Investigating the Energy Cost of Control Packet in Hybrid MAC

Qian Dong
Department of Computational Engineering
Dresden University of Technology
01/09/2009
Abstract

A wireless sensor network is composed of a large number of distributed sensor nodes that are densely deployed either inside the phenomenon or very close to it. Each node senses data periodically in their sensing filed and transmits the sampling data through a multi-hop manner until the data finally reaches the sink. It’s very difficult to change or recharge batteries for these battery-supported sensor nodes, so the first thing we should consider is the energy efficiency when designing a good MAC protocol for wireless sensor networks. Control packet, being as one of the major sources of the energy waste, consumes significant amount of energy which however does not directly benefit the real data transmission. However, in the other side, control packet is a good method to avoid some problems such as collision and overhearing, which are in turn another two major sources of the energy waste. Therefore, whether to apply control packet or not in the medium access mechanism of the MAC protocol is a dilemma appearing in front of us.

In this thesis, we will solve this difficulty by making analysis of the tradeoff between the extra energy consumption introduced if control packet is activated and the extra energy consumption brought if control packet is deactivated by setting up a comprehensive and accurate analytic energy model in both cases based on the SMAC protocol, in the assumption of finite number of sensor nodes, ideal channel conditions and infrequent communications among sensor nodes. Furthermore, Matlab is used to present the visual relationship between the extra energy consumption and the sampling rate in both situations through simulation figures, based on which we are able to make the decision whether or not to adopt control packet in the SMAC protocol.
# Contents

1 Introduction .................................................................................................................. 1
2 Related Work .................................................................................................................. 3
3 Network Model ............................................................................................................... 4
   3.1 Sensing task ............................................................................................................. 4
   3.2 System deployment ................................................................................................. 5
   3.3 Network topology ................................................................................................. 5
   3.4 Coverage ................................................................................................................. 6
   3.5 Connectivity ............................................................................................................. 6
4 MAC Protocol ................................................................................................................. 6
   4.1 Protocol selection criteria ....................................................................................... 7
   4.2 SMAC protocol ...................................................................................................... 8
5 Energy Model of Arising Consequences with Control Packet .................................... 13
   5.1 Features of energy model ...................................................................................... 13
   5.2 Collision probability model .................................................................................. 14
      5.21 Collision probability in saturated period ......................................................... 16
      5.22 Collision probability in unsaturated period .................................................... 18
      5.221 Active neighbor model .................................................................................. 18
      5.222 T model .......................................................................................................... 25
      5.223 Collision probability model .......................................................................... 27
      5.23 Overall collision probability model ................................................................. 31
   5.3 Overlap neighbor model ......................................................................................... 42
   5.4 Energy model of extra control packet RTS and CTS ............................................. 44
   5.5 Energy model of extra overhearing of RTS and CTS ............................................. 49
   5.6 Energy model of extra sleep time ......................................................................... 51
   5.7 Energy model of the extra radio mode switching .................................................. 54
6 Energy Model of Arising Consequences Without Control Packet .......................... 61
   6.1 Collision probability model .................................................................................. 63
      6.11 In saturated period ........................................................................................... 64
      6.12 In unsaturated period ....................................................................................... 64
      6.121 Active neighbor node model ........................................................................... 65
      6.122 T model .......................................................................................................... 66
      6.123 Collision probability model .......................................................................... 68
      6.13 Overall collision probability model ................................................................. 71
   6.2 Energy model of the first data fragment retransmission due to collision ............... 79
   6.3 Energy model of the extra overhearing of the first data fragment and ACK frame ... 85
   6.4 Energy model of the extra idle listening ............................................................... 89
   6.5 Energy model of the extra radio mode switching .................................................. 94
7 Methodology and Simulation ....................................................................................... 98
   7.1 Overall collision probability vs. sampling rate with control packet ................... 100
   7.2 Overall collision probability vs. sampling rate without control packet ............... 111
   7.3 Extra energy consumption vs. sampling rate with control packet ...................... 121
   7.4 Extra energy consumption vs. sampling rate without control packet ................ 122
7.5 Comparison between the energy consumption with and without control packet ..........124
8 Conclusion and Future Work ..................................................................................125
1 Introduction

The wireless sensor network is considered as one of the popular technologies of the future which has a wide range of application areas such as health systems, military explorations and habitat monitoring. Such a network normally consists of a large number of distributed nodes with sensing, computation, wireless communication and in-network processing capabilities. These tiny nodes are densely and predeterminedly located either inside a particular phenomenon or very close to it. Each sensor node has one or more sensors which sense the raw data in their sensing filed periodically. Instead of transmitting the raw data directly to their one-hop neighbor, each node firstly makes use of their in-network processing ability to locally carry out simple computations and routes only the required and interesting processed data hop by hop until the sink receives. In this way, a significant amount of the network data traffic can be reduced. The embedded low-power radios enable the wireless communication between any two one-hop sensor nodes. Once the radio is turned on, the energy of nodes are being consumed. Since all sensor nodes are battery operated, energy awareness is an essential design issue when we plan to set up sensor network protocols and algorithms.

Like in all shared-medium networks, medium access control (MAC) is an important technique which establishes communication link for data transfer and provides fair and efficient channel access control mechanisms that enables sensor nodes to communicate within a multi-hop self-organizing wireless network. Many MAC protocols have been developed so far such as time division multiple access (TDMA), carrier sense multiple access (CSMA) [1], and contention-based protocols like IEEE 802.11 [2], SMAC [3], and TMAC [4].

There are several attributes which determine the performance of MAC protocols. The most important one that we firstly consider is energy efficiency. As stated before, all sensor nodes are normally battery supported, and it’s difficult to change or recharge the batteries for these nodes. So we should try our best to keep the sources of the energy waste as less as possible. In fact, there are four major sources of energy waste in wireless sensor networks. The first one is collision. When two or more packets sent from different senders are received by one and the same receiver at the same moment, these packets are corrupted. This usually happens because of the broadcasting property of the radio transmission. After corruption, these packets need to be retransmitted which leads to the increase of the energy consumption. The second source is overhearing. This means a node receives packets which are destined to others. Overhearing is energy consuming since receiving useless information is a waste. The third source is idle listening. This means a node’s radio is on while does neither transmit nor receive. Idle listening is the major source of the energy waste. The last but not the least is control packet (RTS, CTS and ACK) overhead. Sending, receiving and listening control packet consumes energy though it benefits data transfer to some extent. To design a good MAC protocol, there are some other important attributes we must consider as well, such as latency, network lifetime, scalability, throughput, fairness and so forth. Which attributes are more important than others depends on which MAC protocols we choose. For example, for traditional protocols such as Bluetooth and MANET, latency, throughput, maintenance and high quality of service (QoS) should be put in the first place. Not alike, for the
IEEE 802.11 and SMAC protocol, the energy consumption is the biggest issue. Nevertheless, they have some different attributes. IEEE 802.11 aims at promoting the per-node fairness. However, the application-level fairness is more preferable than per-node fairness for SMAC where all the sensor nodes coordinate their effects to perform for the same purpose.

Our analytical model is built up based on the SMAC protocol. In most sensor network applications, individual nodes remain idle for a long period of time and become suddenly active as interesting events are detected. This characteristic of sensor networks motivates SMAC different from the traditional wireless MAC protocols such as IEEE 802.11 in many aspects: energy conservation and self-configuration are primary goals while per-node fairness and latency are less important. SMAC uses three novel techniques to reduce energy depletion in the meantime support self-configuration. The first technique is the introduction of the periodic listen and sleep. Given the fact that the sampling rate is very low comparing with the transmission rate, it is unnecessary to keep nodes listening all the time. So SMAC asks nodes to turn off their radios and go to sleep mode for some time, and then wake up to transmit the accumulated data packets which are sampled during their previous sleep phases. In order to reduce control packet overhead, neighboring nodes form virtual clusters to auto-synchronize on sleep schedules. The second technique is the prevention of the overhearing. All the neighbor nodes will turn off their radios during the communications between the transmitter and the receiver. The last optimization is the adoption of the message passing mechanism so as to reduce the contention latency as well as the retransmission cost for sensor network applications, which requires the capability of store-and-forward processing when data is transmitted through the network.

Deserving to be mentioned, different from the medium access mechanism of the SMAC protocol, the RTS and CTS frame in the four-way handshaking mechanism of the 802.11 protocol does not have the capability of the overhearing avoidance, since each node keeps listening to all the transmission of its neighbors so as to perform the effective virtual carrier sense. The NAV vector inside both the RTS frame transmitted by the transmitter and the CTS frame sent out by the receiver could only delay their neighbor nodes’ transmission, rather than let them go to sleep.

In this article, we carefully analyze the relationship between the sampling rate and the energy cost associated with control packet in the wireless sensor networks based on SMAC. On one side, if the sampling rate is very small comparing with the transmission rate, the probability of the collision occurring during the unsaturated period of the listen phase will be very small. As a consequence, the energy spent on the retransmission caused by collision will be very small. Thus, from this perspective, maybe the introduction of control packet is a totally unnecessary move, since the energy it reduces might be less than the energy it brings. On the other side, if the sampling rate is large, deactivating the control packet may lead to several negative consequences such as the increase of the collision probability which results in the increase of the retransmission cost and the appearance of the hidden terminal problem. Therefore, whether we should adopt control packet at all when designing a good MAC protocol makes us rather confused. So in this thesis, motivating to successfully find out the exact answer, we will closely analyze the trade off between the extra energy consumption introduced if control packet is activated and the extra energy consumption brought if control packet is deactivated by setting up an accurate and
comprehensive analytical model based on the SMAC protocol, in the assumption of finite number of sensor nodes, ideal channel conditions and infrequent communication among sensor nodes. By means of the comparison through simulation figures with the help of Matlab, version 7.0.1, we could eventually make our decision on whether to use control packet or not.

The rest of this thesis is organized as follows: in section 2, we introduce the related work. In section 3, the network model will be briefly described. In section 4, we will show the choice criteria of the MAC protocols and elaborate on the working principle of the SMAC protocol. In section 5 and 6, we will set up a comprehensive and accurate analytical model of the energy consumption based on the SMAC protocol where the analysis is classified into two parts, one of which is operated with control packet while the other one is investigated without control packet. In the following section 7 we will use Matlab to show the visualized result of the energy consumption tradeoff by several figures and discuss some observations. Finally in section 8, we will summarize the conclusion and remark on the future work.

2 Related Work

Several researches on the collision probability in the saturated situation based on several MAC layer protocols have been proposed for wireless sensor networks in the recent past in order to evaluate the maximum throughput that systems are able to carry in stable conditions. Bianchi [2] provides a simple but extremely accurate analytical model to compute the asymptotic throughput performance of 802.11 DCF based on the basic access mechanism by firstly introducing a Markov Chain model for the backoff window size to set up the saturated collision probability model in the assumption of finite number of sensor nodes and ideal channel conditions.

Wu, Xia, Zhang et al. [5] propose a collision model for a random access and random spreading direct-sequence code-division multiple access (DS-CDMA) network where collision is defined as the situation where it is impossible to guarantee the SINR requirement for all concurrent transmissions through power control. Such collision model presents one concrete example of multi-packet reception with spread spectrum in the assumption that none of the active users are able to successfully deliver its packet when collision occurs.

Ning [6] gives a box-ball model to resolve the packet collision probability in order to analyze the power-consumption performance of 802.11 DCF in a saturated WLAN environment based on the conception of virtual transmission period and average contention window in the assumption of limited retransmissions of each packet and finite-state one-dimension Markovian process. The result shows that the power consumption performance is mainly affected by the receiving power of nodes rather than the transmitting power, the packet length and the channel rate.

Vu and Sakurai [7] study the impact of channel capture following a busy period on collision probabilities in a saturated IEEE 802.11 network by setting up a new analytical model based on a simpler mean value approach that utilizes the channel access probability conditioned upon the
status of the channel from the previous time slot.

Tay and Chua [8] use an analytic model to study the channel capacity using the basic access method in the IEEE 802.11 protocol. It provides closed-form approximations for the collision probability, the maximum throughput and the limit on the number of stations in a wireless cell.

In addition, energy conservation is important for wireless sensor networks, thus many researchers have done research work in this area. Zhu, Qiao and Wang [9] take into account of the energy consumption due to not only data packets, but also control packets and retransmission by developing energy consumption models for both the End-to-end retransmission mode and hop-by-hop retransmission mode so as to avoid underestimating the total energy cost.

Dargie, Chao and Denko [10] motivate toxic gas detection during oil exploration and refinery, demonstrate how existing or proposed protocols can be employed to establish a fully functional network and provide a comprehensive energy model to evaluate the feasibility of employing wireless sensor network for the monitoring task.

Tseng, Yang, Chuang et al. [11] Present an analytical model for evaluating the energy consumption at nodes in a SMAC based wireless sensor network and make analysis of the energy consumption under different traffic conditions for distinct network topologies which may help designers to evaluate the energy consumption and determine the corresponding dominating factors.

Bruno, Conti and Gregori [12] deeply investigate the efficiency and the energy consumption of MAC protocols that can be described with a p-persistent CSMA model, study the theoretical performance bounds from the throughput and the energy consumption standpoint and discuss the trade-off between efficiency and energy consumption.

3 Network Model

To better estimate the energy consumption related with control packet of a MAC protocol in wireless sensor networks, it is vital to firstly set up network models. In the following subsections, we will describe several factors which may affect the establishing of the energy model in relation to control packet.

3.1 Sensing task

Normally, sensor nodes are densely placed either very close to or directly inside some particular phenomenon. Therefore, they usually operate unattended in uneasily reached areas of human beings while transmitting the interesting and important events periodically to researchers for analysis purpose and thus benefit for scientific communities and society as a whole. These sensor nodes may work in remote geographic environments, such as in islands or plains for habitat monitoring of some sensitive wild animals and endangered species [13]; in potentially dangerous
zones, such as around the active volcano places to collect seismic and infrasonic signals for monitoring volcanic eruptions[14]; in polluted and toxic sites such as inside the hydrogen or H2S gas systems to monitor these poisonous gas leakage and in the filling stations to identify the gas concentration and locating the leakage sources [15] [16]; in civil structure places such as in the Wisden application where wireless sensor nodes are used for locating the damages in structures like bridges and buildings [17] and the indirect detection of the structural state of the Golden Gate Bridge through the measurement and interpretation of ambient vibrations and strong motion [18].

Since wireless sensor networks are pretty suitable for applying to omnigenous fields of applications in various places, the energy consumption model with relation to control packet based on MAC protocols we will set up later will have a large use scope and high use value.

3.2 System deployment

System deployment [10] refers to the way how wireless sensor nodes are placed in an area where the sensing task should be effectively carried out. Such arrangement directly influences the sensing quality, the MAC protocol decision as well as the energy consumption evaluation. There are three basic strategies of the node deployment in theory. Spot monitoring, being as one of the deployment methods, needs to deploy only a few sensors in the target area with the assumption that the exact pinpoints of the sensing objectives have been known. Such type of the deployment requires the accuracy in accordance with specific environments, yet needed more time and effort to acquaint with the sensing field. Area and fence monitoring are the deployments which save much time of looking for the exact sensing locations. The former one requires the sensor nodes placing in all the regions where the sensing objectives may appear everywhere, which results in a large number of the necessary sensor nodes. The latter one, however, constructs a maximum range limit in order to guarantee that all the sensing targets are enclosed.

In our system, we prefer to use area deployment since it’s the most common and practical way of monitoring, thus based on which, the energy model will be more applied to reuse, reinvention and optimization.

3.3 Network topology

In wireless sensor networks, all nodes coordinate to support self-management for a common application and attempt to consume as little energy as possible. Mainly three types of network structures exist which are flat topology, hierarchical topology and location-based topology [10] [19]. In flat networks, all the sensor nodes typically play the same role concerning the routing task and collaborate to perform the data routing task based on only the local and neighborhood knowledge. The main problem of the flat topology is that per-node fairness can not be guaranteed. Since the nodes which are closer to the sink suffer more from the early power exhaustion, the energy consumption may not be evenly distributed through the entire wireless network. Not alike, hierarchical networks have the advantages of the scalability and efficient communication in data propagation. Clusters are formed by the self-organized sensor nodes with some of which act as the cluster heads. Such cluster heads run the role of data aggregation and data fusion so as to decrease the traffic forwarded to the sink. However, the creation and maintenance of the clusters consume much energy and in the meantime require the global knowledge of the whole network. And
location based routing makes use of the position information rather than the identity information of the sensor nodes to route data with the help of smaller routing tables.

In our network, for simplicity and efficiency reasons, we adopt the flat topology with the rectangle shape $A = a \times b$. Since such network structure proposes fewer assumptions of node relationships, that is, all the sensor nodes share the same network rank; it’s easier to calculate the average number of the neighbors for a particular node and the average number of the intersection neighbors for two one-hop nodes. And it’s easier to consider the choice of the routing protocols.

### 3.4 Coverage

Coverage [10], a significant system performance metric, is defined as how well a given area can be monitored by the network. Several papers [20] [21] propose models for computing the number of the right magnitude of the sensor nodes which are just needed to well cover the entire network in the meantime with high detection probability of the interested events. Coverage is dependent upon system deployment and network density.

For an area topology, the entire limited area should be effectively monitored with the assumption that all the nodes enclosed in such range are fully sponsored and the off-duty conflict problem is nonexistent.

### 3.5 Connectivity

Connectivity is a fundamental aspect which enables the wireless connection between any two immediate nodes. In order to make sure the sampled data can be successfully forwarded to the sink, at least one multi-hop path between the source and the sink node should be established, either directly or with the help of several intermediate nodes. The connection probability of a network mainly depends on the number of sensor nodes, the area density and the radio transmission range of the sensor nodes [22].

### 4 MAC Protocol

The data link layer [23] is in charge of the multi-plexing of data streams, data frame detection, medium access and error control. It ensures reliable point-to-point and point-to-multipoint connections in communication networks. The MAC protocol, being as the control strategy in one of the sub-layers of the data link layer, is responsible for creating the wireless network infrastructure which enables the sensor nodes to have the hop by hop communication and the self-organization ability, and regulating the access mechanism to guarantee the fairly and efficiently sharing of the communication resources among the contending sensor nodes.

Unlike cellular systems, Bluetooth and mobile ad hoc networks (MANET) where the MAC protocols mainly focus on providing high quality of service (QoS) and bandwidth efficiency, the MAC protocols employed in wireless sensor networks put the power efficiency in the prime
importance. In contrast to these systems, the wireless sensor network may have a much larger number of nodes however with the difficulties in replenishing the embedded batteries when they become exhausted. Besides, the transmission power and radio range of a sensor node is much less than those of Bluetooth and MANET, in addition, the topology changes more frequently. Therefore, the MAC protocols built in wireless sensor network should be in accordance with its unique characteristics, that is, prolong the network lifetime and minimize the total energy consumption.

There are several MAC protocols have been developed for wireless sensor networks. However, only a small part of them are suitable for the use as a basis on which a particular energy model can be built later due to some unique constraints and application requirements. In the following subsections, we will show the selection criteria of judging the suitability of the existing MAC protocols and explain the principles of them in detail in order to facilitate the comprehension and analysis of the energy model which is associated with control packet.

4.1 Protocol selection criteria

Thus far, several MAC protocols have been developed for wireless sensor networks, such as Time Division Multiple Access protocol (TDMA), Hybrid TDMA/FDMA protocol, Carrier Sense Multiple Access (CSMA) protocol, IEEE 802.11, SMAC, TMAC, BMAC and so forth. All these MAC protocols have their own merits and drawbacks, based on which, these MAC protocols should be used in their respective fields while meeting the application requirements and research objectives.

However, not all of these above MAC protocols fit for our view of research since in our analysis, we mainly focus on how the sampling rate affect the total energy consumption in the situations with and without control packet. On one hand, if control packet is used, extra energy will be consumed in the communication of control packet, which however does not contain the information that shows the condition of the sensing area and thus seems not benefit the real data transfer. On the other hand, without employing the control packet, problems such as hidden terminal, collision and overhearing will arise. Especially, collision and overhearing are another two primary energy sources in wireless sensor networks. Therefore, in light of the particular research objectives in our article, the MAC protocols we use must meet three criterions.

1. The control packet must be employed in the medium access mechanism. Obviously, it is not possible to analyze the contribution that control packet makes to the energy conservation without using it. In terms of this rule, only IEEE 802.11, SMAC, TMAC protocols meet the requirement. IEEE 802.11 protocol firstly adopts the CSMA/CA mechanism to establish a brief handshake between the sender and the receiver before the actual data transmission proceeds. SMAC and TMAC, being as its following optimized products, both operates with the medium access mechanisms employing control packet.

2. The MAC protocols must be energy-aware. Nowadays, network experts are more and more aware of the significance of the energy consumption in wireless sensor networks. As a result, we should try our best to avoid the energy sources as much as possible. From this perspective,
802.11, SMAC, TMAC and BMAC are preferable. In the 802.11 protocol, the random chosen backoff time is carried out immediately after the free-sensed DIFS period which highly decreases the probability of collision on two or more simultaneous transmitted packets. Thus the energy consumed in the retransmission will be minimized. In SMAC, the introduced periodic listen/sleep cycle solves the problem of idle listening. In addition, control packet mechanism successfully avoids overhearing of data fragments and reduces the retransmission cost due to its smaller size if collision occurs. TMAC, as a design optimized from SMAC, the duty cycle is further reduced by making nodes go to sleep based on the network traffic, however at the cost of the early sleeping problem. BMAC uses adaptive Low Power Listening (preamble sampling) to reduce duty cycle and provide flexible interface for reconfiguration and performance optimization.

