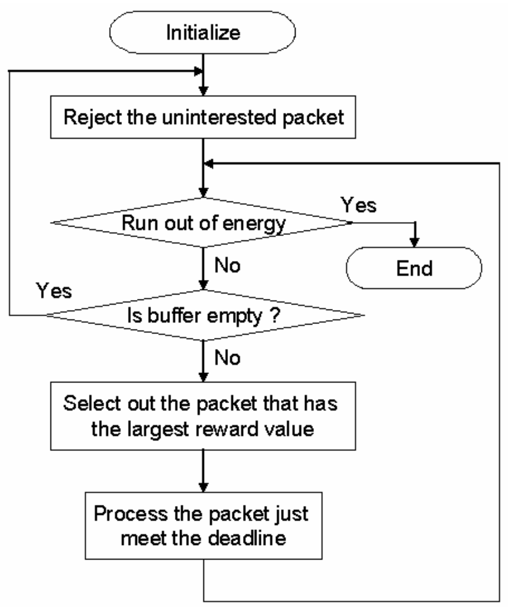

Figure 3. Example: flowchart of ETRI V-7

In next section, we will present our simulation environment as well as approach. And we compare our new algorithms with these foregoing existing algorithms based on simulation results.

5. Simulation and discussion

In our simulation, we randomly deploy nine different sensor nodes. And we randomly initialize these sensor nodes with: the total energy of sensor nodes (scope: from 111 to 888), the buffer size of sensor nodes (scope: from 6 to 9). In addition, we design 8 different packets that are randomly initialized with the following four parameters: energy consumption (scope: from 3 to 10), deadline, reward value (scope: from 3 to 10) and interest value (scope: from 3 to 10). Eight of these nine sensor nodes are chose to be the packet generators which randomly create these eight different packets and send to the remaining one. This remaining one worked as the ASN (cluster head). For this ASN we design four parameters: the total energy of ASN

```
For i = 1 To cluster.buffer
If packet(i).interest < packet(interest_threshold).interest
  Then
     Do_not_accept_this_packet()
End If
Next
While continue
For j = 1 To cluster.buffer
 If pt(j) <> 0 Then  'If cluster buffer is not empty,
       p_b = pt(C_pt)    'then pick out one of them
      tracker = C_pt
  For it = 1 To cluster.buffer
  'Find out the packet that has the largest interest value
  If packet(pt(it)).interest >= packet(p_b).interest Then
         p_b = pt(it)
         tracker = it
   End If
  Next
  For it = 1 To cluster.buffer
   If packet(pt(it)).interest = packet(p_b).interest Then
    'Maximize the reward value
     If packet(pt(it)).reward > packet(p_b).reward Then
       p_b = pt(it)
       tracker = it
     End If
    End If
   Next
    Packet_processing()
    If sum >= cluster.energy Then
    'ASN run out of energy
        continue = False
    End If
 Else
    continue = False
 End If
Next
End While
```

Figure 4. Example: source code of ETRI V-7

(scope: from 111 to 888), the buffer size of ASN (scope: from 6 to 9), the interest threshold and the reward threshold. And we also design that this ASN can work in two kinds of working models: 1) processing packets as fast as possible; 2) always compromising the energy consumption and time delay. In terms of energy consumption, we mainly consider the following two parts that have strong relationship with our proposed packet scheduling algorithm, which are the processing energy $\{E_{(\text{Returning ACK})} + E_{(\text{Receiving packet})} + E_{(\text{Processing})} + E_{(\text{Broadcasting event})} + E_{(\text{Listening})} + E_{(\text{Accepting ACK})} + E_{(\text{Sending packet})}\}$ and the checking energy $\{E_{(\text{Accepting event})} + E_{(\text{Deciding})}\}$. And the checking energy is designed to be 0.3, which is 10% of the minimum packet consumption 3. We designed the simulation parameters as following: 1) energy utilization, 2) checking energy, 3) processing energy, 4) processed packet number, 5) lifetime of ASN, 6) total process interest value, 7) total process reward value, 8) average interest value of each packet, 9) average reward value of each packet.

These foregoing 8 versions ETRI packet algorithms are executed on this ASN to get the simulation result. The simulated ASN has the parameters as the table 1 shows.