3. The MAC protocols must be representative and future-popular. Research on the wireless sensor networks becomes one of the hot topics in wireless communications. Among the factors affecting the network performance, the media access control mechanism plays a critical role. We feel preferable to make analysis of the performance of those MAC protocols which are typical in use so as to maximize the research value, which will facilitate and benefit the further analysis of other MAC protocols optimized from them. From this standpoint, 802.11 and SMAC are the best ones.

From the above expatiation, we could draw a conclusion that our analytical energy model focusing on the influence that control packet brings to the energy consumption will be more suitable analyzed based on either IEEE 802.11 or SMAC protocol. In my small thesis, I set up the energy model with relation to control packet based on the 802.11 protocol. In order to further develop the affection that control packet brings to the total energy consumption when more factors and optimizations are required to be taken into account, we will build up a more complicated energy model based on the SMAC protocol. Therefore, in the following subsection, we will first of all give an explicit description of the working principle of SMAC, especially concentrate on explaining the optimizations achieved based on the IEEE 802.11 protocol.

4.2 SMAC protocol

In the previous section of the introduction, we have briefly described the there new optimizations in SMAC [3] comparing with the IEEE 802.11 protocol. In this part, we plan to further explain such three optimizations mainly by figures, which will facilitate the building of the collision probability model which runs the most important part in our analytical energy model.

Firstly, the SMAC protocol adopts periodic listen and sleep. In many sensor network applications, nodes are in idle status for a long period since the interesting events occur infrequently. Given the fact that sampling data rate during such period is very low, we prefer asking nodes to go to sleep mode periodically instead of letting them listen all the time. The description of the periodic listen and sleep is showed in the following figure 1.
Fig. 1. Periodic listen and sleep

Each node goes into listen and sleep mode periodically. During the listen phase, nodes turn on the radio to communicate with their intended neighbors for transmitting their own packets as well as receiving potential packets from the intermediate neighbors. During the sleep phase, however, nodes turn off the radio and set a timer to awake themselves when the next listen period starts. For any two nodes, their listen and sleep duration can be different, but for simplicity, these values are set the same for all the nodes. Please note during the sleep phase, the interesting events in the phenomenon are not stopped being sampled. As a result, many sampled data packets will be accumulated in the queue of each node and they will be transmitted during the next listen period. In fact, the actually reason we develop this periodical listen and sleep method is because the sampling rate is much lower comparing with the transmission rate. By using periodic listen and sleep, we highly reduce the largest source of the energy waste—idle listening and thus save huge amount of energy.

We decide to explain the relationship among the listen time, the sleep time, the frame and the duty cycle by using formulations in order to give a better understanding of the period listen and sleep mechanism. A complete cycle of a listen and sleep is a Frame and the duty cycle is the ratio of the listen duration to the entire frame phase.

\[ T_{frame} = T_{listen} + T_{sleep} \]  \hspace{1cm} (1)

\[ \text{Duty Cycle} = \frac{T_{listen}}{T_{frame}} \]  \hspace{1cm} (2)

All the sensor nodes are free to choose their own listen and sleep schedules, that is, when go to listen period and when go to sleep period. However, if these schedules are different for most of the nodes, the control overhead will be increased. For example, if the sender transmits the RTS frame to the receiver which is however still in its sleep period, so without receiving the CTS frame from the receiver, the sender has to re-transmit the RTS frame, described as the following figure 2. Or after the receiver receives the RTS frame, it sends back the CTS frame to the sender which however has already gone to sleep mode, displayed in the figure 3.
Therefore, to reduce the retransmission overhead of control packet, we need to synchronize the schedules of the neighboring nodes together by ensuring them listening at the same time and sleeping at the same time. Synchronization of schedules is accomplished by sending a SYNC packet periodically during the first transmission of the listen period. The SYNC packet is very short, which includes the sender address and the next sleep time of the node. A node will adjust its own schedule according to the received one and goes to sleep when the timer fires if it receives such schedule from a neighbor before choosing its own schedule. However, if a node does not hear a SYNC packet from one of its neighbors, it will randomly and independently select a schedule and broadcast it immediately in the SYNC fashion, based on which other nodes are able to synchronize. If a node receives a different schedule after it chooses and broadcasts its own schedule, it adopts both schedules and thus goes to sleep according to the time in both schedules. The reason that a node may fail in hearing a schedule within a predetermined amount of time is because that two or more SYNC packets sent from its neighbors are corrupted due to collision or a SYNC packet fails to be sent successfully due to errors during the transmission. Anyhow, nodes are expected to seldom have multiple schedules, since every node firstly tries to follow the existing schedules from its neighbors before choosing an independent one. Moreover, the probability of collision and error is very low during the schedule transmission. Therefore, for most of the sensor nodes in the wireless network, each of them has only one synchronized schedule.

In order for a node to receive both SYNC and data packets, we divide the whole listen period into two parts. As shown in the figure 4, the first part is for sending and receiving SYNC
packets and the second part is for carrying out the normal data communication. Before sending the SYNC and data packets, sensor nodes commence with the carrier sense by firstly listening for a DIFS period. If during this time, the channel is detected free, the randomly chosen backoff time begins to decrease. Only if the backoff time reaches zero, does the transmission starts.

Fig.4. Schedule synchronizations in the first part of the listen period

Secondly, SMAC brings in the message passing scheme. Data packets being transmitted may be short or long. For those long-size data packets, if errors occur during only few parts of the transmission, the whole packet must pay the price of the retransmission. However, if we fragment a long message into several independent and small fragments, a large amount of control overhead will be raised since for each fragment we need one RTS and one CTS to contend for the medium for its transmission, in the mean time, a longer delay will appears since receivers could only perform in-network data processing and aggregation after all the data units are obtained.

The message passing approach is to fragment the long data packet into many small fragments and transmit them in a burst. Only one RTS frame and one CTS frame are used. They perform the role of reserving the medium for transmitting all the fragments until the last ACK frame is received by the sender. Each time a data fragment is transmitted, the receiver will immediately send back an ACK frame to the sender to inform its successful reception. If the sender fails to receive the ACK frame, it will extend the reservation time for one more fragment and ACK frame and start the retransmission immediately. To interpret the message passing algorithm explicitly, please look at the figure 5.

Fig.5. Message Passing Algorithm

There are also duration time in both fragments and ACK frames. After the neighbors of the sender and receiver hear the RTS and CTS frames respectively, they will go to sleep mode for the entire message time. However, if the sender extends the transmission time due to fragment losses or errors, the sleeping neighbors will not notice such extension immediately. However, they are able to learn it from the extended fragments or ACK frames when they wake up.

Lastly, SMAC supports overhearing avoidance. In the 802.11 protocol, each node keeps listening to all the transmissions of its neighbors in order to perform effective virtual carrier sense. As a result, each node overhears a lot of packets which are not destined to it. This is a significant
source of the energy waste, especially when the network density is high and the traffic load is heavy.

In order to avoid overhearing, SMAC makes the nodes which receive the RTS frame or the CTS frame which are not directed to themselves to go to sleep mode. Since in each RTS frame and CTS frame, there is a duration field indicating how long the current transmission will last. Such duration value is recorded in a variable called Network Allocation Vector (NAV). As soon as the NAV timer fires, the NAV value decreases until it reaches zero. Thus by overhearing a control packet, a node will be aware that how long it should turn off the radio and keep silent. Since data packets are normally much larger than control packet, this approach can effectively prevent neighbor nodes from overhearing the long data packets and the corresponding ACK frames. The above figure 6 visualizes this technique of the overhearing avoidance. Suppose station A and B are neighbor nodes for each other. If A wants to send a data packet to one of its neighbors C, it firstly senses for a DIFS period, if during which, the channel is detected free, A will begin to decrease the randomly chosen backoff time. When such backoff time reaches zero, station A sends a RTS frame to C in order to reserve the medium for the transmission of the data packet. If at the same moment, station B also has a packet to send to one of its neighbors D, the same process is handled until it overhears the RTS frame transmitted by A. At this moment, B will be aware that the medium has been occupied by A already, so it should turn off its radio and go to sleep mode until the transmission between A and C completes.

In conclusion, SMAC optimizes itself in the above three aspects based on the 802.11 protocol. Here, we think it’s necessary and important to combine all these new features to display them in a single picture, as the figure 7 describes, from which all the characteristics of the SMAC protocol will be displayed unambiguously as an integral whole. Moreover, this figure will help us...
to deeply understand the collision probability model and the energy model with relation to control packet that we will set up in the following sections.

5 Energy Model of Arising Consequences with Control Packet

In this article, we focus on the analysis of the tradeoff between the extra energy consumption that control packet brings if we activate control packet and the extra energy consumption introduced if we deactivate control packet in the assumption of infrequent communications among sensor nodes. Our main purpose is to find how the sampling rate affects the energy consumption in the situation with and without control packet. Such trade-off analysis must be made based on some MAC layer protocols. In the previous sections, we have selected the most suitable protocol SMAC which enables and facilitates the setting up of a series of models. In this section, we only concentrate on one side of the trade-off, that is, the analysis of the extra energy consumption introduced if control packet is activated. Besides, we attempt to discover the relationship between the sampling rate and such extra energy consumption in both the saturated and the unsaturated period. Therefore, a series of models should be set up, such as the collision probability mode, the overlap neighbor mode and the extra energy model.

5.1 Features of energy model

Before setting up those concrete series of models, we prefer to firstly take into account of the factors that will affect the energy model. We identify the following quantifiable features.

1. We expect sensor networks to be composed of many small nodes to take advantage of physical proximity to the target to simplify signal processing and communication in a short-range multi-hop fashion.
2. We assume that finite nodes N are randomly distributed in a rectangular area A with the size $A = a \times b$. Without loss of generality, we assume $a \leq b$. The node distribution can be modeled as a two-dimensional Uniform distribution with average network density $\lambda$ where $\lambda = \frac{N}{a \times b}$.
3. Nodes in the wireless sensor network A are battery powered with identical initial energy where the battery for each node is exhaustible.
4. Nodes are deployed casually rather than carefully positioned. However, the locations are fixed once they are placed in the network. Thus we do not consider the extra energy consumption caused by mobility.
5. There is a single fixed sink located in the field A with the assumption of sufficient power supply.
6. All nodes N in the network communicate with the sink in a multi-hop fashion and act as both a data source and a relay.
7. Most communication is between nodes as peers, rather than to the sink.
8. Each node has the same radio transmission range R, so any two nodes are able to communicate via a wireless link only if their Euclidean distance is less then R.
9. For simplification, we assume the ideal channel conditions. Therefore, fading, path efficiency
and obstacles in the data propagation are not taken into account.

10. The energy consumption is analyzed in the assumption of infrequent communications among sensor nodes. As a result, applications will have long idle periods and thus can tolerate some latency.

11. Most sensor networks are dedicated to a single application or a few collaborative applications. Therefore, instead of node-level fairness, SMAC supports application-level fairness.

12. Sensor nodes have the capability of in-networking processing, data aggregation, collaborative signal processing and self-configuration.

5.2 Collision probability model

Collision probability has been extensively studied as it is a vital ingredient for many performance evaluations such as for throughput and delay in several MAC-layer protocols [2] [3] [6] [7]. Especially, collision probability is most analyzed in the 802.11 protocol by setting up analytical models. Such collision analysis is made however based on some limitations. For example, it is only focused on the “saturation situation” which assumes the sensor nodes always have packets to send in the wireless networks. In other words, it assumes that the sampling rate is always larger than or equal to the transmission rate ($R_s \geq R_t$). However, the saturated case only has a low probability of occurrence in the real world. In fact, what happens mostly is the unsaturated situation where the sampling rate is much less than the transmission rate.

As a result, nodes are in idle for a long time period if the new sampled data hasn’t come. Therefore, it is not necessary to keep nodes listening all the time. So SMAC is optimized in reducing the listen duration by letting nodes go to periodic sleep mode. Since during the sleep period, the sampling work does not stop, there will be many packets accumulated in each node’s queue when the next listen period arrives. Hence, in the first part of the saturated period ($t_0$ in the figure 8), nodes will transmit all these in-queue packets which are sampled during the previous sleep phase by using the transmission rate. In the same way, since the sampling work still continues during the saturated period $T_{saturated}$, several newly sampled packets will be accumulated in each node’s queue. And nodes should also keep transmitting such packets by using the transmission rate until a time point is reached (at the end of $t_1$). On this time point, all the data packets accumulated in the queue which are sampled during the whole saturated period $T_{saturated}$...
will be finished transmitting. And no packets will be stocked in the queue anymore from then on. So whenever comes a new sample, the node will send it out immediately. In other words, the unsaturated period starts, where a nonzero time interval will stand between every two neighbor transmissions. This interesting characteristic of the SMAC protocol is described in the figure 8.

Please note that the unsaturated time period must be existed in the listen phase although we want to reduce the idle listening as much as possible. The reason is because some extra time must be spared for carrying out the retransmission caused by collision and errors during the data packet transmission. Since the number of the packets which needed to be retransmitted is random and the number of the retransmission for every data packet is random, the unsaturated period $T_{unsaturated}$ should be set long enough to handle such retransmission problems. For example, the unsaturated period could equal to the time $t$ during which all the packets transmitted during $T_{saturated}$ have to be retransmitted until their maximum transmission attempts are reached plus the time $t'$ during which all the newly sampled packets during $t$ and $t'$ should be finished transmitting and retransmitting. Interestingly, if this maximum retransmission situation does occur, the time duration of the unsaturated period will be zero. However, the total required retransmission time must be much less than this threshold value in reality, so the packets will be definitely transmitted unsaturatedly from some moment on.

$$T = 0$$

![Diagram](image.png)

Fig.9. The saturated case and the unsaturated case with regard to $T$

If we recall the working principle of the SMAC protocol discussed in the section 4.2, it says after the successful communication of the final ACK frame between the sender A and the receiver B, all the neighbor nodes of both A and B will finish their sleep and wake up to contend for the medium, since there is a duration timer in both the RTS frame sent by A and the CTS frame sent by B indicating how long the remaining transmission is. Thus, when the timer fires and reaches zero, the channel is free and a new round of the competition starts. If these neighbor nodes have some available packets at hand, that means several sampled packets have already been stocked in their queues, they will contend for the medium immediately after the previous transmission by firstly monitoring the channel for a period of DIFS. Therefore, in the saturated period, the DIFS sensing interval is immediately follows the previous final ACK frame. In the unsaturated period, however, since the sampling rate is far smaller than the transmission rate, the newly sampled data
packet has not come yet when the previous transmission finishes. As a consequence, there will be a nonzero time interval \( T \) between the end of the previous final ACK packet and the beginning of the new DIFS phase. The difference between them is described in the figure 9 where (a) represents the time interval in the saturated period and (b) represents the time interval in the unsaturated period.

Depending on the size of the time interval between the end of the previous final ACK frame and the beginning of the new DIFS phase, we will classify the collision probability research into two parts, in the first part we will set up the collision probability model in the saturated period where \( T=0 \), and in the second part the collision probability will be analyzed in the unsaturated period where \( T>0 \). Since in the SMAC protocol, the sampling rate is always smaller than the transmission rate (that’s the mainly reason we develop SMAC) and the duty cycle is always smaller than one, both the saturated and the unsaturated period will exist in each listen phase. After the two corresponding collision probabilities are separately evaluated, we will present the overall collision probability by averaging these two collision probabilities using weight which is analyzed based on the total number of the data packets transmitted during the saturated and the unsaturated period. Only if this overall collision probability model is set up, can we begin to build the energy consumption mode in the situation with control packet.

Recall that in the 802.11 protocol, we only set up the energy consumption model based on the collision probability either considered in the saturated situation or unsaturated situation, rather than an average one. This is simply because the data packets are transmitted either saturatedly or unsaturatedly in the 802.11 protocol for a predetermined sampling rate. Differently, in SMAC, since the duty cycle is always smaller than one, even if the sampling rate is much smaller than the transmission rate, the saturated period will still appear during each listen period.

In the following subsections, we will deeply analyze the characteristics of the collision in the SMAC protocol and set up the corresponding collision probability model in both the saturated and unsaturated period by employing equation sets.

### 5.21 Collision probability in saturated period

In [2], the author gave the definition of the collision probability as the probability of a collision seen by a packet being transmitted on the channel. A discrete-time Markov chain model is constructed to deduce the exact formula of the collision probability in the saturated situation of the 802.11 protocol, which was then simplified and further developed in [7].

In my small thesis, I summarized the formation of the collision probability model built in the saturated situation based on the 802.11 protocol. Since the average neighbor node is one of the parameters affecting the evaluation of such collision probability, I set up the average neighbor model firstly in the assumption that the sensor nodes are uniformly and independently distributed in the wireless sensor network, which effectively estimated the average number of the neighbor nodes from the perspective of a transmitter or a receiver. The average neighbor model is recalled below, where \( N \) represents the total number of the sensor nodes in the network, \( a \times b \) is the area...
of the wireless sensor network, \( R \) stands for the average radio transmission range and the symbol \( \lfloor \rfloor \) indicates the floor integer of the value of the fraction located inside.

\[
N_{\text{neighbor}} = \left\lfloor \frac{N}{a \times b \times \pi R^2} \right\rfloor - 1 \quad (3)
\]

Afterwards, I successfully evaluated the overall average backoff time by making use of the relationship among the collision probability in the saturated situation, the size of the contention window and the number of the retransmission attempts. After the simplification and the transformation, the overall average backoff model is summarized as below, where \( T_{\text{avg}} \) is the overall average backoff time, \( p_s \) is the collision probability in the saturated situation, \( m \) is the maximum backoff stage and \( K \) is the maximum transmission attempt.

\[
T_{\text{avg}} = \left[ \frac{(1 - p_s) \left( \frac{CW_{\text{min}} (1 - (2p_s)^m)}{2(1 - 2p_s)} \right) + \left( \frac{p_s^{m+1} 2^m CW_{\text{min}} (1 - p_s^{K-m-1})}{2(1 - p_s)} \right)}{1 - p_s^K} \right] \quad (4)
\]

Since both \( m \) and \( K \) are two single parameters, they can be directly assigned a value. Not alike, \( T_{\text{avg}} \) and \( p_s \) are two complicated parameters, the value of which are determined by some other single parameters. Therefore, by only applying the formula (4), the value of the collision probability in the saturated situation is not enough to be calculated. However, if we are able to find another formula which also comprises these two parameters \( T_{\text{avg}} \) and \( p_s \), then by solving the equation set, the value of the collision probability in the saturated situation will be successfully estimated. And this formula is set up by employing the so called ball-box model, where the ball stands for the data packet transmitted by each neighbor of the sender and the box represents the every beginning of the slot time. In conclusion, the relationship between the overall average backoff time \( T_{\text{avg}} \) and the collision probability in the saturated situation \( p_s \) deduced from the ball-box model is summarized below.

\[
p_s = 1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}} - 1} \quad (5)
\]

As stated before, no matter how low the sampling rate is in SMAC, since the duty cycle is always smaller than one, the saturated period must be definitely appeared during the listen phase. Even though the SMAC protocol is optimized based on the 802.11 protocol in several aspects, such as the periodic listen and sleep, the message passing scheme and the overhearing avoidance...
skill, however, none of these newly developed technologies has an influence on the method which evaluates the collision probability in the saturated case, since all of the random backoff mechanism, the network density and the size of the time interval share the same value in both the 802.11 and SMAC protocols. As a result, the equation set which used to estimate the collision probability in the saturated case in the 802.11 protocol also fits for the evaluation of the collision probability in the saturated period of the listen phase in SMAC. In order to give a clearer observation of the relationship among all the related parameters, such equation set is fully displayed as the formula (6) here.

$$
\begin{align*}
N_{\text{neighbor}} &= \left\lfloor \frac{N}{a \times b} \times \pi R^2 \right\rfloor - 1 \\
T_{\text{avg}} &= \frac{(1-p_s) \left(\frac{CW_{\text{min}} (1-(2p_s)^m)}{2(1-2p_s)} \right) + \left(\frac{p_s^m 2^m CW_{\text{min}} (1-p_s^{K-m})}{2(1-p_s)} \right) - \left(\frac{1-p_s^K}{2(1-p_s)} \right)}{1-p_s^K} \\
p_s &= 1 - \left(\frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}}}
\end{align*}
$$

(6)

5.22 Collision probability in unsaturated period

As stated before, due to the potential demand for the retransmission if collision or errors appear during the data packet transmission, the unsaturated period will occur from some moment on in each listen phase of the SMAC protocol. During the unsaturated period, the sampled data packets will be transmitted unsaturatedly since the sampling rate is far smaller than the transmission rate. As a consequence, after the current data packet is finished transmitting, the newly sampled data packet has not come to the queue yet. Hence, for an arbitrary node, there always exists a time interval $T$ ($T>0$) between the end of its previous final ACK frame and the beginning of its new DIFS period. Since this time interval is no longer zero, the transmitters do not always have packets to send at hand. Thus, not all the neighbors of a particular sensor node are active transmitting neighbors, so the method which evaluates the collision probability in the saturated period of the listen phase will not suit for the analysis of the collision probability in the unsaturated period any more.

Since the number of the active transmitting neighbor nodes is the key parameter affecting the performance of the collision probability in the unsaturated period, we should first of all set up the active neighbor node model.

5.221 Active neighbor model

In my small thesis, I developed an analytical model to accurately and effectively evaluate the number of the active transmitting neighbor nodes in the unsaturated situation of the 802.11
protocol based on its carrier sense mechanism and transmission format. Since the carrier sense scheme and the transmission manner in the SMAC protocol is similar with that in the 802.11 protocol, the analogous analysis method can be used to estimate the number of the active transmitting neighbor nodes in the unsaturated period of the SMAC protocol.

In the same way, we assume the node B, C, D and E are all the neighbor nodes of A. And they all constantly transmit sensed data to A. If B’s DIFS sensing time has been predetermined, then according to which, there are three and only three possibly different transmission patterns in the unsaturated period of the SMAC protocol. The difference among these three transmission patterns is their relative time of the DIFS period with respect to B’s DIFS time. Nevertheless, they also share some identical characteristic that is all their final ACK frames must occur before B’s DIFS sensing period. For all neighbors of A, after they sense idle for a period of DIFS, they will start their random chosen backoff time respectively. Therefore, the backoff time for each of them is mostly different. For simplification and efficiency reasons, we regard all these different backoff time as the same value which equals to the overall average backoff time $T_{avg}$ we have calculated.

For any remaining neighbor node of A, besides considering the possible transmission time of its final ACK frame according to B’s transmission pattern, we should also take the possible sense time of its DIFS period into account. Actually, its DIFS period is allowed to occur either before B’s DIFS sensing period, during B’s backoff time or after B’s transmission process. So just according to these three possible time positions of the DIFS period, we classify the remaining neighbor nodes of A into three categories, each of which matches one unique transmission pattern.

In the unsaturated period of the SMAC protocol, each of the remaining neighbor nodes of A has one of those three transmission patterns according to the predetermined transmission pattern of the node B. However, not all of these remaining neighbor nodes are active transmitting nodes. Some of them are active while some of them are non-active. Since collision may only occur in C’s transmission pattern, besides the node B, only those C-like remaining neighbor nodes are active transmitting nodes. As a result, those neighbor nodes which have the same transmission pattern as that D and E has are the non-active transmitting nodes since their transmission will never suffer from collision with B’s transmission. Since all the remaining neighbor nodes of A could only have C’s transmission pattern when the time interval is required to be zero, all the neighbor nodes are active in the saturated period of the SMAC protocol.

In each of the three transmission patterns, there are several possible cases. That because those three different transmission patterns are classified by the time range of the DIFS periods rather than the exact time point. Therefore, in order to acquire the accurate number of the active transmitting neighbor nodes, we need to know the ratio of the possible cases in the transmission pattern of the active neighbor nodes to the total possible cases in the transmission pattern of both the active and non-active neighbor nodes.

Deserve to be mentioned, we observe that the number of the possible cases in D and E’s transmission pattern is mostly $T$ related. And such number varies in the different conditions.
Therefore, we will evaluate the number of the active transmitting neighbor nodes in each condition respectively. As showed in the figure 10 below, the time coordinate $t$ is divided into four zones and only the corresponding specific transmission patterns make sense depending on which time zone $T$ belongs to. The notation $\sigma$ stands for a slot time, $T_{\text{avg}}$ is overall average backoff duration, $T_{\text{DIFS}}$ represents the time needed for sensing a DIFS period and $T_{\text{trans}}$ means the time used for the whole packet transmission by applying the message passing. As we can see, the transmission pattern of B certainly exists, and so does the transmission pattern of C no matter what value $T$ is taken. Please note that $T$ must be larger than 0. Otherwise, it will become the saturated situation. Furthermore, for C’s transmission pattern, its number of the transmission cases is constant permanently. Not alike, D’s transmission pattern only makes sense when $T$ is equal to or larger than $\sigma$, and the number of its transmission cases dose not remain the same when $T$ changes from the time zone $[0, \sigma)$ to $[T_{\text{avg}} \sigma + T_{\text{DIFS}} + T_{\text{trans}})$. That’s why we use the letter D and D’ in these two different time zones respectively. As to E’s transmission pattern, it only possible to appear as long as $T$ is equal to or larger than $T_{\text{avg}} \sigma + T_{\text{DIFS}} + T_{\text{trans}}$. In the following part of this subsection, we plan to concentrate on evaluating the number of the active transmitting nodes in the unsaturated period of the SMAC protocol based on the classification of $T$.

Fig.10. The classification criterion according to $T$

1. $T \in (0, \sigma)$

   Obviously, only the transmission pattern of B and C exists in this situation. As we analyzed before, besides the active neighbor node B, only those remaining neighbor nodes which have the same transmission pattern as C has are active. There is only one possible transmission case in B’s transmission pattern since we have assumed B’s transmission pattern as a point of reference in order to analyze all the other possible transmission patterns. And there are possible $(T_{\text{avg}} + 1)$ transmission cases together in C’s transmission pattern which share the identical occurrence probability.

   $$N_B = 1$$  \hspace{1cm} (7)

   $$N_C = T_{\text{avg}} + 1$$  \hspace{1cm} (8)

   Under this first condition, since only the transmission pattern of the active neighbor nodes appears, the numerator and the denominator of the ratio which calculates the average number of
the active transmitting neighbor node $M$ should share the same value, which is summarized as
below, where $N_{\text{neighbor}}$ is the average number of the neighbor nodes of a node, say, node $A$.

\[
M = \frac{1+(T_{\text{avg}} + 1)}{1+(T_{\text{avg}} + 1)} N_{\text{neighbor}} 
\]

(9)

This formula can be further simplified as:

\[
M = N_{\text{neighbor}} 
\]

(10)

As we will analyze later, this result is the same as the average number of the active neighbor node evaluated in the saturated period.

2. $T \in [\sigma, T_{\text{avg}} \sigma)$

In this case, besides the transmission of $B$, both the transmission patterns of $C$ and $D$ make sense. The possible cases in $C$’s transmission pattern are not changed and thus the number of its transmission cases remains the same, that is, $(T_{\text{avg}} + 1)$ as well.

However, the transmission cases in $D$’s transmission pattern are $T$ dependent. In the $D_{\frac{T}{\sigma}}$ th case, the finish time point of the DIFS sense period is not able to reach the start time of $B$’s transmission process since $T$ is smaller than $T_{\text{avg}} \sigma$. Therefore, with the limitation of $T$, there are merely $\left\lfloor \frac{T}{\sigma} \right\rfloor$ cases in all, where only the integer part of the quotient of this fraction is accepted since this number has a practical meaning. The number of the transmission case in the transmission pattern $D$ is summarized below.

\[
N_D = \left\lfloor \frac{T}{\sigma} \right\rfloor 
\]

(11)

In this situation, both $C$ and $D$’s transmission patterns exist besides $B$’s transmission, among which, only $B$ and $C$ are the active transmitting neighbor nodes. Therefore, the proportion of the transmission cases of the active nodes to the transmission cases of all the existent nodes no matter they are active or non-active should be calculated as:

\[
M = \left\lfloor \frac{1+(T_{\text{avg}} + 1)}{1+(T_{\text{avg}} + 1)} + \frac{T}{\sigma} \right\rfloor N_{\text{neighbor}} 
\]

(12)
The above formula can be simplified as:

\[
M = \frac{T_{\text{avg}} + 2}{T_{\text{avg}} + 2 + \frac{T}{\sigma}} N_{\text{neighbor}}
\] (13)

3. \( T \in \left[ T_{\text{avg}} \sigma, T_{\text{avg}} \sigma + T_{\text{DIFS}} + T_{\text{trans}} \right) \)

When \( T \) belongs to this time zone, all the transmission patterns of the node B, C and D could happen. The number of the transmission cases of C’s transmission pattern still remains the same. However, the number of the transmission cases of D’s transmission pattern changes to \( T_{\text{avg}} \), since \( T \) is big enough now that it will not be a limitation anymore for the total number of the possible transmission cases. Specifically speaking, from the case in which its DIFS is closest to B’s DIFS to the case in which the finish time point of its DIFS reaches B’s zero backoff time point are all included.

\[
N_D = T_{\text{avg}}
\] (14)

As a result, no matter how large \( T \) is, there are always \( T_{\text{avg}} \) cases in the transmission pattern of D. Thus, the number of the active transmitting neighbor nodes in this case can be computed as:

\[
M = \frac{1 + (T_{\text{avg}} + 1)}{1 + (T_{\text{avg}} + 1) + T_{\text{avg}}} N_{\text{neighbor}}
\] (15)

The above formula can be further simplified as:

\[
M = \left[ \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 2} N_{\text{neighbor}} \right]
\] (16)

4. \( T \in \left[ T_{\text{DIFS}} + \sigma T_{\text{avg}}, +\infty \right) \)

If \( T \) belongs to this time area, all of the transmission patterns of B, C, D and E make sense. As it is known already, only one case under B’s transmission pattern, \( (T_{\text{avg}} + 1) \) cases under C’s transmission pattern, \( T_{\text{avg}} \) cases under D’s transmission pattern since \( T_{\text{avg}} \) is the maximum value of the transmission cases that take place when \( T \) is equal to or larger than \( \sigma T_{\text{avg}} \). As to the transmission cases happened under E’s transmission pattern, please have a look at the figure 11.
As we can see, the number of the transmission cases of E’s transmission pattern is 
\[ \left( \frac{T - T_{\text{DIFS}} - \sigma T_{\text{avg}} - T_{\text{trans}}}{\sigma} + 1 \right) \] in this situation where \( T_{\text{trans}} \) is the transmission time by using the message passing mechanism where a long message is divided into \( n \) small fragments to reduce the retransmission overhead if errors only occur in the very first transmission part. So every time a data fragment is transmitted, the sender waits for an ACK from the receiver. If it fails to receive the ACK, it will extend the reserved transmission time for one more fragment and retransmit the current fragment immediately rather than re-contending for the medium to handle the retransmission. There are two cases here. If the data fragment is successfully received by the receiver which however fails in sending back an ACK frame, the retransmission of both the fragment and the ACK frame are required. And if the data fragment is transmitted unsuccessfully by the sender which is informed by the lack of the ACK frame, only the data fragment is needed to be retransmitted. Therefore, the average time duration for a data fragment which includes both the transmission time and the retransmission time is 
\[ \frac{N_f}{R_i} \] while the average time duration needed for both the transmission and the retransmission of an ACK frame is 
\[ \frac{N_{\text{ack}}}{R_i} \]. Here, 
\( N_f \) and \( N_{\text{ack}} \) is the size of a data fragment and an ACK frame respectively. \( R_i \) indicates the...
transmission rate which is a constant value in the assumption of ideal channel conditions. $p_{f_{ij}}, p_{a_{ij}}$ represents the error probability during the fragment transmission and the ACK frame transmission between the node i and its neighbor node j respectively. If the message is divided into n fragments, the total SIFS time where the SIFS phase only occurs between every neighbor packets is $[(2n + 1) \times T_{SIFS}]$, which does not include those extra SIFS time caused by the retransmission of the data fragments and ACK frames, since the retransmission seldom occurs due to the very low error probability. In conclusion, the average time duration of the whole transmission in the situation where control packet is activated is:

$$T_{trans} = \frac{N_{cts}}{R_t} + \frac{N_{ct}}{R_t} + n \frac{R_t}{p_{f_{ij}} p_{a_{ij}}} + n \frac{R_t}{p_{a_{ij}}} + (2n + 1)T_{SIFS} \quad (17)$$

The formula can be simplified as:

$$T_{trans} = \frac{N_{rts}}{R_t} + \frac{N_{ct}}{R_t} + \frac{nN_{f}}{R_t p_{f_{ij}} p_{a_{ij}}} + \frac{nN_{ack}}{R_t p_{a_{ij}}} + (2n + 1)T_{SIFS} \quad (18)$$

Here, we do not take the retransmission of the RTS and CTS frame into account. Since only if they are successfully transmitted, does the transmission of the fragments and ACK frames can appear. In other words, if either the RTS or the CTS fails in the transmission, a new round of the competition for the medium will start, which begins with a new DIFS sensing period.

Deserve to be noticed, in the 802.11 protocol, the retransmission of the data packet and the ACK frame is not considered in the whole transmission time, since every failed data packet or ACK frame will lead to the retransmission from the beginning, that is, start with a new round of DIFS sensing period. Thus, both the RTS and CTS frame will pay the price of the unsuccessful transmission of every following data packet or ACK frame. Differently, in SMAC, a failed fragment or ACK frame will be retransmitted immediately in the extended reserved transmission time instead of causing the retransmission of the RTS and CTS. And this is the exact reason that we should only consider the retransmission time of the fragment and the ACK frame caused by errors when we are evaluating the whole average transmission duration in SMAC.

In conclusion, the number of the transmission cases of E’s transmission pattern is:

$$N_E = \left[ \frac{T - T_{DIFS} - \sigma T_{avg} - T_{trans}}{\sigma} \right] + 1 \quad (19)$$

And the number of the average active transmitting neighbor nodes is:
\[ M = \frac{1+(T_{\text{avg}}+1)}{1+(T_{\text{avg}}+1)+T_{\text{avg}}+\left[\frac{T-T_{\text{DIFS}}-\sigma T_{\text{avg}}-T_{\text{trans}}}{\sigma}\right]+1} N_{\text{neighbor}} \quad (20) \]

This formula is further transferred to:

\[ M = \frac{T_{\text{avg}}+2}{2T_{\text{avg}}+3+\left[\frac{T-T_{\text{DIFS}}-\sigma T_{\text{avg}}-T_{\text{trans}}}{\sigma}\right]} N_{\text{neighbor}} \quad (21) \]

Where

\[ T_{\text{trans}} = \frac{N_{\text{ets}}}{R_t} + \frac{N_{\text{cts}}}{R_i} + \frac{n N_f}{R_t p_{fij} p_{aji}} + \frac{n N_{\text{ack}}}{R_i p_{aji}} + (2n+1)T_{\text{SIFS}} \quad (22) \]

As we can obviously observe, the average number of the active transmitting neighbor node M in the unsaturated period of the SMAC protocol is a function of the parameters \( T_{\text{avg}} \), \( T \), \( \sigma \), \( T_{\text{DIFS}} \), \( N_{\text{ets}} \), \( N_{\text{cts}} \), \( N_f \), \( N_{\text{ack}} \), \( R_t \), \( T_{\text{SIFS}} \), \( p_{fij} \), \( p_{aji} \) and \( n \). All these parameters are only simple variables except for \( T \) and \( T_{\text{avg}} \) which are two unknown complicated parameters. In order to fully resolve the value of M in all these four different classifications, we must look for the exact value of \( T \) by establishing the specific \( T \) model.

5.222 T model

It’s known already that the reason why \( T>0 \) in the unsaturated period of SMAC is because the sampling rate is much smaller than the transmission rate. So after a round of transmission, a blank time interval T appears. This phenomenon is described in the following figure 12. Due to the very low error probability occurred during the transmission of every data fragment and ACK frame, the retransmission probability is very small. Thus we decide not to show such retransmission phenomenon in the figure 12.

The period T1 from the beginning of the first DIFS to the end of the last ACK frame is the time needed for transmitting a group of packets which include the RTS frame, CTS frame, several data fragments and the corresponding ACK frames. And the period T2 from the beginning of the first DIFS to the beginning of the second DIFS is the time used for sampling an interesting event in the environment and forming the sampled bytes into a corresponding packet with the
assumption that the sensors within each sensor node sample unsimultaneously. Therefore, by
multiplying together the sampling rate reciprocal and the packet size, the total time required for
sampling a data packet is able to be evaluated.

Fig. 12. T is a function of T1 and T2

Therefore, we could find out the time duration T1 and T2 with the help of the related
transmission rate $R_t$ and sampling rate $R_s$ respectively.

$$T_1 = \frac{N_{\text{req}}}{R_t} + \frac{N_f}{R_s} + n \frac{R_s}{p_{fj}p_{aji}} + n \frac{R_s}{p_{aji}} + (2n + 1)T_{\text{SIFS}} + T_{\text{DIFS}} + T_{\text{avg}}$$

And

$$T_2 = \frac{N_{\text{data}}}{R_t}$$

Where $N_{\text{data}} = nN_f$, since the whole data packet size equals to the summation of the size
of all the data fragments in the assumption that all the data fragments share the same size.

So obviously, the time interval T between the end of the previous final ACK frame and the
beginning of the new DIFS sensing period we have mentioned many times in the subsection when
considering the average number of the active transmitting node in the unsaturated period equals to
T2 minus T1.

$$T = T_2 - T_1$$

After inserting the formula (23) and (24) into the above formula (25), the exact value of the
time interval T in the unsaturated period of the SMAC protocol can be further written as:

$$T = \frac{nN_f}{R_t} - \frac{N_{\text{req}}}{R_t} - \frac{N_{\text{acks}}}{R_t} - \frac{nN_f}{R_t} \frac{p_{aji}}{R_t} - \frac{nN_{\text{ack}}}{R_t} - (2n + 1)T_{\text{SIFS}} - T_{\text{DIFS}} - T_{\text{avg}}$$

Since all these parameters deciding the value of the time interval T in the unsaturated period
are easy and convenient to be evaluated except for the overall average backoff time $T_{\text{avg}}$ and the
sampling rate $R_s$, the time duration $T$ is actually a function of the parameter $R_s$ and $T_{avg}$. As it was analyzed in the previous subsection, the average number of the active transmitting neighbor nodes $M$ was dependent of the time interval $T$ only under the second and fourth condition. So if we insert the formula (26) which gave the exact value of $T$ into the previous formulas (13) and (21) which successfully evaluated the number of the active transmitting neighbor node in the second and fourth classification respectively, the specific relationship among the average number of the active transmitting neighbor nodes $M$, the sampling rate $R_s$ and the overall average backoff time $T_{avg}$ is able to be discovered.

5.223 Collision probability model

In the previous section when we discuss the collision probability in the saturated period of the SMAC protocol, the formula $p_s = (1 - (1 - \frac{1}{T_{avg}})^{M-1})$ is set up by using the ball-box model. Obviously, such collision probability in the saturated period is a function of the parameter $T_{avg}$ and $M$ where $T_{avg}$ stands for the overall average backoff time and $M$ represents the average number of the active transmitting neighbor node. In the saturated period of the listen phase, since all the neighbor node of an arbitrary node are active transmitting neighbor nodes, $M$ equals to $N_{neighbor}$. Not alike, in the unsaturated period, $M$ is not a single function anymore. Instead, $M$ has different representing formulas in different conditions which are classified based on the value of the time interval $T$. More concretely, the collision probability in the unsaturated period depends mostly on the relationship among the transmission rate $R_t$, the sampling rate $R_s$ and the overall average backoff time $T_{avg}$.

Besides the change of the value $M$, the coefficient of the formula changes as well since not all the neighbor nodes of a particular node are active any more in the unsaturated period. As a consequence, we should firstly select those active transmitting nodes from all the neighbor nodes, and then consider and analyze the collision probability based on these active transmitting neighbor nodes. Therefore, after some modification, the collision probability occurred on the RTS frame in the unsaturated situation of the SMAC protocol is given as:

$$P_u = \frac{M}{N_{neighbor}} \left[1 - \left(1 - \frac{1}{T_{avg}}\right)^{M-1}\right]$$

(27)

In the above formula (27), the value of the parameter $N_{neighbor}$ can be easily calculated by
using the formula (3). Then the collision probability in the unsaturated period $p_u$ becomes a function of the parameters $T_{avg}$ and M. Interestingly, the average number of the active transmitting neighbor node M in the unsaturated period is in fact a function of the overall average backoff time $T_{avg}$. So if we combine each formula (10), (13), (16), (21) which estimates the value of the active neighbor nodes in each condition with the formula (27) which evaluates the collision probability in the unsaturated period respectively, we are able to find out the relationship between the collision probability in the unsaturated period $p_u$ and the overall average backoff time $T_{avg}$. So we may be aware that if another formula which also gives the relationship between $p_u$ and $T_{avg}$ can be discovered, by solving this equation set, we are able to eventually obtain the value of the collision probability in the unsaturated period $p_u$ in theory. Fortunately, such formula has been emerged already. Recall that when we analyzed the collision probability in the saturated period $p_s$, a formula that comprised both the variables $p_s$ and $T_{avg}$ had been established, which is re-displayed below.

$$T_{avg} = \left(1 - p_s\right) \left(\frac{CW_{min}\left(1 - (2p_s)^{-1}\right)}{2\left(1 - 2p_s\right)}\right) + \left(\frac{p_s^{-1} 2^m CW_{min}\left(1 - p_s^{K^{-1}}\right)}{2\left(1 - p_s\right)} - \left(\frac{1 - p_s^{K}}{2\left(1 - p_s\right)}\right)\right) \quad (28)$$

Although such formula is set up in the saturated period of the SMAC protocol, it is applied to the unsaturated period as well since the value of $T_{avg}$ is independent of the time interval T which is actually the fundamental difference between the saturated and unsaturated period. Thus, just by modifying the form of the expression, the relationship between the overall average backoff time $T_{avg}$ and the collision probability $p_u$ in the unsaturated period where control packet is activated is able to be set up.

$$T_{avg} = \left(1 - p_u\right) \left(\frac{CW_{min}\left(1 - (2p_u)^{-1}\right)}{2\left(1 - 2p_u\right)}\right) + \left(\frac{p_u^{-1} 2^m CW_{min}\left(1 - p_u^{K^{-1}}\right)}{2\left(1 - p_u\right)} - \left(\frac{1 - p_u^{K}}{2\left(1 - p_u\right)}\right)\right) \quad (29)$$

From all of the above, the value of the collision probability in the unsaturated period $p_u$ is
able to be finally resolved under different conditions of SMAC in theory. Here, we’d like to summarize the entire formulas which contribute to the evaluation of the collision probability in the unsaturated period in the form of equation set under different conditions.