ASN:	Total energy	Buffer size	Interest threshold	Reward threshold
Parameter	666	6	5	5

Table 1: The analyzed sensor node

The meaning of threshold is that we just accept the packets when their interest value and reward value are belonging to the top 5 among these 8 different packets. The simulation result as the following tables:

	ETRI V-1	ETRI V-2	ETRI V-3	ETRI V-4
Energy utilization	93.94%	93.04%	92.29%	91.69%
Checking energy	40.2	45.9	51	54.6
Processing energy	621	615	611	607
Processed packets	83	**96**	88	92
Lifetime of ASN	**2890**	**10770**	**11290**	**10850**
Total interest	528	616	**656**	567
Total reward	433	501	478	**531**
Average interest	6.36	6.41	**7.45**	6.16
Average reward	5.21	5.21	5.43	**5.77**
As fast as possible	✓	×	×	×
Energy & delay	×	✓	✓	✓

Table 2: ETRI V-(1-4) simulation result

From the simulation result of table 2, we can find out that by using the ETRI V-1, ASN's lifetime is much shorter than others. But in other words, the shorter lifetime means the higher performance, because the ASN's energy is fixed. Compared with the ETRI V-1, the ETRI V-2 can give the ASN longer lifetime and let it process more packets. By using the ETRI V-3 and V-4 the ASN can have relatively higher totally processed interest value and reward value. The average interest value and reward value of each packet are also relatively higher than other versions. Through these highlighted simulation results we can find out that these algorithms can achieve their original purposes well.

From the simulation result of table 3, we can find out that the ETRI V-5 and ETRI V-6 have relatively higher performance than others. The bottleneck problem that usually happens on the cluster head also can be well addressed by using ETRI V-5 or ETRI V-6. Whenever one sensor node changes its working

	ETRI V-5	ETRI V-6	ETRI V-7	ETRI V-8
Energy utilization	92.37%	93.03%	**87.61%**	**87.08%**
Checking energy	48.3	45.9	**81.6**	**85.5**
Processing energy	613	614	580	573
Processed packets	79	78	72	**73**
Lifetime of ASN	**2970**	**3020**	**11640**	**11350**
Total interest	510	575	**659**	**663**
Total reward	466	414	422	385
Average interest	6.45	**7.37**	**9.15**	**9.10**
Average reward	**5.89**	5.30	**5.86**	**5.27**
As fast as possible	✓	✓	×	×
Energy & delay	×	×	✓	✓

Table 3: ETRI V-(5-8) simulation result

role into the cluster header, it can change its original energy saving packet scheduling algorithm to a new packet scheduling algorithm which emphasizes on processing performance. For example, we can change the algorithm from ETRI V-2 to ETRI V-6 to address the bottleneck problem as well as improve the performance of whole sensor networks. By using ETRI V-7 and ETRI V-8, the cluster header can have longer lifetime and provide relatively higher total processed interest value and reward value as well as the average interest value and reward value. But the problem of ETRI V-7 and ETRI V-8 is that their checking energy is nearly two times larger than other ETRI versions. The reason is that we give more constraints in these two versions. As a result, the ETRI V-7 and ETRI V-8's energy utilization is relatively lower than that of other versions, but the distance is not too much. Actually, in this simulation we design the checking energy is 10% of the minimum packet consumption 3. If the packet processing energy is significant larger than the checking energy, for example the checking energy is 5% of the minimum packet consumption, we can have higher energy utilization. We also can conclude from this simulation that if we give more constraints in the FTB, we will have to pay more energy consumption for checking these stricter requirements. That is also the tradeoff of between energy consumption and quality of information.

6. Future work and challenge issues

In this paper, we mention the ETRI scheduling principles that sensor nodes can know the reward value and interest value of packets well. In the simulation we randomly design the interest value and reward value for 8 different packets to test the performance of our new algorithms and compare them with existing research works. But we do not mention the method that how to design the reward value and interest value for different packets based on the each packet's content. Therefore, as a challenge issue to be solved in the future, we are going to explore the appropriate measure methods to evaluate the interest level and important level of different packets. Another critical issue raised by this paper is that sensor nodes should be working-role-aware and query-aware so that they can dynamically change their packet scheduling algorithms to optimize the energy-utilization and processing-performance. In other words, to provide better flexibility for current existing packet scheduling algorithms which are used for communication subsystems will be one of the key issues for future research.