1. $T \in (0, \sigma)\n$\n\[
\begin{align*}
N_{\text{neighbor}} &= \frac{N}{ab} \pi R^2 - 1 \\
M &= N_{\text{neighbor}} \\
p_u &= \frac{M}{N_{\text{neighbor}}} \left(1 - \left(1 - \frac{1}{T_{\text{avg}}} \right)^M \right) \\
T_{\text{avg}} &= \frac{(1-p_u) \left(\frac{CW_{\text{min}} (1-(2p_u)^{m+1})}{2(1-2p_u)} + \frac{p_u^{m+1} 2^m CW_{\text{min}} (1-p_u^{K-m-1})}{2(1-p_u)} - \frac{1-p_u^K}{2(1-p_u)}\right)}{1-p_u^K}
\end{align*}
\]

(30)

2. $T \in [\sigma, T^u_{\text{avg}}] \sigma$\n\[
\begin{align*}
N_{\text{neighbor}} &= \frac{N}{ab} \pi R^2 - 1 \\
M &= \frac{T_{\text{avg}} + 2}{T_{\text{avg}} + 2 + \frac{2}{\sigma}} N_{\text{neighbor}} \\
T &= \frac{nN_f}{R_t} - \frac{N_{\text{ra}}}{R_t} - \frac{N_{\text{ca}}}{R_t} - n \frac{R_r}{p_{\text{fj}}} - \frac{R_r}{p_{\text{aj}}} - n \frac{R_i}{p_{\text{fj}}} - \frac{R_i}{p_{\text{aj}}} - (2n+1)T_{\text{SIFS}} - T_{\text{DIFS}} - T_{\text{avg}} \sigma \\
p_u &= \frac{M}{N_{\text{neighbor}}} \left(1 - \left(1 - \frac{1}{T_{\text{avg}}} \right)^M \right) \\
T^u_{\text{avg}} &= \frac{(1-p_u) \left(\frac{CW_{\text{min}} (1-(2p_u)^{m+1})}{2(1-2p_u)} + \frac{p_u^{m+1} 2^m CW_{\text{min}} (1-p_u^{K-m-1})}{2(1-p_u)} - \frac{1-p_u^K}{2(1-p_u)}\right)}{1-p_u^K}
\end{align*}
\]

(31)

3. $T \in [T_{\text{avg}} \sigma, T_{\text{avg}} \sigma + T_{\text{DIFS}} + T_{\text{trans}}]$
4. $T \in [T_{DIFS} + \sigma T_{avg} + T_{trans}, +\infty)$

\[
N_{neighbor} = \left\lfloor \frac{N}{ab} \pi R^2 \right\rfloor - 1
\]

\[
M = \frac{T_{avg} + 2}{2T_{avg} + 2} N_{neighbor}
\]

\[
p_u = \frac{M}{N_{neighbor}} \left(1 - \left(1 - \frac{1}{T_{avg}}\right)^{M-1}\right)
\]

\[
T_{avg} = \frac{(1-p_u) \left\{ \frac{CW_{min} \left(1-(2p_u)^{m+1}\right)}{2(1-2p_u)} + \left(\frac{p_u^{m+1} 2^m CW_{min} (1-p_u^{K-m-1})}{2(1-p_u)}\right) - \left(1-p_u^K\right) \right\}}{1-p_u^K}
\]

\[
\begin{align*}
N_{neighbor} & = \left\lfloor \frac{N}{ab} \pi R^2 \right\rfloor - 1 \\
M & = \frac{T_{avg} + 2}{2T_{avg} + 2} N_{neighbor} \\
p_u & = \frac{M}{N_{neighbor}} \left(1 - \left(1 - \frac{1}{T_{avg}}\right)^{M-1}\right) \\
T_{avg} & = \frac{(1-p_u) \left\{ \frac{CW_{min} \left(1-(2p_u)^{m+1}\right)}{2(1-2p_u)} + \left(\frac{p_u^{m+1} 2^m CW_{min} (1-p_u^{K-m-1})}{2(1-p_u)}\right) - \left(1-p_u^K\right) \right\}}{1-p_u^K}
\end{align*}
\]

Up to now, the value of the collision probability in both the saturated and unsaturated period of the SMAC protocol has been successfully resolved. According to the result of such collision probabilities, we can observe that they are quite different from the value of the collision.
probabilities evaluated in the 802.11 protocol. Such difference is caused by the optimizations the SMAC protocol made based on the 802.11 protocol.

Due to the introduction of the message passing mechanism, the value of the transmission process changes since every long message is divided into several small and independent fragments. As a result, more SIFS frames are needed during the transmission. In addition, the retransmission of the data fragments and the ACK frames are required to be carried out immediately in the extended reserved time if errors occur during their transmission. Therefore, more ACK frames will be appeared during the transmission process, which directly affects the evaluation of the collision probability in the SMAC protocol.

Furthermore, as a result of the introduction of the periodic listen and sleep, the collision probability not merely depends on the relationship between the transmission rate and the sampling rate, but also depends on the duty cycle. For example, if the duty cycle equals to 1, which is the extreme situation indicating that there is no sleep mode in SMAC, then either the saturated case or the unsaturated case occurs. In this case, the collision probability merely depends on the relationship between the transmission rate and the sampling rate. Please note that the duty cycle can never be zero since otherwise the sensor nodes would always stay in the sleep mode and thus can never ever to manage their data transmission and reception. However, if the duty cycle is larger than zero but smaller than one, both the listen and sleep period exist. During the sleep period, many sampled data packets will be accumulated in every node’s queue. And they will be transmitted saturately in the very first part of the next listen period. However, from some time on, when there are no data packets stocked in the queue any more, the newly sampled packets will be transmitted unsaturately. In order to accurately evaluate the overall collision probability in SMAC, the sampling rate, the transmission rate and the duty cycle should be all taken into account in order to develop the average method to set up the overall collision probability model based on the collision probability in both the saturated and unsaturated period. And this average scheme will be deeply described in the next subsection.

5.23 Overall collision probability model

For a certain sampling rate, even if it’s much smaller than the transmission rate, the saturated period will still occur during the listen phase since a lot of data packets will be accumulated in the queue of every sensor node during their previous sleep period. Therefore, instead of considering the collision probability in the saturated and in the unsaturated period independently, we should average these two collision probabilities into an overall collision probability. Such average method should be operated based on the total number of data packets which are transmitted during the saturated and unsaturated period respectively since packets are the objects of collision.

During the sleep period, since all the sensor nodes will turn off their radio, no data packets will be transmitted. As a result, the collision could never occur during this time. Therefore, we should only focus on the listen phase to analyze the overall average collision probability. The
character $p_s$ and $p_u$ are adopted to denote the collision probability in the saturated and unsaturated period respectively. In the meanwhile, the symbol $n_s$ and $n_u$ are used to represent the total number of data packets which are transmitted during the saturated and unsaturated period within the listen phase respectively. Then the overall collision probability $p$ can be described as the arithmetic average of the collision probability in the saturated and unsaturated period.

$$p = p_s, p_u$$  \hspace{1cm} (34)

As we discussed before, the average scheme should be handled based on the number of data packets since collision only occur on the simultaneously transmitted packets. As a result, we should use the ratio of the total number of data packets transmitted during the saturated period to the total number of data packets transmitted during the whole listen phase as the weight for the collision probability in the saturated period. In the same manner, we will choose the ratio of the total number of data packets transmitted during the unsaturated period to the total number of data packets transmitted during the whole listen phase as the weight for the collision probability in the unsaturated period.

$$p = p_s \frac{n_s}{n_s + n_u} + p_u \frac{n_u}{n_s + n_u}$$  \hspace{1cm} (35)

This formula can be simplified and transformed as:

$$p = \frac{p_s n_s + p_u n_u}{n_s + n_u}$$  \hspace{1cm} (36)

Since the sampling work never stops during the whole frame, the newly sampled data packets are constantly arrived at the queue of each sensor node during all the synchronized, saturated and sleep phase. Therefore, the total number of data packets transmitted during the saturated period equals to the number of data packets accumulated during the previous sleep period $T_{sleep}$ plus the number of data packets sampled during the current synchronized interval and the saturated period. Thus, the total number of data packets transmitted during the saturated period $n_s$ is related with the sampling rate $R_s$, the sleep time $T_{sleep}$, the saturated period $T_{saturated}$ and the synchronized interval $T_{sync}$.

$$n_s = \frac{R_s T_{sleep} + R_s (T_{saturated} + T_{sync})}{N_{data}}$$  \hspace{1cm} (37)

As expressed in the section 4.2, from the perspective of an arbitrary sensor node, synchronization is only maintained periodically, which is accomplished by transmitting a SYNC
packet after the carrier sense in the beginning of the listen phase. Moreover, different from the saturated period during which many data packets are transmitted, there is only one small SYNC frame sent in the synchronization interval, which indicates the synchronization period is much shorter comparing with the saturated period. Hence, the total data packets sampled during the synchronization period is too small to be counted. So for simplicity and efficiency reasons, we decide to ignore the poor packet number contribution made by the synchronization interval. Thus, the above formula (37) is should be transferred to:

$$n_s = \frac{R_s T_{\text{sleep}} + R_s T_{\text{saturated}}}{N_{\text{data}}}$$

Where the sleep phase $T_{\text{sleep}}$ is able to be expressed by the frame time $T_{\text{frame}}$ and the duty cycle $D$:

$$T_{\text{sleep}} = T_{\text{frame}} (1 - D)$$

Since $T_{\text{saturated}}$ is the parameter whose value we can’t determine directly, the number of the data packets transmitted during the saturated period can not be easily evaluated. Therefore, we need to analyze this total number in a different way. Since the saturated period $T_{\text{saturated}}$ is the summation of the time interval $T_0$ the $T_1$, total number of data packets transmitted during the saturated period $T_{\text{saturated}}$ equals to the number of data packets transmitted during $T_0$ plus the number of data packets transmitted during $T_1$. As discussed before, the data packets transmitted during the current $T_0$ comes from the data packets sampled during the previous sleep phase. Thus, the number of the data packets transmitted during $T_0$ equals to the number of data packets sampled during the sleep phase.

$$\frac{R_s T_{\text{sleep}}}{N_{\text{data}}}$$

In the SMAC protocol, the message passing mechanism is adopted in order to reduce the retransmission overhead, so every long message will be divided into several small and independent fragments and they will be transmitted in burst. Therefore, the size of a sampled data packet $N_{\text{data}}$ equals to the summation of the size of all its separated fragments $N_f$. The relationship between the data packet and the data fragment is showed in the formula (41).

$$N_{\text{data}} = nN_f$$
Therefore, the formula (40) can be transferred to:

\[ \frac{R_s T_{\text{sleep}}}{nN_f} \]  

(42)

During the time \( T_0 \), the packets are transmitted saturately, which however does not mean that the time interval between every two neighbor packets is zero, where packets indicate all types of messages. The reason of this phenomenon is because there is a DIFS sensing period, a randomly chosen backoff time and many SIFS frames exist during each packet transmission, shown in the figure 12. As a result, when we consider the total time used for transmitting the data packets which are sampled during the sleep period, we should exclude those time spent on the DIFS, the backoff time and the SIFS frames in each transmission. So the total backoff time during \( T_0 \) equals to the single backoff duration times the number of data packets sampled during the previous sleep phase.

\[ T_{\text{avg}} \sigma \frac{R_s T_{\text{sleep}}}{nN_f} \]  

(43)

Similarly, the total time used for the DIFS sensing period during \( T_0 \) is able to be evaluated.

\[ T_{\text{DIFS}} \frac{R_s T_{\text{sleep}}}{nN_f} \]  

(44)

As explained before, without taking the retransmission of data fragments and ACK frames into account, there are \((2n+1)\) SIFS frames in all appearing during each round of the transmission where one SIFS frame stands between every two neighbor packets. So the total time spent on the SIFS frames during \( T_0 \) is:

\[ (2n + 1)T_{\text{SIFS}} \frac{R_s T_{\text{sleep}}}{nN_f} \]  

(45)

Since the error rate which depends on the reliability of the wireless sensor network is usually very small, the time used for the retransmission of control packet is not worth to be taken into consideration here. So the total time spent on the transmission of the control packet RTS, CTS and ACK frames during \( T_0 \) is:

\[ \left( \frac{N_{\text{rts}}}{R_t} + \frac{N_{\text{cts}}}{R_t} + n \frac{N_{\text{ack}}}{R_t} \right) \frac{R_s T_{\text{sleep}}}{nN_f} \]  

(46)
By combining the formula (43) (44) (45) and (46), the total time which is not directly used for the packet transmission during $T_0$ can be concluded as:

$$
T_{avg} + T_{DIFS} + (2n + 1)T_{SIFS} + \frac{N_{\text{rx}}}{R_t} + \frac{N_{\text{cts}}}{R_t} + n\frac{N_{\text{ack}}}{R_t} R_i T_{sleep} \frac{n N_f}{n N_f}
$$

As a consequence, subtracting this useless time from $T_0$, the time which is actually spent in transmitting only the data packets which are sampled during the sleep period can be estimated as:

$$
t_0 - \frac{\left( T_{avg} + T_{DIFS} + (2n + 1)T_{SIFS} + \frac{N_{\text{rx}}}{R_t} + \frac{N_{\text{cts}}}{R_t} + n\frac{N_{\text{ack}}}{R_t} \right) R_i T_{sleep} \frac{n N_f}{n N_f}}{R_t}
$$

Obviously, all the data packets sampled during the previous sleep period will be transmitted by using the transmission rate during this remaining time of $T_0$. Therefore, we can set up an equation by making use of the relationship among the number of data packets sampled during the sleep time, the transmission rate and the remaining time of $T_0$.

$$
t_0 - \frac{\left( T_{avg} + T_{DIFS} + (2n + 1)T_{SIFS} + \frac{N_{\text{rx}}}{R_t} + \frac{N_{\text{cts}}}{R_t} + n\frac{N_{\text{ack}}}{R_t} \right) R_i T_{sleep} \frac{n N_f}{n N_f}}{R_t} = \frac{R_i T_{sleep}}{R_t}
$$

The data packets transmitted during $T_0$, which is the second part of the saturated period, are sampled during the whole saturated period $T_{\text{saturated}}$. Likewise, there are some time within $T_i$ which are spent on the carrier sense, the SIFS frames and the transmission of the control packet rather than directly used for the transmission of the data packets. Thus, in order to get rid of such time period we are not interested in, this valueless time duration should be firstly evaluated.

$$
\left( T_{avg} + T_{DIFS} + (2n + 1)T_{SIFS} + \frac{N_{\text{rx}}}{R_t} + \frac{N_{\text{cts}}}{R_t} + n\frac{N_{\text{ack}}}{R_t} \right) R_i T_{saturated} \frac{n N_f}{n N_f}
$$

Then the useful time period during $T_1$ which only concentrates on the data packet transmission is able to be estimated, where $T_1$ can be obtained by subtracting $T_0$ from $T_{\text{saturated}}$.

$$
T_{\text{saturated}} - t_0 - \left( T_{avg} + T_{DIFS} + (2n + 1)T_{SIFS} + \frac{N_{\text{rx}}}{R_t} + \frac{N_{\text{cts}}}{R_t} + n\frac{N_{\text{ack}}}{R_t} \right) R_i T_{saturated} \frac{n N_f}{n N_f}
$$
Since all the data packets sampled during the saturated period $T_{saturated}$ will be transmitted by suing the transmission rate during this remaining time of $T_1$, we can set up another equation which expresses the relationship among the number of data packets sampled during the saturated period $T_{saturated}$, the transmission rate $R_t$, and such remaining time of $T_1$.

\[
T_{saturated} - t_0 = (T_{avg} \sigma + T_{DIFS} + (2n + 1)T_{SIFS} + \frac{N_{rs} + N_{ex} + nN_{ack}}{R_t}) + \frac{R_s T_{saturated}}{nN_f} = \frac{R_s T_{saturated}}{R_t}
\]  

Delightedly, we can combine the above equation (49) with (52) and transfer them into an equation set. Then we are able to successfully obtain the value of the saturated period simply by solving this equation set.

\[
T_{saturated} = \left[ T_{avg} \sigma + T_{DIFS} + (2n + 1)T_{SIFS} + \frac{N_{rs} + N_{ex} + nN_{ack}}{R_t} \right] + \frac{R_s T_{saturated}}{nN_f} - \frac{R_s T_{sleep}}{R_t}
\]

Up to now, since we have already found out the value of the complicated parameter $T_{sleep}$ and $T_{saturated}$ in the function $n_s$, the value of the total number of data packets transmitted during the saturated period can be successfully acquired. In order to fully develop the value of the overall average collision probability, we still need to look for the value of the total number of data packets which are transmitted during the unsaturated period within the listen phase. Since the packets which are transmitted during the unsaturated period are all from the packets which are sampled during the unsaturated period, it’s easy to evaluate the number $n_u$. 

\[
n_u = \frac{R_s T_{unsaturated}}{N_{data}}
\]

As shown in the figure 8, the unsaturated period equals to the frame time excluding the sleep phase and the saturated period.

\[
T_{unsaturated} = T_{frame} - T_{sleep} - T_{saturated}
\]

Thus, the formula (54) can be transferred to:

\[
n_u = \frac{R_s (T_{frame} - T_{sleep} - T_{saturated})}{N_{data}}
\]
By inserting the formula (38) and (56) into the formula (36), we can finally obtain the equation which evaluates the overall collision probability, where the parameter $T_{saturated}$ is the most complicated variable affecting the value of the overall average collision probability $p$. 

$$p = p_s \left( \frac{R_s T_{\text{sleep}} + R_s T_{\text{saturated}}}{N_{\text{data}}} \right) + p_u \left( \frac{R_s T_{\text{frame}} - R_s T_{\text{sleep}} - R_s T_{\text{saturated}}}{N_{\text{data}}} \right)$$

$$= \frac{R_s T_{\text{frame}}}{N_{\text{data}}}$$ (57)

The above formula (57) can be simplified as:

$$p = \frac{p_s (T_{\text{sleep}} + T_{\text{saturated}}) + p_u (T_{\text{frame}} - T_{\text{sleep}} - T_{\text{saturated}})}{T_{\text{frame}}}$$ (58)

If we insert the expression formula (53) which estimates the length of the saturated period $T_{saturated}$ into the above formula (58), with delight, we will eventually find out the value of the overall collision probability which averages the collision probability in the saturated and unsaturated period.

$$p_s \left( \frac{\left( T_{\text{avg}} + T_{DIFS} + (2n+1)T_{SIFS} + \frac{R_s}{R_i} N_{\text{rs}} + N_{\text{cs}} + nN_{\text{ack}} \right) \frac{R_s T_{\text{sleep}}}{nN_f} + \frac{R_s T_{\text{sleep}}}{R_i}}{1 - \frac{R_s}{R_i} \left( T_{\text{avg}} + T_{DIFS} + (2n+1)T_{SIFS} + \frac{R_s}{R_i} N_{\text{rs}} + N_{\text{cs}} + nN_{\text{ack}} \right) \frac{R_s}{nN_f}} \right)$$

$$p_u \left( \frac{\left( T_{\text{avg}} + T_{DIFS} + (2n+1)T_{SIFS} + \frac{R_s}{R_i} N_{\text{rs}} + N_{\text{cs}} + nN_{\text{ack}} \right) \frac{R_s T_{\text{sleep}}}{nN_f} + \frac{R_s T_{\text{sleep}}}{R_i}}{1 - \frac{R_s}{R_i} \left( T_{\text{avg}} + T_{DIFS} + (2n+1)T_{SIFS} + \frac{R_s}{R_i} N_{\text{rs}} + N_{\text{cs}} + nN_{\text{ack}} \right) \frac{R_s}{nN_f}} \right)$$

(59)

In conclusion, the collision probability in the saturated period $p_s$ is a constant value which does not rely on the time interval $T$. On the contrary, the collision probability in the unsaturated period $p_u$ acts as different values when the time interval $T$ belongs to different conditions. Therefore, the overall average collision probability $p$ which is established in terms of these two collision probabilities occurred during the saturated and unsaturated period respectively will also behave differently depending on the value of $T$. As a result, the overall average collision probability $p$ will be able to be finally obtained by solving the following equation sets under the different four conditions which are classified according to the value of $T$. 

37
\[ 1. T \in (0, \sigma) \]

\[
N_{\text{neighbor}} = \left[ \frac{N}{\alpha \times b} \times \pi R^2 \right] - 1
\]

\[
T_{\text{avg}} = \left[ 1 - \frac{1}{1 - \left( \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}}}} \right] - 1
\]

\[
p_s = 1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}}}
\]

\[
M = N_{\text{neighbor}}
\]

\[
p_u = \frac{M}{N_{\text{neighbor}}} \left[ 1 - \left( \frac{1}{T_{\text{avg}}} \right)^{M-1} \right]
\]

\[
T_{\text{avg}} = \left[ 1 - \frac{1}{1 - \left( \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}}}} \right] - 1
\]

\[
p_s T_{\text{frame}} (1 - D) + \left( \frac{T_{\text{avg}}}{R_i} \sigma + T_{\text{DIFS}} + (2n+1)T_{\text{SIFS}} + \frac{N_{\text{rts}} + N_{\text{cts}} + nN_{\text{ack}}}{R_i} \right) \frac{R_{sT_{\text{sleep}}} + R_{T_{\text{sleep}}}}{R_{i}}
\]

\[
p = \left( \frac{T_{\text{frame}}}{R_i} \right)
\]

\[
p_u T_{\text{frame}} D - \left( \frac{T_{\text{avg}}}{R_i} \sigma + T_{\text{DIFS}} + (2n+1)T_{\text{SIFS}} + \frac{N_{\text{rts}} + N_{\text{cts}} + nN_{\text{ack}}}{R_i} \right) \frac{R_{sT_{\text{sleep}}} + R_{T_{\text{sleep}}}}{R_{i}}
\]

\[
= \left( \frac{T_{\text{frame}}}{R_i} \right)
\]

(60)
\[ 2. T \in [\sigma, T_{\text{avg}} \sigma) \]

\[
N_{\text{neighbor}} = \left[ \frac{N}{ab} \pi R^2 \right] - 1
\]

\[
T_{\text{avg}} = \left[ \frac{N_f}{N_{\text{neighbor}}} \right] \left( 1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}}} \right)
\]

\[
p_s = 1 - \left( \frac{T_{\text{avg}} + 2}{T_{\text{avg}} + 2 + \frac{T}{\sigma}} \right)^{N_{\text{neighbor}}} - 1
\]

\[
T = \frac{n N_f}{R_s} - \frac{N_{\text{ack}}}{p_{\text{aj}} p_{\text{aj}}} \frac{N_f}{R_s} - \frac{R_s}{p_{\text{aj}}} - (2n + 1) T_{\text{SIFS}} - T_{\text{DIFS}} - T_{\text{avg}}
\]

\[
p_u = \frac{M}{N_{\text{neighbor}}} \left( 1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{M-1} \right)
\]

\[
T_{\text{avg}} = \left[ \frac{1 - \left( 1 - \left( \frac{1}{T_{\text{avg}}} \right)^{M-1} \right)}{1 - \frac{1}{p_{\text{u}}} \left( \frac{2 N_{\text{ack}}}{R_s} \right) \left( 2^{\frac{1}{N_{\text{ack}}} \left( \frac{1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{M-1} \right)} - \frac{1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{M-1} \left( \frac{1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{M-1} \right)}{2^{\frac{1}{N_{\text{ack}}} \left( \frac{1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{M-1} \right)}} \right) \right) \right) \right)
\]

\[
p = \left( \frac{T_{\text{frame}}}{T_{\text{avg}}} \right) (1 - D) + \left( \frac{T_{\text{avg}} + T_{\text{DIFS}} + (2n + 1) T_{\text{SIFS}} + \frac{N_{\text{rx}} + N_{\text{cs}} + n N_{\text{ack}}}{R_s}}{R_{\text{rx}}} \frac{R_{T_{\text{sleep}}}}{n N_f} + \frac{R_{T_{\text{sleep}}}}{R_t} \right)
\]

\[
p_u \left( \frac{T_{\text{frame}}}{T_{\text{avg}}} \right) (1 - D) \left( \frac{T_{\text{avg}} + T_{\text{DIFS}} + (2n + 1) T_{\text{SIFS}} + \frac{N_{\text{rx}} + N_{\text{cs}} + n N_{\text{ack}}}{R_s}}{R_{\text{rx}}} \frac{R_{T_{\text{sleep}}}}{n N_f} + \frac{R_{T_{\text{sleep}}}}{R_t} \right)
\]

\[
p_u \left( \frac{T_{\text{frame}}}{T_{\text{avg}}} \right) (1 - D) \left( \frac{T_{\text{avg}} + T_{\text{DIFS}} + (2n + 1) T_{\text{SIFS}} + \frac{N_{\text{rx}} + N_{\text{cs}} + n N_{\text{ack}}}{R_s}}{R_{\text{rx}}} \frac{R_{T_{\text{sleep}}}}{n N_f} + \frac{R_{T_{\text{sleep}}}}{R_t} \right)
\]
\[ T \in [T_{\text{avg}} \sigma, T_{\text{avg}} \sigma + T_{\text{DIFS}} + T_{\text{trans}}] \]

\[
N_{\text{neighbor}} = \left\lfloor \frac{N}{a \times b} \times \pi R^2 \right\rfloor - 1
\]

\[
T_{\text{avg}} = \left[ \frac{(1-p_s) \left( \frac{CW_{\min}(1-(2p_u)^{m+1})}{2(1-2p_s)} \right) + \left( \frac{p_u^{m+1} 2^m CW_{\min}(1-p_u^{K-m-1})}{2(1-p_s)} \right) - \left( \frac{1-p_s^K}{2(1-p_s)} \right)}{1-p_s^K} \right]^{N_{\text{neighbor}}^{-1}}
\]

\[
p_s = 1 - \left( \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}}^{-1}}
\]

\[
M = \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 2} N_{\text{neighbor}}
\]

\[
P_u = \frac{M}{N_{\text{neighbor}}} \left( \frac{1}{1 - \left( \frac{1}{T_{\text{avg}}} \right)} \right)^{M-1}
\]

\[
T_{\text{avg}} = \left[ \frac{(1-p_u) \left( \frac{CW_{\min}(1-(2p_u)^{m+1})}{2(1-2p_s)} \right) + \left( \frac{p_u^{m+1} 2^m CW_{\min}(1-p_u^{K-m-1})}{2(1-p_s)} \right) - \left( \frac{1-p_u^K}{2(1-p_s)} \right)}{1-p_u^K} \right]^{N_{\text{neighbor}}^{-1}}
\]

\[
p = \frac{T_{\text{frame}}(1-D) + \left( \frac{T_{\text{avg}} + T_{\text{DIFS}} + (2n+1)T_{\text{SIFS}} + N_{\text{rx}} + N_{\text{ctn}} + nN_{\text{ack}}}{R_t} \right) \frac{R_s T_{\text{sleep}}}{nN_f} + \frac{R_s T_{\text{sleep}}}{R_i}}{1 - \frac{R_s}{R_i} - \left( \frac{T_{\text{avg}} + T_{\text{DIFS}} + (2n+1)T_{\text{SIFS}} + N_{\text{rx}} + N_{\text{ctn}} + nN_{\text{ack}}}{R_t} \right) \frac{R_s}{nN_f}}
\]

\[
P_u = \frac{T_{\text{frame}} - \left( \frac{T_{\text{avg}} + T_{\text{DIFS}} + (2n+1)T_{\text{SIFS}} + N_{\text{rx}} + N_{\text{ctn}} + nN_{\text{ack}}}{R_t} \right) \frac{R_s T_{\text{sleep}}}{nN_f} + \frac{R_s T_{\text{sleep}}}{R_i}}{1 - \frac{R_s}{R_i} - \left( \frac{T_{\text{avg}} + T_{\text{DIFS}} + (2n+1)T_{\text{SIFS}} + N_{\text{rx}} + N_{\text{ctn}} + nN_{\text{ack}}}{R_t} \right) \frac{R_s}{nN_f}}
\]

\[
(62)
\]
4. \( T \in [T_{\text{DIFS}} + \sigma T_{\text{avg}} + T_{\text{trans}}, +\infty) \)

\[
N_{\text{neighbor}} = \left\lfloor \frac{N}{ab\pi R^2} \right\rfloor - 1
\]

\[
T_{\text{avg}} = \left[ \frac{(1 - p_s) \left( \frac{CW_{\text{min}} (1 - (2p_s)^{m+1})}{2(1 - 2p_s)} \right) + \left( p_u^{m+1} 2^n CW_{\text{min}} (1 - p_u^{K-m-1}) \right) }{1 - p_u^K} \right]^{-1}
\]

\[
p_s = 1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}}}
\]

\[
M = \left[ \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 3} \right] \left( \frac{T - T_{\text{DIFS}} - \sigma T_{\text{avg}} - T_{\text{trans}}}{\sigma} \right) N_{\text{neighbor}}
\]

\[
T = \frac{nN_f}{R_s} - \frac{N_{\text{cts}}}{R_t} - n \frac{R_s}{P_{fj} P_{aji}} - n \frac{R_t}{P_{aji}} - (2n + 1)T_{SIFS} - T_{\text{DIFS}} - T_{\text{avg}}
\]

\[
T_{\text{trans}} = \frac{N_{\text{cts}}}{R_t} + \frac{nN_f}{R_j} + \frac{nN_f}{R_i} P_{fj} P_{aji} + \frac{nN_f}{R_i} P_{aji} + (2n + 1)T_{SIFS}
\]

\[
p_u = \frac{M}{N_{\text{neighbor}}} \left( \frac{1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{M-1}}{1} \right)
\]

\[
T_{\text{avg}} = \left[ \frac{(1 - p_u) \left( \frac{CW_{\text{min}} (1 - (2p_u)^{m+1})}{2(1 - 2p_u)} \right) + \left( p_u^{m+1} 2^n CW_{\text{min}} (1 - p_u^{K-m-1}) \right) }{1 - p_u^K} \right]^{-1}
\]

\[
p_s T_{\text{frame}} (1 - D) + \left( T_{\text{avg}} \sigma + T_{\text{DIFS}} + (2n + 1)T_{SIFS} + \frac{N_{\text{cts}} + N_{\text{cts}} + nN_{\text{ack}}}{R_t} \right) \frac{R_s T_{\text{sleep}}}{nN_f} + \frac{R_j T_{\text{sleep}}}{R_t}
\]

\[
p = \frac{T_{\text{frame}}}{T_{\text{frame}}}
\]

\[
p_u T_{\text{frame}} D = \left( T_{\text{avg}} \sigma + T_{\text{DIFS}} + (2n + 1)T_{SIFS} + \frac{N_{\text{cts}} + N_{\text{cts}} + nN_{\text{ack}}}{R_t} \right) \frac{R_s T_{\text{sleep}}}{nN_f} + \frac{R_j T_{\text{sleep}}}{R_t}
\]

\[
+ \left( 1 - \frac{R_s}{R_t} \right) \left( T_{\text{avg}} \sigma + T_{\text{DIFS}} + (2n + 1)T_{SIFS} + \frac{N_{\text{cts}} + N_{\text{cts}} + nN_{\text{ack}}}{R_t} \right) \frac{R_s}{nN_f}
\]

\[
(63)
\]
5.3 Overlap neighbor model

In the following sections 5.6 and 5.6, we will consider the energy consumed in the extra RTS and CTS overhearing and the extra sleep time in the situation where control packet is applied. The average neighbor node, as stated before, is one of the key parameters which affect the evaluation of such energy consumption. However, the average neighbor node only fits for the situation where neighbor nodes either belong to the transmitter or belong to the receiver. In other words, the average neighbor node model only makes sense if among all the neighbor nodes of the transmitter, none of which are also the neighbor nodes of the receiver.

However, it’s not always the case that the neighbor nodes of the transmitter and the receiver are independent of each other. The fundamental reason is because the communication radio of two immediate neighbor nodes, for example the transmitter and the receiver, will always intersect. But the overlap of the communication radio will not surely indicate the overlap of the neighbors of two immediate nodes. Whether the neighbor nodes intersect or not and the average number of the neighbor nodes which intersect depend on the density of the wireless sensor network as well as the overlap area of the communication radio of a sensor node.

If the phenomenon of the neighbor intersection occurs, the average neighbor node model is not enough to evaluate the exact energy consumption of the extra overhearing and the extra sleep time in the situation where the control packet RTS and CTS is activated. Let’s take the extra energy consumption of the overhearing as an example. Since overhearing is in fact receiving, after the reception energy of the extra RTS frame of one neighbor node of the transmitter is successfully evaluated, by simply multiplying by the total number of the neighbor nodes of the transmitter, we will obtain the extra overhearing energy of the RTS frame from the side of the neighbors of the transmitter. After receiving the RTS frames which are not destined to them, all the neighbor nodes of the transmitter will go to sleep mode until the whole transmission process completes. As a result, they are not able to transmit or receive any data packet in the middle until they wake up. However, after the receiver gets the RTS frame, it will send back the CTS frame to notify the successful reception. As soon as the CTS frame is transmitted, not only the transmitter, but also all the neighbors except for those overlap neighbor nodes which are also the neighbors of the transmitter will receive such CTS frame if no errors occur during the transmission. The reason why the overlap neighbor nodes lose the capability of overhearing here is because they have already turned off their radio and thus know nothing. Therefore, instead of directly using the average neighbor model in this case, we should only estimate the extra energy consumption of the CTS overhearing by applying the left neighbor nodes of the receiver which exclude those overlap neighbor nodes. And this is the reason we must build up the overlap neighbor model in this section.
As discussed just now, the number of the overlap neighbor nodes which are both the neighbors of the transmitter and the receiver is related with the intersection of the radio communication coverage of two random immediate neighbors. When two nodes become neighbors, their transmission circles intersect, in which case the figure 13 shows the two extreme scenarios. If the Euclidean distance between the transmitter and the receiver is assumed as \( d \) where \( 0 \leq d \leq R \), the possible maximum value of \( d \) equals to the radio range \( R \) while the minimum value of \( d \) is zero. Therefore, the radio intersection area of two neighbor nodes \( S_{\text{int/sect}} \) can be described as the parameters \( d \) and \( R \) [25], presented in the following formula (64).

\[
S_{\text{int/sect}} = 2R^2 \cos^{-1}\left(\frac{d}{2R}\right) - \frac{1}{2}d\sqrt{4R^2 - d^2}
\]  

Taking the assumption of the random Poisson distribution of the nodes into account, then \( d \) is bound in \([0, R]\) with a uniform probability distribution. Accordingly, the average distance between two neighbor nodes will be:

\[
d_{\text{avg}} = \frac{R}{2}
\]  

If the formula (65) is inserted into the above formula (64), the average overlap of the communication radio of two neighbor nodes can be described as:

\[
S_{\text{avg}}^{\text{int/sect}} = 2.152R^2
\]  

Besides the parameter of the radio intersection area of two neighbor nodes, another variable which will influence the evaluation of the average number of the overlap neighbor nodes is the density of the wireless sensor network. As analyzed in the section 5.2 when we are building up the average neighbor node model for a transmitter or a receiver, we successfully estimated the density of the network in the assumption that the total number of the sensor nodes in the network is \( N \) and the size of the wireless sensor network is \( a \times b \) where \( a < b \).
\[ \hat{\lambda} = \frac{N}{a \times b} \] (67)

We have already known the average overlap of the radio range of two neighbor nodes and the density of the wireless sensor network. By simply multiplying together this two parameters, showed by the formula (66) and (67), the average overlap nodes is able to be evaluated. Since in the intersection area, both the transmitter and the receiver are enclosed in, so we should exclude two nodes from all the nodes which stand in this overlap area.

\[ N_{\text{avg \ int \ sect}}^{\text{int \ sect}} = S_{\text{avg \ int \ sect}}^{\text{int \ sect}} \times \hat{\lambda} - 2 \] (68)

After combination and simplification, the average intersection neighbor nodes of the transmitter and the receiver can be written as:

\[ N_{\text{avg}}^{\text{int \ sect}} = \frac{2.152NR^2}{ab} - 2 \] (69)

Up to now, we have already successfully found out the overall average collision probability of the SMAC protocol theoretically as well as the average number of the intersection neighbor nodes. As a result, it’s the time now to evaluate the extra energy consumption that control packet brings if we activate control packet in the SMAC protocol. There are mainly four sources of the extra energy consumption comparing with the situation without control packet, one comes from the extra transmission, retransmission and reception of the control packet RTS and CTS, another one derives from the extra overhearing of the control packet RTS and CTS by both the neighbors of the transmitter and the receiver. The third one springs from the extra sleep time of the neighbors of the transmitter and the receiver and the last is caused by the extra radio mode switching. So in the following four sections, we will set up the corresponding analytic energy model according to each extra energy consumption.

5.4 Energy model of extra control packet RTS and CTS

In this section, we develop energy consumption model which takes into the account of the extra energy consumption due to the extra introduced control packet RTS and CTS. This extra energy consumption exists in both the saturated period and the unsaturated period within the listen phase comparing with the situation deactivating the control packet. In the SMAC protocol if control packet is employed, collision happens only during the transmission of the RTS, so only the RTS frame faces the problem of retransmission caused by collision. Moreover, both of the RTS and CTS frames might be corrupted during the transmission due to the error occurrence such as the breakdown of the wireless network, packet destroys by network attackers and packet loss. Thus, Energy spent on the retransmission of the RTS and CTS packets caused by collision and error will account for a significant part in the total energy consumption.

Besides the 802.11 protocol, the SMAC protocol should also be studied under the Hop-by-Hop Retransmission scheme (HHR). Instead of quantifying the total extra energy consumption control packet brings all over the network, we prefer to effectively evaluate the extra
energy consumed along a route in the network. Such evaluation is divided into two parts. In the first part, we will calculate the energy consumption of the control packet RTS and CTS between any two neighboring communication sensor nodes, which is implemented by classifying the energy model into two parts. One is focused on the estimation of the average transmission and retransmission energy consumed in the control packet RTS and CTS between two nodes; the other one will be concentrated on the average receiving energy consumption of the control packet RTS and CTS between such two nodes. By adding these two values together, the extra average communication energy consumption of the control packet RTS and CTS between this pair of the node is able to be obtained. For simplicity, we do not take the routing mechanism into account. So we only consider the simplest routing approach, that is, we only focus on one dimensional path from a source node to the sink to analyze the energy consumption. This one line path is actually sufficient for the comparison of the trade-off between the energy consumption that control packet brings and the energy consumption of the arising troubles if control packet is deactivated.

Before analysis, we draw a state diagram for transmitting a data packet from the node i to one of its neighbors, say, the node j, which is showed in the following figure 14, where S0 is the initial state, from which the node j is able to successfully receive the RTS frame with the probability \((1-p_{rij})(1-p)\). Here \(p_{rij}\) denotes the error probability of the RTS during the transmission from the node i to j and p denotes the overall average collision probability. Since in the SMAC protocol collisions could occur only during the RTS frame, collision probability is only related to the RTS frame. S2 is the state in which the node i successfully receives the CTS frame from the state S1 with probability \((1-p_{cji})\), where \(p_{cji}\) represents the error probability of the CTS sent from the node j to i. S3 is the state in which the node j successfully receives the first data fragment from the state S2 with the probability \((1-p_i)\), where \(p_i\) means the error probability of the data fragment between the node i and j. We assume all the fragments of a data message share the same error probability. And S4 is the state in which the node i successfully receives the corresponding ACK frame of the first fragment from the node j with the probability \((1-p_{aji})\), where \(p_{aji}\) stands for the error probability of the ACK packet between the state S3 and S4. In the same way, all the ACK frames are assumed to have the same error probability. And this state diagram continuous until the transmission of the final \(n_{th}\) fragment and its corresponding \(n_{th}\) ACK frame successfully completes.

For simplicity and efficiency reasons, we denote \((1-p_{roj}),(1-p),(1-p_{cj}), (1-p_{of})\) as well as \((1-p_{aji})\) by \(p_{roj}^*, p^*, p_{cj}^*, p_{of}^*\) and \(p_{aji}^*\) respectively.

![Fig.14. State diagram for the SMAC protocol](image)

Usually, control packet is sent at a higher power level than the normal data fragments for dealing with the hidden terminal problem to guarantee reliable transmission. We assume the
transmission power used for transmitting control packet and the normal data fragment is $P_{tx}$ and $P_{ij}$ respectively, where $P_{tx} > P_{ij}$. Although the transmission power of control packet and data fragment is different, the receiving power level for them is the same, denoted by $P_{rx}$. In addition, we denote the packet size of the RTS, the CTS, a data fragment and the ACK by $N_{rts}$, $N_{cts}$, $N_{f}$ and $N_{ack}$ respectively. In the meantime, we use $\bar{x}$ to represent the mean value of the variable $x$.

### 5.4.1 Transmission energy model of control packet RTS and CTS

Our purpose in this section is to look for the difference between the energy consumed in the transmission if control packet is applied and the energy consumed in the transmission if control packet is abandoned. In the former case, the energy is used for the transmission of the RTS frame, retransmission of the RTS frame due to error, retransmission of the RTS frame due to collision, transmission of the CTS frame, retransmission of the CTS frame due to error, transmission of the data fragments, retransmission of the data fragments due to error, transmission of the ACK frames and retransmission of the ACK frames due to error. In the latter case, however, the energy is used for the transmission of the data fragments, retransmission of the data fragments due to error, retransmission of the data fragments due to collision, transmission of the ACK frames and retransmission of the ACK frames due to error. By comparison of these two cases, the extra energy consumption if control packet is activated is the transmission energy of the RTS frame, retransmission energy of the RTS frame due to error, retransmission energy of the RTS frame due to collision, transmission energy of the CTS, retransmission energy of the CTS due to error. The extra energy consumption if control packet is deactivated is, however, the retransmission energy of the first data fragment due to collision. As a result, when we consider the extra energy in the situation with control packet, we should not take into account of the energy consumed in the data fragments and ACK frames.

In order to evaluate the average transmission energy in sending the control packet RTS successfully between the node i and its neighbor the node j, we should firstly consider the transmission power needed to successfully transmit the RTS frame from the node i to j, that is, the transmission power from S0 to S1. The formation of this transmission energy formula will be accomplished based on the figure 14.

$$
\frac{P_{tx,rts}}{P_{ij} P_{cji}} = \frac{P_{tx}}{P_{ij} P_{cji}}
$$

(70)

During this transmission, we do not only consider the power consumed in the successful transmission of the RTS frame, but also consider the power consumed in the retransmissions of the RTS frame due to both collision and packet errors, where both $P_{rr}$ and $P_{col}$ are in the range of (0,1). As I analyzed in my small thesis, in the 802.11 protocol, if the transmission of the ACK frame fails, all of the RTS, CTS and data packet must be retransmitted. In other words, the validness of the RTS transmission depends on the successful transmission of the following CTS, data packet and ACK frame. That’s the reason we also add all the items $p_{cji}$, $P_{ij}$, $P_{cji}$ in the denominator of the fraction. However, in the SMAC protocol, since the message passing mechanism is developed, the transmitter needs only to extend the reserved transmission time for one more fragment and
retransmits this current fragment immediately if it fails to receive the ACK frame. In other words, a failed transmitted fragment or ACK frame will not cause the retransmission of the previous RTS and CTS anymore. Therefore, besides considering the collision probability and error probability during the RTS transmission itself, we should only add the item $P_{\text{cji}}$ in the denominator when estimating the power consumption of the RTS frame.