7. Conclusion

Battery-operated heterogeneous sensor network should have a meaningful lifetime and can provide high quality information to users. Packet scheduling algorithm for communication subsystems is a potential approach to accomplish these critical issues. On the other hand, different sensor nodes in the sensor networks should have different packet scheduling approaches to correspond to their real working environments and purposes. In this paper we present the Two Ties Buffer model and ETRI packet scheduling algorithm. Reward and Interest are added as the new constraints to the conventional timing and energy constraints for packet scheduling algorithms. The key idea is that instead of using one uniform packet scheduling algorithm for the whole sensor networks, we can dynamically combine our ETRI packet-scheduling algorithms to adapt to different sensor nodes' real working purposes. By using the Two Ties Buffer model and ETRI packet scheduling algorithms, we can easily utilize different ETRI versions to different sensor nodes to reduce energy consumption, enhance information quality as well as the performance of sensor networks.

REFERENCES

[1] Cosmin Rusu, Rami Melhem, Daniel Mosse, "Maximizing the System Value while Satisfying Time and Energy Constraints", IBM Journal of R&D, vol 47, no 5/6, 2003.

[2] Cosmin Rusu, Rami Melhem, Daniel Mosse, "Maximizing Rewards for Real-Time Applications with Energy Constraints", ACM TECS, vol 2, no 4, 2003.

[3] Cosmin Rusu, Rami Melhem, Daniel Mosse, "Multi-version Scheduling in Rechargeable Energy-aware Real-time Systems", to appear in Journal of Embedded Computing, 2004.

[4] Curt Schurgers, Vijay Raghunathan, Mani B. Srivastava, "Modulation Scaling for Real-time Energy Aware Packet Scheduling", *Global Communications Conference (GlobeCom'01)*, San Antonio, Texas, pp. 3653-3657, November 25-29, 2001.

[5] Curt Schurges, Olivier Aberthorne, Mani B. Srivastava, "Modulation Scaling for Energy-Aware Communication Systems", *International Symposium on Low Power Electronics and Design (ISLPED'01)*, Huntington Beach, CA, pp. 96-99, August 6-7, 2001.

[6] Bhaskar Krishnamachari, Deborah Estrin, Stephen Wicker, "Modeling Data-Centric Routing in Wireless Sensor Networks", proceedings of 6th international workshop on Modeling analysis and simulation of wireless and mobile systems, 2003.

[7] Huseyin Ozgur Tan and Ibrahim Korpeoglu, "Power Efficient Data Gathering and Aggregation in Wireless Sensor Networks", SIGMOD/PODS volume 32, number 4, December 2003

[8] Jonathan Beaver, Mohamed A. Sharaf, Alexandros Labrinidis, Panos K. Chrysanthis, "Power-Aware In-Network Query Processing for Sensor Data", In the Proceedings of the 2nd Hellenic Data Management Symposium, September 2003.

[9] Andrej Bogdanov, Elitza Maneva, Samantha Riesenfeld, "Power-Aware Base Station for Sensor Networks", the 23rd Conference of IEEE Communication Society, 2004.

[10] Konstantions Kalpakis, Koustuv Dasgupta, and Parag Namjoshi, "Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks", in To Appear in Proceedings of IEEE Networks'02 Conference, 2002.

[11] Jinfeng Liu, Pai H. Chou, Nader Bagherzadeh, Fadi Kurdahi, "A Constraint-based Application Model and Scheduling Techniques for Power-aware Systems", Proceedings of the 9th International Symposium on Hardware/Software Codesign, pages 153-158, Copenhagen, Denmark, April 2001.

[12] Jinfeng Liu, Pai H. Chou, Nader Bagherzadeh, Fadi Kurdahi, "Power-Aware Scheduling under Time Constraints for Mission-Critical Embedded Systems", Proceedings of the 38th Design Automation Conference, pages 840-845, Las Vegas, NV, USA, June 2001.

[13] Jinfeng Liu, Pai H. Chou, and Nader Bagherzadeh, "Power-Aware Task Motion for Enhancing Dynamic Range of Embedded Systems with Renewable Energy Sources", Proceedings of the 2nd Workshop on Power-Aware Computing Systems, Cambridge, MA, USA, Feb. 2002.