To further calculate the transmission energy needed for the successful transmission of the RTS frame from the node $i$ to $j$, we just simply multiply the transmission power by the time used for sending the RTS frame, which is $\frac{N_{\text{rts}}}{R_i}$, where $R_i$ is the transmission rate in the wireless sensor network. Thus, the average transmission energy of the RTS frame from the state $S_0$ to $S_1$ is:

$$E_{\text{tx,rts}} = \frac{P_{\text{tx}} N_{\text{rts}}}{R_i} \frac{1}{P_{\text{rj}} P_{\text{cji}}}$$ (71)

Similarly, we are able to get the formulas showing the transmission energy of the CTS.

$$E_{\text{tx,cts}} = \frac{P_{\text{tx}} N_{\text{cts}}}{R_i} \frac{1}{P_{\text{cji}}}$$ (72)

In conclusion, the average transmission energy in sending the control packet RTS and CTS between the node $i$ and its neighbor node $j$ is given in the following formula (73), where the first item is the transmission and the retransmission energy consumed in the RTS from the state $S_0$ to $S_1$ and the second item is the transmission and the retransmission energy consumed in the CTS from the state $S_1$ to $S_2$. And both of these two items only focus on the transmission energy between the node $i$ and its neighbor node $j$.

$$E_{\text{tx}}(i,j) = \frac{P_{\text{tx}} N_{\text{rts}}}{P_{\text{rj}} P_{\text{cji}}} + \frac{P_{\text{tx}} N_{\text{cts}}}{P_{\text{cji}}}$$ (73)

5.42 Reception energy model of control packet RTS and CTS

Similarly with the method of the analysis of the transmission energy model, the reception energy model is only concentrated on the energy consumed in receiving the RTS and CTS packets. Moreover, the reception of the RTS frame makes sense on the condition that the CTS frame is successfully received by either the node $j$. Thus, the average reception energy model of the RTS and CTS frames are formulated respectively as below.
\[
\overline{E}_{rx,rts} = \frac{P_{rx} \frac{N_{rs}}{R_i}}{p_{cji}}
\]

(74)

\[
\overline{E}_{rx,cts} = P_{rx} \frac{N_{cts}}{R_i}
\]

(75)

Therefore, by adding these two reception energy formulas, the total average reception energy in successfully receiving the control packet RTS and CTS between the node i and its neighbor node j is obtained.

\[
\overline{E}_{rx}(i,j) = \frac{P_{rx} \frac{N_{rs}}{R_i}}{p_{cji}} + P_{rx} \frac{N_{cts}}{R_i}
\]

(76)

5.43 Total energy model of control packet RTS and CTS

To compute the total energy of the control packet RTS and CTS between the node i and the node j which includes both the transmission energy and the reception energy, we merely need to add the formula (73) and (76) together. In this way, we can obtain the average total energy consumption during the communication of the control packet RTS and CTS between the node i and j:

\[
\overline{E}(i,j) = \overline{E}_{tx}(i,j) + \overline{E}_{rx}(i,j)
\]

(77)

Since \( \overline{E}(i,j) \) is only the energy of the control packet RTS and CTS between two one-hop nodes. In order to compute the total energy consumption of the control packet RTS and CTS along a path in the wireless sensor network, we should add such energy consumption in every hop along the path from the source node to the sink. We assume the average number of hop in a route in a wireless sensor network is M, that is, the average number of node in a route is M+1. Then we can evaluate the average total energy consumption of the control packet RTS and CTS along the path from the source node (0) to the sink (M) in a wireless sensor network.

\[
\overline{E}_{total} = \sum_{i=0}^{M-1} (\overline{E}_{tx}(i,i+1) + \overline{E}_{rx}(i,i+1))
\]

(78)

To facilitate the understanding of the differentiation between \( \overline{E}(i,j) \) and \( \overline{E}_{total} \) showed in the formula (77) and (78) respectively, we use the following figure 15 to help to visually explain such difference.
In the formula $E_{\text{Total}}$, all the parameters are simple, that is, they can be easily assigned a defined value except for the collision probability. However, fortunately, we have already built up the collision probability model where the value of the overall average collision probability can be worked out in theory by the supporting of the average neighbor model, the T model, the active neighbor node model and the average collision probability model. Therefore, we could successfully calculate the final energy consumed in the communication of the control packet RTS and CTS from the source node to the sink node theoretically by just inserting one of the corresponding formulas of the collision probability depending on the different conditions. That is, if we combine each of the overall average collision probability formulas (60), (61), (62) (63) in each classification with the formula (78), the total average energy consumption of the control packet RTS and CTS in a one dimensional routing is able to be acquired of the SMAC protocol.

Consequently, we have already successfully gained one of the extra energy consumption if control packet is used in the SMAC protocol. Besides the extra overhead of the control packet RTS and CTS, another arising phenomenon which also brings extra energy consumption is the overhearing of the control packet RTS and CTS by the neighbor nodes of both the transmitter and receiver. In the next section, we will specifically focus on analyzing this extra overhearing energy consumption.

5.5 Energy model of extra overhearing of RTS and CTS

Besides the extra energy consumed in the transmission, retransmission, reception of the RTS and CTS packets, in the medium access mechanism of the SMAC protocol where the control packet is applied, another source of the extra energy consumption comes from the overhearing of the RTS and CTS packets by the neighbors of the transmitters and receivers along a path. In SMAC, the control packet RTS and CTS frames both carry the information of the length of the transmission which is determined by the total size of the packets being transmitted and the transmission rate. This information can be read by any listening sensor node, which is then able to know the period of time that the channel will remain busy. Therefore, when a neighbor node is hidden from either the transmitter or the receiver, by detecting just one frame among the RTS and CTS packets, it can suitably delay their own transmission by going into sleep mode. Thus, if the medium access mechanism with control packet is applied, the overhearing would only occur in the
RTS and CTS frames. And if the medium access mechanism without control packet is used, the overhearing in this case would occur during the transmission of the first data fragment and the corresponding first ACK frame since each data fragment and ACK packet also has the duration field inside which can prevents the neighbor nodes of the transmitter and the receiver listening for the whole transmission process. So after comparison, the extra overhearing energy consumption comes from the RTS and CTS frames in the situation with control packet and the first data fragment and the first ACK frame in the situation without control packet.

Since overhearing means a node receives the packet which is not destined to it. The fact of the overhearing is the receiving of the packets by the neighbors of the transmitter or the receiver. Therefore, if control packet is adopted, the energy consumed in the overhearing is the energy consumed in receiving the RTS and CTS packets by the neighbors of the transmitter and the receiver respectively.

Firstly, we consider the energy of overhearing between the node \( i \) and its neighbor node \( j \) within a path. The energy of overhearing the transmitted RTS frame by the neighbors of the transmitter \( i \) can be evaluated as:

\[
E_{\text{overhear}}^{\text{rts}}(i, j) = P_{\text{rx}} \frac{N_{\text{rts}}}{R_i} N_{\text{neighbor}}
\]  

(79)

Possibly, there are some neighbor nodes which are both the neighbors of the transmitter and the receiver, so these overlap neighbor nodes will turn off their radio after they overhear the RTS frame transmitted by the sender. As a result, they can’t overhear the CTS frame transmitted by the receiver since they have already gone to sleep. So the energy of overhearing the transmitted CTS frame by the remaining neighbors of the receiver \( j \) should be evaluated as:

\[
E_{\text{overhear}}^{\text{cts}}(i, j) = P_{\text{rx}} \frac{N_{\text{cts}}}{R_t} (N_{\text{neighbor}} - N_{\text{overlap}})
\]  

(80)

By summing up together the above formula (79) and (80), the total average energy of the overhearing of the RTS and CTS frames between the node \( i \) and its immediate neighbor node \( j \) is able to be obtained.

\[
E_{\text{overhear}}^{\text{total}}(i, j) = P_{\text{rx}} \frac{N_{\text{rts}}}{R_i} N_{\text{neighbor}} + P_{\text{rx}} \frac{N_{\text{cts}}}{R_t} (N_{\text{neighbor}} - N_{\text{overlap}})
\]  

(81)

Then the total average extra energy of the RTS and CTS overhearing along the path from the source (node 0) to the sink (node \( M \)) is calculated as:

\[
E_{\text{overhear}}^{\text{total}} = \sum_{i=0}^{M-1} E_{\text{overhear}}^{\text{total}}(i, i+1)
\]  

(82)

As we may already observed, the difference between \( E_{\text{overhear}}^{\text{total}}(i, j) \) and \( E_{\text{overhear}}^{\text{total}} \) is that
the overhearing energy of the control packet RTS and CTS between the node i and its neighbor node j is only a small portion of the total overhearing energy of the such control packet RTS and CTS frames along a path, which is visually described as the following figure 16.

![Fig.16. The difference between $E_{\text{overhear}}^{(i,j)}$ and $E_{\text{total}}$](image.png)

5.6 Energy model of extra sleep time

As we stated before, different from the 802.11 protocol, in the SMAC protocol, the NAV vector in each packet does not only play the role of delaying the transmission of those neighbor nodes which fail in the competition of the channel reservation, but also prevents them from overhearing. So after a node receives a packet which is not destined to it, by reading the value of the NAV vector inside, it will turn off the radio and go to the sleep mode until the NAV timer reaches zero. In the situation with the control packet RTS and CTS, all the neighbor nodes of the transmitter will go to sleep after they overhear the RTS frame and all the neighbor nodes of the receiver will also go to sleep after they overhear the CTS frame. All these neighbor nodes of both the transmitter and the receiver will wake up when the final ACK frame is successfully transmitted. In the situation without the control packet RTS and CTS, however, all the neighbor nodes of the transmitter will go to sleep after they overhear the first data fragment rather than the RTS frame and all the neighbor nodes of the receiver will go to sleep after they overhear the first ACK frame instead of the CTS frame. As a result, the neighbor nodes of the transmitter and the receiver will sleep for a longer time $(T_{\text{CTS}} + T_{f1} + 2T_{\text{SIFS}})$ and $(T_{f1} + T_{\text{ack1}} + 2T_{\text{SIFS}})$ respectively in the situation with control packet comparing with the sleep time in the situation without control packet, described in the figure 17 (a) and (b). And this extra sleep time exist and is changeless no matter it is considered in the saturated or in the unsaturated period within the listen phase of the SMAC protocol.
Fig. 17. The difference of the sleep time between the situation with and without control packet

In the same manner, we firstly quantify the extra sleep period of the neighbor nodes between the node i and its neighbor node j, then we will add up this extra sleep period in every hop along a path. From the above figure 17, we can observe that in the situation with control packet, besides the extra sleep time of the CTS frame and the first data fragment, the neighbor nodes of the transmitter i sleep for $2T_{SIFS}$ longer. Likewise, the neighbor nodes of the receiver j sleep for $2T_{SIFS}$ longer besides the extra sleep period spent on the first data fragment and the first ACK frame after they hear the CTS packet. Therefore, the extra sleep time of the neighbor nodes of the transmitter i can be evaluated as:

$$
\frac{N_{cts} + N_{f1}}{R_i} + 2T_{SIFS}
$$

(83)

In the same way, the extra sleep time of the neighbor nodes of the receiver j can be estimated as:

$$
\frac{N_{f1} + N_{ack1}}{R_i} + 2T_{SIFS}
$$

(84)

So we can easily formulate the average extra energy caused by the longer sleep time of the neighbor nodes of the transmitter i in the situation with control packet as

$$
E_{i}^{\text{sleep}} = P_{\text{sleep}} \left( \frac{N_{cts} + N_{f1}}{R_i} + 2T_{SIFS} \right) N_{\text{neighbor}}
$$

(85)
And similarly, since some neighbor nodes of the transmitter are also the neighbors of the receiver, after these overlap neighbor nodes receive the RTS frame transmitted by the sender, they will turn off their radio. Hence, when we consider the extra sleep time from the perspective of the neighbors of the receiver, we should exclude those overlap neighbor nodes from all the neighbors of the receiver. So the average extra energy which is brought about by the longer sleep time of the valid neighbor nodes of the receiver $j$ should be evaluated as:

$$E_{\text{sleep}}^j = P_{\text{sleep}} \left( \frac{N_{f_1} + N_{\text{ack}_1}}{R_j} + 2T_{\text{SIFS}} \right) (N_{\text{neighbor}} - N_{\text{overlap}})$$ (86)

So if we add up the above formula (85) and (86), we will obtain the total average extra energy consumed in the longer sleep time of both the neighbor nodes of the transmitter $i$ and the receiver $j$ in the situation with control packet between the node $i$ and its neighbor node $j$.

$$\overline{E}_{\text{total}}(i, j) = P_{\text{sleep}} \left( \frac{N_{f_1} + N_{\text{ack}_1}}{R_i} + 2T_{\text{SIFS}} \right) N_{\text{neighbor}} + P_{\text{sleep}} \left( \frac{N_{f_1} + N_{\text{ack}_1}}{R_j} + 2T_{\text{SIFS}} \right) (N_{\text{neighbor}} - N_{\text{overlap}})$$ (87)

Here, $\overline{E}_{\text{total}}(i, j)$ is only the extra sleep energy between two one-hop nodes. In order to compute the total extra energy consumption caused by the longer sleep time of the neighbor nodes of the transmitters and the receivers along a path in the wireless sensor network, we should add such extra energy consumption in every hop along the path from the source node to the sink. We assume the average number of hops in a route in a wireless sensor network is $M$, so the average number of nodes in a route is $M+1$. Then we can successfully evaluate the average total extra energy consumed in the longer sleep time of the neighbor nodes of the transmitters and the receivers along the path from the source node (0) to the sink (M) in a wireless sensor network.

$$\overline{E}_{\text{total}} = \sum_{j=0}^{M-1} \overline{E}_{\text{total}}(i, i+1)$$ (88)

Obviously, every parameter in the formula (88) is simple which can be easily assigned a value. In order to give a better understanding of the relationship between $\overline{E}_{\text{total}}(i, j)$ and $\overline{E}_{\text{total}}$, we use the following figure 18 for visualization. As we can see, the energy consumption of the extra sleep time between the node $i$ and its neighbor node $j$ $\overline{E}_{\text{total}}(i, j)$ is only a small part of the total extra sleep energy consumed along the path $\overline{E}_{\text{total}}$. 
In the 802.11 protocol, since the sensor nodes do not have the sleep mode, they keep in the listen status all the time. Therefore, all the sensor nodes in the wireless sensor network do not need to change the power status from sleep to active or from active to sleep. Due to the non-sleep property of the 802.11 protocol, significant amount of energy is consumed in the idle listening. In order to avoid the idle listening which is the main source of the energy cost, the SMAC protocol is optimized by introducing the duty cycle as well as the overhearing avoidance.

The duty cycle reduces the unnecessary idle time between every two neighbor transmission processes, which make use of the fact that the sampling rate is far smaller than the transmission rate. As a result, the same number of the data packets is able to be transmitted but use less period of the time. The overhearing avoidance scheme, however, reduces the unnecessary idle time of all the neighbor nodes of the transmitter and the receiver within every transmission process. That is, all the neighbor nodes of the transmitter will go to sleep mode after they overhear the RTS frame in the situation with control packet or the first data fragment in the situation without control packet, and all the neighbor nodes of the receiver will go to sleep mode after they receive the CTS frame in the situation with control packet or the first ACK frame in the situation without control packet. And all these neighbors will wake up when a round of the transmission completes. However, another source of the energy consumption will be brought in if the overhearing avoidance mechanism is adopted in the SMAC protocol, which is the radio switching among different modes. So whether to employ overhearing avoidance technology seems to be a trade off.

Specifically speaking, switching the radio from sleep mode to reception mode or from sleep mode to transmission mode consumes energy. In the situation where control packet is applied, if we assume that the errors would not occur during the transmission of any packet. Then after all the neighbor nodes of the transmitter overhear the RTS frame, they will change the power status from the reception mode to the sleep mode. In the same way, after all the left neighbor nodes of the receiver overhear the corresponding CTS frame, they will also switch the power status from the reception mode to the sleep mode, where left means that all the neighbor nodes of the receiver excludes those nodes which are also the neighbors of the transmitter. As soon as a round of the transmission process finishes, all the neighbor nodes of both the transmitter and the receiver will wake up and change the power status from the sleep mode back to the reception mode. As we can
observe, in the normal cases, during a transmission process, the phenomenon of the radio mode switching occurs and only occurs there times, as the following figure 19 presents.

Similarly, in the situation where control packet is deactivated, if no retransmission occurs in the middle due to the errors of the data packets, after all the neighbor nodes of the transmitter overhear the first data fragment, they will change the power status from the reception mode to the sleep mode, and after all the left neighbor nodes of the receiver receive the first ACK frame which are not destined to them, they will switch the power status from the reception mode to the sleep mode as well. And all these neighbor nodes will wake up and thus change the power status from the sleep mode back to the reception mode after the last ACK frame is successfully received by the transmitter. Likewise, in the normal cases, the phenomenon of the radio mode switching appears and only appears there times within a round of the transmission process.

In some special cases, if the errors occur during the transmission of some data packets no matter in the situation with or without control packet, after the time which is informed by the NAV vector in the RTS and CTS frame or in the first data fragment and the first ACK frame decreases to zero, the transmission process has not finished yet, however, all the neighbor nodes of the transmitter and the receiver has already turned on the radio. Therefore, both the neighbor nodes of the transmitter and the receiver will overhear one more data packet respectively, only after which they will go to sleep mode again until the new NAV vector inside these two data packets decreases to zero again. This process keeps going until the transmission process finally completes. As a result, the times of the radio mode switching is no longer three in each transmission round, instead, the times will be more than three due to the retransmission of the data packets. The fundamental
reason why the radio mode switching is more than three times is because of the development of the message passing mechanism. In the message passing scheme, every time a data fragment is transmitted, the sender waits for an ACK from the receiver. If it fails to receive the ACK frame, it will extend the reserved transmission time for one more fragment, and retransmit the current fragment immediately. So if the transmission is extended due to fragment losses or errors, the sleeping neighbors will not be aware of the extension immediately. However, the nodes will learn it from the extended data fragments or ACK frames when they wake up.

Since the error probability is usually very low, the probability of the retransmission occurrence of the data packets in the middle of the transmission is also very low. So in most and usual cases, all the data fragments and ACK frames during the transmission will not face the problem of the retransmission. As a consequence, the times of the radio mode switching during a round of the transmission is three in both situations no matter the control packet RTS and CTS is activated or not as we analyzed in the previous paragraphs. So we could draw the conclusion that the times of the radio mode switching has nothing to do with the length of a transmission process or the number of the packets sent during a round of the transmission. On the contrary, it's only related with the number of the data packets transmitted within a certain time period.

As the figure 20 describes above, in the \( T_0 \) period of the listen phase, the total number of the data packets transmitted is the same for both of the situations with and without control packet even though each single transmission size in the situation without control packet is a little smaller than the transmission size in the situation with control packet due to the extra RTS and CTS frames. During both the \( T_0 \) periods, the number of the data packets transmitted equals to the
number of the data packets accumulated during the previous sleep phase.

\[
T_{\text{sleep}} = \frac{R_s T_{\text{saturated}}}{N_{\text{data}}} 
\]

Since all these parameters \( R_s, T_{\text{sleep}} \) and \( N_{\text{data}} \) are the same in both situations with and without control packet, \( n_s \) remains the same in both situations. However, during the \( T_1 \) period of the saturated phase, since the size of a transmission in the situation with control packet is a little larger than that in the situation without control packet, a sensor node in the situation with control packet needs more time to transmit the same number of the data packets. However, during such transmission period, the newly sampled data packets will reach the queue continuously, since the time period needed to transmit the same number of the data packets is larger, the period \( T_1 \) in the situation with control packet will be larger than that in the situation without control packet. Since the saturated period will only completes if all the accumulated sampled data packets in the queue are all sent out, the number of the data packets transmitted during \( T_1 \) in the situation with control packet \( n_{T_1}^{\text{with}} \) will be larger than that in the situation without control packet \( n_{T_1}^{\text{without}} \), where \( n_{T_1}^{\text{with}} > n_{T_1}^{\text{without}} \).

\[
n_{T_1}^{\text{with}} = \frac{R_s T_{\text{saturated}}^{\text{with}}}{N_{\text{data}}} 
\]

\[
n_{T_1}^{\text{without}} = \frac{R_s T_{\text{saturated}}^{\text{without}}}{N_{\text{data}}} 
\]

And this is the exact source of the extra energy consumption of the radio mode switching in the situation with control packet. Deserve to be mentioned, the radio mode switching does not occur instantaneously. For example, the RFM radio on the test bed needs 20 \( \mu \)s to switch from sleep mode to reception mode [26], as the \( T_{\text{down}} \) and \( T_{\text{up}} \) indicate in the following figure 21. Therefore, it is desirable to reduce the frequency of switching modes. The message passing scheme tires to put nodes into sleep state as long as possible, and hence reduces switching overhead.
Fig. 21. Radio switches between the reception mode and the sleep mode

So in the situation where control packet RTS and CTS is activated, after all the neighbors of the transmitter overhear the RTS frame, they will go to sleep mode by changing their radio from the reception mode to the sleep mode. And such radio switching energy consumption equals to the area of the left yellow triangle times the total number of the neighbor nodes of the transmitter.

\[ E_1 = S_{\text{triangle}}^{\text{left}} \times N_{\text{neighbor}} \quad (92) \]

Where the area of the left yellow triangle can be calculated as:

\[ S_{\text{triangle}}^{\text{left}} = \frac{1}{2} T_{\text{down}} (P_{\text{reception}} - P_{\text{sleep}}) \quad (93) \]

If we combine these two formulas (92) and (93) together, the energy consumption of the radio mode switching by the neighbors of the transmitter can be successfully evaluated.

\[ E_1 = \frac{1}{2} T_{\text{down}} (P_{\text{reception}} - P_{\text{sleep}}) N_{\text{neighbor}} \quad (94) \]

In the same manner, we could evaluate the extra energy consumption of the radio mode switching from the reception status to the sleep status by the left neighbors of the receiver, which equals to the area of the left yellow triangle in the figure 21 times the total number of the left neighbor nodes of the receiver, since those neighbor nodes which are both the neighbors of the transmitter and the receiver have already changed from the reception mode to the sleep mode after they overhear the RTS frame. As a result, the extra energy consumption of the radio mode switching by the neighbor nodes of the receiver can be estimated as:

\[ E_2 = \frac{1}{2} T_{\text{down}} (P_{\text{reception}} - P_{\text{sleep}}) (N_{\text{neighbor}} - N_{\text{overlap}}) \quad (95) \]

When the vector inside the RTS and the CTS frame received by the neighbor nodes of the
transmitter and the receiver respectively decreases to zero, both the neighbor nodes of the transmitter and the receiver will wake up and thus change the radio from the sleep mode back to the reception mode. And the extra energy consumption of the radio switching from the sleep mode to the reception mode by all these neighbors equals to the area of the right yellow triangle in the figure 21 times the total valid neighbors of the transmitter and the receiver.

\[ E_3 = S_{\text{triangle}}^{\text{right}} N_{\text{total}}^{\text{neighbor}} \]  

(96)

Where the area of the right yellow triangle equals to the area of the left yellow triangle since the time that radio changes from the reception mode to the sleep mode \( T_{\text{down}} \) is the same as the time that radio changes from the sleep mode to the reception mode \( T_{\text{up}} \).

\[ S_{\text{triangle}}^{\text{right}} = \frac{1}{2} T_{\text{up}} (P_{\text{reception}} - P_{\text{sleep}}) \]  

(97)

And the total valid neighbors of the transmitter and the receiver is the summation of the total number of the neighbor nodes of the transmitter and the receiver, however, excludes those overlap neighbors which are the neighbor nodes of both the transmitter and the receiver.

\[ N_{\text{total}}^{\text{neighbor}} = 2 N_{\text{neighbor}} - N_{\text{overlap}} \]  

(98)

So if we insert the formulas (97) and (98) into the previous formula (96), the extra energy consumption of the radio mode switching from the sleep mode to the reception mode by all the valid neighbors of the transmitter and the receiver will be able to be finally estimated.

\[ E_3 = \frac{1}{2} T_{\text{up}} (P_{\text{reception}} - P_{\text{sleep}})(2 N_{\text{neighbor}} - N_{\text{overlap}}) \]  

(99)

Therefore, just by adding together the energy consumption \( E_1, E_2 \) and \( E_3 \), we will obtain the total energy consumption of the radio mode switching by the neighbors of the transmitter and the receiver for a round of the transmission, that is the transmission between the transmitter \( i \) and its immediate receiver \( j \).

\[ E = E_1 + E_2 + E_3 \]  

(100)

The formula \( E \) can be finally solved out if we combine the previous formulas (94) and (95) and (99). After the transformation, the formula \( E \) can be written as:

\[ E = \frac{1}{2} (P_{\text{reception}} - P_{\text{sleep}})(T_{\text{down}} N_{\text{neighbor}} + T_{\text{down}} (N_{\text{neighbor}} - N_{\text{overlap}}) + T_{\text{up}} (2 N_{\text{neighbor}} - N_{\text{overlap}})) \]  

(101)

Comparing with the situation without control packet, the extra number of the data packets being transmitted in the situation with control packet only appears during the \( T_1 \) period of the
saturated phase, so we should average the energy consumption $E$ by adding a coefficient in front of the item. Since the total energy consumption of the radio mode switching is only related with the number of the data packets being transmitted, such coefficient should be developed in terms of the total number of the data packets transmitted during $T_1$ in both situations with and without control packet and the total number of the data packets transmitted during the whole frame.

$$a = \frac{R_s T_{\text{saturated}}}{N_{\text{data}}} - \frac{R_s T_{\text{without saturated}}}{N_{\text{data}}}$$

The numerator of the fraction represents the extra data packets transmitted during $T_1$ in the situation with control packet comparing with the data packets transmitted during $T_1$ in the situation without control packet. And the denominator describes the total data packets transmitted during the whole frame which is the same for both the situations with and without control packet. After simplification, the above formula can be transferred to:

$$a = \frac{T_{\text{saturated}}}{T_{\text{frame}}} - \frac{T_{\text{without saturated}}}{T_{\text{frame}}}$$

So if we substitute the concrete formula (53) which presents the saturated duration in the situation with control packet $T_{\text{saturated}}$ and formula (134) which shows the saturated period in the situation without control packet $T_{\text{without saturated}}$ (which will be analyzed later), the final result of the coefficient $a$ will be able to be successfully evaluated.

$$a = \frac{R_s T_{\text{sleep}}}{R_t} \left( 1 - \frac{T_{\text{saturated}}}{T_{\text{frame}}} \right)$$

So if we add this coefficient $a$ in front of the item in the formula (101). The weighted
average extra energy consumption of the radio switching between the sleep mode and the reception mode of the neighbors of the transmitter and the receiver will be successfully calculated. Such weighted average energy consumption is only analyzed between the transmitter i and the immediate receiver j within a routing.

\[ E_{\text{radio switching}}(i, j) = \frac{1}{2} (P_{\text{reception}} - P_{\text{sleep}}) (2T_{\text{down}} N_{\text{neighbor down}} - T_{\text{down}} N_{\text{overlap}} + 2T_{\text{up}} N_{\text{neighbor up}} - T_{\text{up}} N_{\text{overlap}}) \]  

(105)

Assume there are together M+1 nodes in an one dimensional routing in the wireless sensor network, if we add this extra energy consumption of the radio mode switching in every hop along the path from the source node (node 0) to the sink (node M), the total extra energy consumption of the radio mode switching by the neighbors of the transmitter and the receiver in the situation with control packet will be eventually evaluated.

\[ E_{\text{radio switching}} = \sum_{i=0}^{M-1} E_{\text{radio switching}}(i, i+1) \]  

(106)

In order to make the understanding of the relationship between the energy consumption of the extra radio mode switching in one hop \( E_{\text{radio switching}}(i, j) \) and the energy consumption of the extra radio mode switching for an entire routing \( E_{\text{radio switching}} \) more clearly, we prefer to use the following figure 22 to illustrate. From it, we will easily figure out that the former energy overhead contributes only a small part but essential to the total radio switching energy consumption of the whole routing.

![Fig.22. The relationship between \( E_{\text{radio switching}}(i, j) \) and \( E_{\text{radio switching}} \)](image)

6 Energy Model of Arising Consequences Without Control Packet

Comparing with the medium access technique of SMAC with control packet, the basic access mechanism of SMAC is carried out without control packet, where the control packet mainly refers to the RTS and CTS frame. In the medium access approach of the SMAC protocol where control packet is activated, the RTS and CTS frame carry the information of the length of packets needed to be transmitted. Since each sensor node has the ability to obtain the current
transmission rate of the channel, the information that how long the remaining transmission process will be can be realized and such information is recorded in a variable called network allocation vector (NAV) within the RTS and CTS frame. Therefore, when a station is hidden from either the transmitting node or the receiving node, by detecting the NAV value in control packet, it can suitably delay its own transmission by going into sleep mode. More concretely, when the neighbor nodes of the transmitter hear the RTS frame which destined to send to the destination node, they know how long they should turn off their radio. In the same principle, when the neighbor nodes of the receiver hear the CTS frame, they get the idea that how long they must keep sleeping.

In the meanwhile, each data fragment and ACK frame also has the duration field. In this way, if a node wakes up or a new node joins in the network in the middle of the transmission, it can properly go to sleep no matter it is the neighbor of the transmitter or the receiver. For example, if the sender extends the transmission time due to fragment losses or errors, the sleeping neighbor node will not be aware of the extension immediately. However, the node will learn it from the NAV vector in the extended fragments or ACK frames when it wakes up. As a result, if the medium access mechanism without control packet is applied, the first data fragment and the first ACK frame also have the capability of avoiding the overhearing of the following transmission. That means the overhearing would still occur during the transmission of the first data fragment and ACK frame, which is one of the main differences from the situation where control packet is activated. In short, not only the RTS and the CTS frames, but also the data fragments and the ACK packets take effect of delaying the transmission, avoiding the hidden terminal problem and preventing overhearing of the transmission process.

Besides these capabilities, the RTS and CTS frames are able to decrease the overhead of the retransmission if collision occurs. The collision, instead of occurring in the data fragment, it only occurs in the RTS frame if control packet is introduced. Since collision takes place in different types of packets if different medium access mechanism is applied, the corresponding retransmission overhead is different if collision appears. Although the transmission power of control packet is more than that of the data fragment, the packet size of the RTS frame, however, is much smaller than that of the data fragment. As a result, the trade off of the retransmission overhead depends on the length of the packet in which collisions take place and the necessary power for transmitting such packet. Even though, it’s difficult to judge which access mechanism is better with respect to the retransmission overhead in theory, we are intuitively aware that the retransmission overhead of the RTS frame should be less comparing with that of the data fragment if control packet is not employed, for the much larger size of the fragment can sufficiently compensate its smaller transmission power as the retransmission overhead due to collision is a function which has direct proportion to the product of the transmission power and the size of the packet which faces collision. Otherwise, the import of control packet does not make much sense from the perspective of the retransmission cost.

Very importantly, the RTS and CTS frames in the IEEE 802.11 protocol do not have the capability of overhearing avoidance, since there is no sleep mode in such protocol, that is, the sensor nodes never turn off their radio. That’s the exact reason that all the neighbor nodes of the sender and the receiver will only delay their transmission rather than going to sleep after they read
the information in the NAV which is kept in the RTS and CTS frame. On the contrary, the control packet in SMAC has the ability of overhearing avoidance. And thus, neighbor nodes will go to sleep after they read the NAV counter until the communication completes.

Since the RTS and CTS frame is eliminated from the transmission in the medium access mechanism of SMAC where control packet is deactivated, the size of the whole transmission is minimized. And as we analyzed before, all the neighbor nodes will go to sleep mode after they hear the NAV value within the first fragment and the first ACK frame. As a result, the sleep duration of all the neighbor nodes in the situation without control packet is decreased due to such minimized transmission length. Such shorter sleep duration, however, will cause the extra idle listening time when the whole transmission is completed but the next data packet hasn’t come yet. More concretely, this phenomenon of the extra energy consumption will arise only in the unsaturated situation of the listen period where T does not equal to zero.

In the following sections, we are going to calculate the energy consumed in the retransmission of the data fragment from a source node to the sink. And as we analyzed before when we consider the energy consumption of the control packet, the energy consumption is a function of the error probability, the collision probability and some other parameters. Since the error rate only depends on the reliability of the wireless network, the error probability is packet length independent. So we assume it has the same value for all types of packet. In addition, since the collision probability in the unsaturated situation is a function of the packet size, if the access mechanism is used where the RTS and CTS frame are eliminated from the whole transmission, the collision probability will change correspondingly. Hence, as a whole, the overall average collision probability will vary as well.

### 6.1 Collision probability model

In the medium access mechanism with control packet, the collision probability, as already been analyzed, in the saturated situation, is a function of the minimum contention window, the average number of neighbor nodes, the maximum backoff stage, the maximum transmission attempt and the overall average backoff time. However, in the unsaturated situation, besides all these parameters, the collision probability is a function of the slot time, DIFS time, SIFS time, transmission rate, sampling rate and packet size as well. Therefore, if the access mechanism without control packet is used, the collision probability remains the same in the saturated situation as its value is packet size independent. However, in the unsaturated situation of the SMAC protocol, the packets being transmitted are only the data fragments and the ACK frames, thus the packet size is decreased. As a result, the time needed for transmitting such packets is changed. Therefore, the value of the corresponding collision probability is changed. Hence, on average of the saturated and unsaturated situation, the overall collision probability is changed.

In the following subsections, the analysis of the collision probability will be made in detail and new collision probability model will be set up accordingly.
6.11 In saturated period

In the saturated situation of the SMAC protocol, the average number of active transmitting node equals to the average number of neighbor node. So the collision probability \( p \) is only a function of the overall average backoff time \( T_{\text{avg}} \), the average number of neighbor node \( N_{\text{neighbor}} \), the maximum backoff stage \( m \), the minimum contention window \( CW_{\text{min}} \) and the maximum transmission attempt \( K \), where \( N_{\text{neighbor}} \) is also a function which is dependent of the total number of sensor node \( N \) in the wireless network, the network area \((a \times b)\) and the radio transmission range \( R \).

As discussed before, the collision probability in the saturated period of the listen phase has nothing to do with the packet size. Therefore, no matter what type of packets and how much the size of packets is used for transmission, the collision probability remains the same value. Thus, no matter whether control packet is employed for the medium reservation, the collision probability is changeless. Hence, the equation set (6) which calculates the collision probability of the RTS frame in the saturated situation where the medium access scheme with control packet is adopted is also suitable to be applied to the medium access mechanism where control packet is eliminated. And such equation set is recalled as below.

\[
\begin{align*}
T_{\text{avg}} &= \left(1 - p\right)\left(\frac{CW_{\text{min}}\left(1 - (2p)^{m+1}\right)}{2(1 - 2p)}\right) + \left(\frac{p^{m+1}2^m CW_{\text{min}}(1 - p^{K-m-1})}{2(1 - p)}\right) - \left(\frac{1 - p^K}{2(1 - p)}\right) \\
N_{\text{neighbor}} &= N \left(\frac{a \times b}{\pi R^2}\right) - 1 \\
p &= 1 - \left(1 - \frac{1}{T_{\text{avg}}}\right)^{N_{\text{neighbor}}}
\end{align*}
\]  

(107)

By resolving such equation set, we are able to obtain the value of the collision probability in the saturated period where the medium access mechanism only adopting the data fragment and the ACK frame is applied.

6.12 In unsaturated period

The only difference between the access mechanism with control packet and without control packet in the unsaturated period is that two more packets RTS and CTS needed to be transmitted
when using the access mechanism with control packet. Specifically speaking, the only difference
lies in the number of packets and the total packet size. Thus, collision occurs in the different type
of packets. In the situation with control packet, collision occurs in the transmission of the RTS
frame, and in the situation without control packet, however, collision takes place in the
transmission of the first data fragment. Nevertheless, only few parameters have a change in the
function average number of active transmitting neighbor node M, thus the value of collision
probability p alters since M is a parameter of p. Even though, the idea of formulating the collision
probability model in the unsaturated situation without control packet is kept the same. Therefore,
the formulas developed in calculating the collision probability in the unsaturated period by
employing the access mechanism with control packet can be similarly used here.

6.121 Active neighbor node model

As we previously analyzed, the collision probability in the unsaturated period with control
packet is considered in different conditions with respect to the time interval T, thus the value of the
collision probability is different when it is assayed in different classifications. Recall those
equation sets built for evaluating the collision probability in the situation where control packet is
applied, all of which are transmission time independent except for the fourth one. In the
unsaturated situation with control packet, the transmission time is mainly dependent on the total
packet size $N_{\text{rtx}} + N_{\text{cts}} + \frac{nN_f}{p_{fij}p_{aji}} + \frac{nN_{\text{ack}}}{p_{aji}}$, the time spent on the SIFS frames $(2n+1)T_{\text{SIFS}}$
and the transmission rate $R_i$. Since in the unsaturated period where control packet is deactivated,
the RTS and CTS frame are eliminated, the total packet size in this case is minimized to
$N_f + N_{\text{ack}} + \frac{(n-1)N_f}{p_{fij}p_{aji}} + \frac{(n-1)N_f}{p_{aji}}$ and the corresponding SIFS time is minimized
to $(2n-1)T_{\text{SIFS}}$, as the formula (97) presents.
\[
T_{\text{trans}} = \frac{N_f}{R_i} + \frac{N_{\text{ack}}}{R_i} + \frac{(n-1)N_f}{R_i p_{fij}p_{aji}} + \frac{(n-1)N_f}{R_i p_{aji}} + (2n-1)T_{\text{SIFS}} \quad (108)
\]

Deserve to be mentioned, the total size of the packet being transmitted in the unsaturated
period without control packet can’t be written as $\frac{nN_f}{p_{fij}p_{aji}} + \frac{nN_f}{p_{aji}}$, for every time the first data
fragment is transmitted, the sender waits for an ACK from the receiver, if it fails to receive the
ACK frame, the transmitter will deduce that the first data fragment may be corrupted caused by
collision, or error, or the ACK frame may be unsuccessfully transmitted. Therefore, due to the
uncertainty of the exact reason for the transmission failure, the transmitter has to contend for the
medium once again instead of retransmitting the first data fragment in the extended reserved
transmission time immediately after the unsuccessful transmission attempt.
Therefore, we could directly use the equations of the collision probability evaluated in the unsaturated period with control packet under the first three conditions and develop a new but similar equation under the fourth condition. In conclusion, the collision probability model in the unsaturated period without control packet can be set up as:

1. \( T \in (0, \sigma) \)

\[
M = N_{\text{neighbor}}
\]  

(109)

2. \( T \in [\sigma, T_{\text{avg}} \sigma) \)

\[
M = \frac{T_{\text{avg}} + 2}{T_{\text{avg}} + 2 + \left\lceil \frac{T}{\sigma} \right\rceil} N_{\text{neighbor}}
\]  

(110)

3. \( T \in [T_{\text{avg}} \sigma, T_{\text{avg}} \sigma + T_{\text{DIFS}} + T_{\text{trans}}) \)

\[
M = \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 2} N_{\text{neighbor}}
\]  

(111)

4. \( T \in [T_{\text{DIFS}} + \sigma T_{\text{avg}} + T_{\text{trans}}, +\infty) \)

\[
M = \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 3 + \left\lceil \frac{T - T_{\text{DIFS}} - \sigma T_{\text{avg}} - T_{\text{trans}}}{\sigma} \right\rceil} N_{\text{neighbor}}
\]  

(112)

Where the time for the whole data transmission is:

\[
T_{\text{trans}} = \frac{N_f}{R_t} + \frac{N_{\text{ack}}}{R_t} + \frac{(n-1)N_f}{R_t P_{ij} P_{aj}} + \frac{(n-1)N_f}{R_t P_{aj}} + (2n-1)T_{\text{SIFS}}
\]  

(113)

6.122 T model

Not like the situation where the medium access mechanism with control packet is used, the control packet RTS and CTS frame is eliminated from the transmission if the medium access scheme without control packet is applied. As a result, the total size of the transmission will be minimized. Such smaller transmission length will not lead to any difference in calculating the collision probability in the saturated period between these two access strategies, since in both cases the time interval between the last final ACK frame and the beginning of the new DIFS
sensing period equals to zero. Quite differently, in the unsaturated period, how often a data packet comes (T2 in the figure 23) is the same for both the situations with and without control packet, since the sampling rate is kept constant. However, the time T1 changes due to the minimized transmission length in the assumption that the error probability in both situations with and without control packet remains the same. As a consequence, the time interval T changes accordingly. The following figure 23 is used to illustrate such difference.

Fig.23. T is a function of T1 and T2

So from the figure 23, we could observe that the time period T1 is changed to a smaller length due to the miss of the control packet RTS and CTS frame. And the time spent on the SIFS frame will thus change from \((2n + 1)T_{SIFS}\) to \((2n - 1)T_{SIFS}\).

\[
T_1 = T_{DIFS} + T_{avg} + \frac{N_f}{R_t} + \frac{N_{\text{ack}}}{R_t} + (n - 1) \left( \frac{R_t}{p_{ij} p_{aji}} \right) + (n - 1) \left( \frac{R_t}{p_{aji}} \right) + (2n - 1)T_{SIFS} \tag{114}
\]

And the time duration T2 is unchanged, which depends on the size of the data packet and the sampling rate of the sensor nodes in the wireless sensor network.

\[
T_2 = \frac{N_{\text{data}}}{R_s} \tag{115}
\]

Since in the SMAC protocol, the message passing mechanism is adopted, the data packet is transmitted in the form of the separated several small and independent data fragments. Obviously, the summation of the size of all the data fragments equals to the size of the data packet.

\[
N_{\text{data}} = nN_f \tag{116}
\]

In the situation where the control packet is deactivated, the time interval T between the last final ACK frame and the beginning of the new DIFS sensing period also equals to the value which subtracting T1 from T2.

\[
T = T_2 - T_1 \tag{117}
\]

Therefore, by combining the formulas (114) (115) (116) and (117), the value of the time
interval $T$ is able to be finally evaluated.

$$T = \frac{nN_f}{R_i} - T_{DIFS} - T_{avg}\sigma - \frac{N_{f1}}{R_i} - \frac{N_{ack1}}{R_i} - \frac{(n-1)N_f}{R_i}p_{ji}^{\prime}p_{aji}^{\prime} - \frac{(n-1)N_{ack}}{R_i}p_{aji}^{\prime} - (2n-1)T_{SIFS} \quad (118)$$

In this formula, all the parameters are simple which can be assigned a predefined value except for the overall average backoff time $T_{avg}$ and the sampling rate $R_s$. Since up to now, we have already successfully estimated the value of the average active transmitting neighbor node $M$ and the time interval $T$, we are ready to draw the conclusion of the collision probability in the unsaturated period where the medium access mechanism without control packet is used.

### 6.123 Collision probability model

In the previous section 5.223, we have successfully evaluated the collision probability in the unsaturated period by building up an equation set where the medium access mechanism with control packet is used. Here, that equation set also fits for estimating the collision probability in the unsaturated situation without control packet, for that $p_u$ has nothing to do with the packet size and the transmission time and the developing thought dose not change.

$$p_u = \frac{M}{N_{neighbor}} \left(1 - \left(1 - \frac{1}{T_{avg}}\right)^{M-1}\right)$$

$$T_{avg} = \left\{ \begin{array}{ll}
p_u & \left(1 - p_u\right) \left(\frac{CW_{min}(1 - (2p_u)^{m+1})}{2(1 - 2p_u)}\right) + \left(\frac{p_u^{m+1}CW_{min}(1 - p_u^{K-m-1})}{2(1 - p_u)}\right) - \left(\frac{1 - p_u^K}{2(1 - p_u)}\right)
\end{array} \right\} \quad (119)$$

Nevertheless, the value of the collision probability of the unsaturated period in the situation without control packet must be different from that in the situation with control packet since the average number of the active transmitting neighbor node $M$ does not remain the same due to the variation of the time interval $T$ and the transmission time $T_{trans}$.

Therefore, if we combine the formulas (3) (118) (119) with each of the formulas (109) (110) (111) (112) showing the value of the average number of the active transmitting neighbor node in the unsaturated period in each condition, We are able to finally resolve the collision probability of the first data fragment under different conditions of the unsaturated period of SMAC in theory.

$1. T \in (0, \sigma)$
\[
N_{\text{neighbor}} = \frac{N}{a R^2} - 1
\]
\[
M = N_{\text{neighbor}}
\]
\[
p_u = \frac{M}{N_{\text{neighbor}}} \left(1 - \left(1 - \frac{1}{T_{\text{avg}}}\right)^{M-1}\right)
\]
\[
T_{\text{avg}} = \frac{(1-p_u) \left(\frac{C_{\text{min}}(1-(2p_u)^{m+1})}{2(1-2p_u)}\right) + \left(\frac{p_u^{m+1} 2^m C_{\text{min}}(1-p_u^{K-m-1})}{2(1-p_u)}\right) - \left(\frac{1-p_u^K}{2(1-p_u)}\right)}{1-p_u^K}
\]

2. \( T \in [\sigma, T_{\text{avg}}] \)

\[
N_{\text{neighbor}} = \frac{N}{a R^2} - 1
\]
\[
M = \frac{T_{\text{avg}}+2}{T_{\text{avg}}+2+\frac{T}{\sigma}} N_{\text{neighbor}}
\]
\[
T = \frac{n N_f}{R_s} + T_{\text{DIFS}} - T_{\text{avg}} \frac{N_{\text{ack}}}{R_i} - \frac{(n-1) N_f}{R_i (R_i p_j + R_i p_{aj})} - \frac{(n-1) N_{\text{ack}}}{R_i p_{aj}} - (2n-1) T_{\text{SIFS}}
\]
\[
p_u = \frac{M}{N_{\text{neighbor}}} \left(1 - \left(1 - \frac{1}{T_{\text{avg}}}\right)^{M-1}\right)
\]
\[
T_{\text{avg}} = \frac{(1-p_u) \left(\frac{C_{\text{min}}(1-(2p_u)^{m+1})}{2(1-2p_u)}\right) + \left(\frac{p_u^{m+1} 2^m C_{\text{min}}(1-p_u^{K-m-1})}{2(1-p_u)}\right) - \left(\frac{1-p_u^K}{2(1-p_u)}\right)}{1-p_u^K}
\]

3. \( T \in [T_{\text{avg}}, T_{\text{avg}} + T_{\text{DIFS}} + T_{\text{trans}}] \)
\[ N_{\text{neighbor}} = \left\lfloor \frac{N}{ab} \pi R^2 \right\rfloor - 1 \]

\[ M = \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 2 N_{\text{neighbor}}} \]

\[ p_u = \frac{M}{N_{\text{neighbor}}} \left( 1 - \left( \frac{1}{T_{\text{avg}}} \right)^{M-1} \right) \]

\[ T_{\text{avg}} = \frac{1 - p_u \left( \frac{C W_{\text{min}} \left( 1 - \left( 2p_u \right)^{m+1} \right)}{2\left( 1 - 2p_u \right)} + \left( p_u \right)^{m+1} 2^m C W_{\text{min}} \left( 1 - p_u \right)^{K-m-1} \right)}{1 - p_u^K} \]

4. \( T \in [T_{\text{DIFS}} + \sigma T_{\text{avg}} + T_{\text{trans}}, +\infty) \)

\[ N_{\text{neighbor}} = \left\lfloor \frac{N}{ab} \pi R^2 \right\rfloor - 1 \]

\[ M = \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 3 + \frac{T_{\text{DIFS}} - \sigma T_{\text{avg}} - T_{\text{trans}}}{\sigma}}^{-1} N_{\text{neighbor}} \]

\[ T = \frac{nN_f}{R_s} - T_{\text{DIFS}} \sigma - \frac{N_f}{R_t} - \frac{N_{\text{ackl}}}{R_t} - \frac{(n-1)N_f}{R_t p_{fj}^* p_{ij}^*} - \frac{(n-1)N_{\text{ackl}}}{R_t p_{ij}^*} + (2n-1)T_{\text{SIFS}} \]

\[ T_{\text{trans}} = \frac{N_f}{R_t} + \frac{N_{\text{ackl}}}{R_t} + \frac{(n-1)N_f}{R_t p_{fj}^* p_{ij}^*} + \frac{(n-1)N_{\text{ackl}}}{R_t p_{ij}^*} + (2n-1)T_{\text{SIFS}} \]

\[ p_u = \frac{M}{N_{\text{neighbor}}} \left( 1 - \left( \frac{1}{T_{\text{avg}}} \right)^{M-1} \right) \]

\[ T_{\text{avg}} = \frac{1 - p_u \left( \frac{C W_{\text{min}} \left( 1 - \left( 2p_u \right)^{m+1} \right)}{2\left( 1 - 2p_u \right)} + \left( p_u \right)^{m+1} 2^m C W_{\text{min}} \left( 1 - p_u \right)^{K-m-1} \right)}{1 - p_u^K} \]

Up to now, we have already successfully estimated both the collision probability in the saturated and in the unsaturated period within the listen phase of the SMAC protocol where control packet is removed. So from now on, we must focus on averaging these two values to develop the final overall average collision probability in the situation without control packet.
6.13 Overall collision probability model

In the situation where the medium access mechanism without control packet is used, the overall collision probability will also be evaluated as the weighted average with the help of the total number of data packets transmitted during the saturated and unsaturated period.

\[ p = p_s p_u = \frac{p_s n_s + p_u n_u}{n_s + n_u} \]  \hspace{1cm} (124)

And the total number of data packets transmitted during the saturated period still equals to the number of data packets which accumulated during the previous sleep period plus the number of data packets which are newly sampled during the saturated phase.

\[ n_s = \frac{R_s T_{sleep}}{N_{data}} + \frac{R_s T_{saturated}}{N_{data}} \]  \hspace{1cm} (125)

Here, the sleep period is also dependent on the frame size as well as the duty cycle. As a consequence, the sleep time is the same in both situations with and without control packet.

\[ T_{sleep} = T_{frame} (1 - D) \]  \hspace{1cm} (126)

Since the sleep time remains the same no matter control packet is used or not, the number of data packets accumulated in every node’s queue does not change.

\[ \frac{R_s T_{sleep}}{N_{data}} \]  \hspace{1cm} (127)

However, the time period \( t_0 \) is minimized in the situation without control packet for that each data packet transmitted with the control packet RTS and CTS out of the picture. As a result, the data packets which accumulated during the sleep time can be finished transmitting more quickly comparing with the case in the situation with control packet. During the time \( t_0 \), there are some time which are not directly used for the data fragment transmission, such as the DIFS sensing period, the backoff time, the control packet ACK frames and several SIFS phases.

\[ \left( T_{DIFS} + T_{avg} \sigma + n \frac{N_{ack}}{R_i} + (2n - 1)T_{SIFS} \right) \frac{R_s T_{sleep}}{N_{data}} \]  \hspace{1cm} (128)

Thus, if we subtract this time period from \( t_0 \), the time interval which is directly used for the transmission of data packets which are sampled during the previous sleep period is obtained.
\[ t_0 = \left( T_{\text{DIFS}} + T_{\text{avg}} + n \frac{N_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_t T_{\text{sleep}}}{N_{\text{data}}} \]  

(129)

And such time interval has something to do with the transmission rate, sampling rate and the sleep duration. So the abstract parameter \( t_0 \) can be described by using several simple parameters which have the practical meaning.

\[ t_0 = \left( T_{\text{DIFS}} + T_{\text{avg}} + n \frac{N_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_t T_{\text{sleep}}}{N_{\text{data}}} = \frac{R_t T_{\text{sleep}}}{R_t} \]  

(130)

In the same manner, the total number of the data packets transmitted saturately during \( T_{\text{saturated}} \) can be easily evaluated. Please note that the value of this saturated time duration is different from the one in the situation with control packet. The reason is because the data packets in the queue will be finished transmitting faster due to the loss of the RTS and CTS frame, however, the time interval between any two newly generated data packets does not change. In other words, the time for transmitting every data packet decreases yet the sampling rate does not vary. As a consequence, less time is needed for finishing transmitting those data packets in the queue.

\[ \frac{R_t T_{\text{saturated}}}{N_{\text{data}}} \]  

(131)

So we can calculate the total time which is not directly spent on the data fragment transmission during \( t_1 \), as the figure 8 shows. Since the summation of the \( t_0 \) and \( t_1 \) is \( T_{\text{saturated}} \), the time period which is directly used for the data fragment transmission during \( T_{\text{saturated}} \) can be evaluated as:

\[ t_{\text{saturated}} - t_0 = \left( T_{\text{DIFS}} + T_{\text{avg}} + n \frac{N_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_t T_{\text{saturated}}}{N_{\text{data}}} \]  

(132)

Similarly, the product of this time duration and the transmission rate \( R_t \) equals to the product of the sampling rate \( R_s \) and the saturated time period \( T_{\text{saturated}} \).

\[ t_{\text{saturated}} - t_0 = \left( T_{\text{DIFS}} + T_{\text{avg}} + n \frac{N_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_t T_{\text{saturated}}}{N_{\text{data}}} = \frac{R_t T_{\text{saturated}}}{R_t} \]  

(133)
Actually, the equation (130) and (133) can make up an equation set, from which we can successfully solve the value of the saturated time period $T_{\text{saturated}}$. Obviously, we can observe that both the value of the numerator and the denominator decreases due to the minimized transmission time for a data packet. As a result, the saturated time period is no longer the one in the situation where control packet is adopted. Since the size of a data packet equals to the summation of the size of all the divided separated data fragments $N_{\text{data}} = nN_f$, the value of $T_{\text{saturated}}$ can be written as:

$$T_{\text{saturated}} = \frac{R_s T_{\text{sleep}}}{R_t} + \left( T_{\text{DIFS}} + T_{\text{avg}} + \frac{nN_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_s T_{\text{sleep}}}{nN_f}$$

(134)

The total number of data packets transmitted during the unsaturated period of the listen phase in the situation without control packet is estimated in the same way as that in the situation with control packet.

$$n_u = \frac{R_s T_{\text{unsaturated}}}{N_{\text{data}}}$$

(135)

However, since the value of the saturated time differs from each other, the related unsaturated time changes, for that the frame duration and the sleep phase still remain the same. So if the saturated time is shorter, the unsaturated time must be larger. For unification reasons, we also prefer to consider the data packet in the form of the summation of the data fragments.

$$n_u = \frac{R_s (T_{\text{frame}} - T_{\text{sleep}} - T_{\text{saturated}})}{nN_f}$$

(136)

After the combination of the equation (124) (125) (126) and (136), the final overall average collision probability is able to be well presented by the saturated period $T_{\text{saturated}}$ and some other simple given parameters.

$$p = \frac{p_s (T_{\text{frame}} (1 - D) + T_{\text{saturated}}) + p_u (T_{\text{frame}} D - T_{\text{saturated}})}{T_{\text{frame}}}$$

(137)

If the formula (134) describing the length of the saturated period in the situation without control packet is inserted into the above equation (137), the completed overall collision probability which average the collision probability in the medium access mechanism with and without control packet will be eventually discovered.
From the previous analysis showed, the collision probability in the saturated situation \( p_s \)
is a constant value. On the contrary, the collision probability in the unsaturated situation \( p_u \) acts as different values when the time interval \( T \) belongs to different conditions. Therefore, the overall average collision probability \( p \) which is established in terms of these two collision probability will also behave differently depending on which classification \( T \) belongs to. So in conclusion, the overall average collision probability \( p \) in the situation without using control packet is able to be finally obtained by solving the following equation sets under the four different conditions which are classified according to the value of \( T \).
1. $T \in (0, \sigma)$

\[
N_{\text{neighbor}} = \left\lfloor \frac{N}{a \times b} \pi R^2 \right\rfloor - 1
\]

\[
T_{\text{avg}} = \frac{(1 - p_s) \left( \frac{C_{W_{\text{min}}} (1 - (2p_s)^{m+1})}{2(1 - 2p_s)} \right) + \left( \frac{p_s^{m+1} 2^m C_{W_{\text{min}}} (1 - p_s^{K-m-1})}{2(1 - p_s)} \right)}{1 - p_s^K}
\]

\[
p_s = 1 - \left( \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}} \times \sigma}
\]

\[M = N_{\text{neighbor}} \left(1 - \left( \frac{1}{T_{\text{avg}}} \right)^{M^{-1}} \right)
\]

\[
p_u = \frac{M}{N_{\text{neighbor}}} \left(1 - \left( \frac{1}{T_{\text{avg}}} \right)^{M^{-1}} \right)
\]

\[
T_{\text{avg}} = \frac{(1 - p_u) \left( \frac{C_{W_{\text{min}}} (1 - (2p_u)^{m+1})}{2(1 - 2p_u)} \right) + \left( \frac{p_u^{m+1} 2^m C_{W_{\text{min}}} (1 - p_u^{K-m-1})}{2(1 - p_u)} \right)}{1 - p_u^K}
\]

\[
p \cdot T_{\text{frame}} \text{ frame}(1 - D) + p_s T_{\text{frame}} \left( \frac{R_s T_{\text{sleep}}}{R_i} + \left( T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{n N_{\text{ack}}}{R_i} + (2n - 1) T_{\text{SIFS}} \right) \frac{R_i T_{\text{sleep}}}{n N_f} \right)
\]

\[p = \frac{p_u \cdot T_{\text{frame}} \cdot D - \left( \frac{R_i T_{\text{sleep}}}{R_s} + T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{n N_{\text{ack}}}{R_i} + (2n - 1) T_{\text{SIFS}} \right) T_{\text{frame}} \left( \frac{R_i T_{\text{sleep}}}{n N_f} \right)}{T_{\text{frame}}}
\]

\[p_u \left( \frac{R_i T_{\text{sleep}}}{R_s} + T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{n N_{\text{ack}}}{R_i} + (2n - 1) T_{\text{SIFS}} \right) \frac{R_i T_{\text{sleep}}}{n N_f}
\]

\[1 - \frac{R_i}{R_s} \left( T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{n N_{\text{ack}}}{R_i} + (2n - 1) T_{\text{SIFS}} \right) \frac{R_i}{n N_f}
\]

\[
(139)
\]
\[ 2. T \in [\sigma, T_{avg} \sigma) \]

\[ N_{\text{neighbor}} = \left\lfloor \frac{N}{ab\pi R^2} \right\rfloor - 1 \]

\[ T_{avg} = \frac{(1-p_s)\left(\frac{CW_{\text{min}}(1-2p_s)\pi^{m+1}}{2(1-2p_s)}\right) + \left(\frac{p_s^{m+1}2^mCW_{\text{min}}(1-p_s^{K-1})}{2(1-p_s)}\right) - \left(\frac{1-p_s^K}{2(1-p_s)}\right)}{1-p_s^K} \]

\[ p_s = 1 - \left(1 - \frac{1}{T_{avg}}\right)^{N_{\text{neighbor}}-1} \]

\[ M = \frac{T_{avg} + 2}{T_{avg} + 2 + \frac{T}{\sigma}} N_{\text{neighbor}} \]

\[ T = \frac{nN_f}{R_f} - T_{\text{DIFS}} - T_{avg}\sigma - \frac{N_{\text{ack}}}{R_s} - \frac{(n-1)N_f}{R_s} p_{fij} + \frac{(n-1)N_{\text{ack}}}{R_s} p_{ij}^s - (2n-1)T_{\text{SIFS}} \]

\[ p_u = \frac{M}{N_{\text{neighbor}}} \left(1 - \left(1 - \frac{1}{T_{avg}}\right)^{M-1}\right) \]

\[ T_{avg} = \frac{(1-p_u)\left(\frac{CW_{\text{min}}(1-2p_u)\pi^{m+1}}{2(1-2p_u)}\right) + \left(\frac{p_u^{m+1}2^mCW_{\text{min}}(1-p_u^{K-1})}{2(1-p_u)}\right) - \left(\frac{1-p_u^K}{2(1-p_u)}\right)}{1-p_u^K} \]

\[ p_s \left( T_{\text{frame}}(1-D) + \frac{R_s T_{\text{sleep}}}{R_t} + \left(T_{\text{DIFS}} + T_{avg}\sigma + \frac{nN_{\text{ack}}}{R_s} + (2n-1)T_{\text{SIFS}}\right) \frac{R_s T_{\text{sleep}}}{nN_f} \right) \]

\[ p_s \left( T_{\text{frame}}D - \frac{R_s T_{\text{sleep}}}{R_t} + \left(T_{\text{DIFS}} + T_{avg}\sigma + \frac{nN_{\text{ack}}}{R_s} + (2n-1)T_{\text{SIFS}}\right) \frac{R_s T_{\text{sleep}}}{nN_f} \right) \]

\[ p = \frac{T_{\text{frame}}}{T_{\text{frame}}} \]

\[ (140) \]
\[3. T \in [T_{\text{avg}} \sigma, T_{\text{avg}} \sigma + T_{\text{DIFS}} + T_{\text{trans}}]\]

\[
N_{\text{neighbor}} = \left\lceil \frac{N}{a \times b \times \pi R^2} \right\rceil - 1
\]

\[
T_{\text{avg}} = \left[ (1 - p_s) \left( \frac{CW_{\min} (1 - (2p_u)^{m+1})}{2(1 - 2p_s)} \right) + \left( \frac{p_u^{m+1} 2^m CW_{\min} (1 - p_s^{K-m-1})}{2(1 - p_s)} \right) - \left( \frac{1 - p_u^K}{2(1 - p_u)} \right) \right] \frac{1}{1 - p^K}
\]

\[
p_s = 1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}} - 1}
\]

\[
M = \left\lceil \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 2} N_{\text{neighbor}} \right\rceil
\]

\[
p_u = \frac{M}{N_{\text{neighbor}}} \left( 1 - \left( 1 - \frac{1}{T_{\text{avg}}} \right)^{M-1} \right)
\]

\[
T_{\text{avg}} = \left[ (1 - p_s) \left( \frac{CW_{\min} (1 - (2p_u)^{m+1})}{2(1 - 2p_s)} \right) + \left( \frac{p_u^{m+1} 2^m CW_{\min} (1 - p_s^{K-m-1})}{2(1 - p_s)} \right) - \left( \frac{1 - p_u^K}{2(1 - p_u)} \right) \right] \frac{1}{1 - p^K}
\]

\[
p_s \left( T_{\text{frame}} (1 - D) + \frac{R_s T_{\text{sleep}}}{R_t} \right) + \left( T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{nN_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_s T_{\text{sleep}}}{nN_f}
\]

\[
p_u \left( T_{\text{frame}} D - \frac{R_s T_{\text{sleep}}}{R_t} \right) + \left( T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{nN_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_s T_{\text{sleep}}}{nN_f}
\]

\[p = \frac{T_{\text{frame}}}{\left( T_{\text{frame}}(1 - D) + \frac{R_s T_{\text{sleep}}}{R_t} \right) + \left( T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{nN_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_s T_{\text{sleep}}}{nN_f}}
\]

\[p_u = \frac{T_{\text{frame}} D - \frac{R_s T_{\text{sleep}}}{R_t} \right) + \left( T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{nN_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_s T_{\text{sleep}}}{nN_f}}{\left( T_{\text{frame}}(1 - D) + \frac{R_s T_{\text{sleep}}}{R_t} \right) + \left( T_{\text{DIFS}} + T_{\text{avg}} \sigma + \frac{nN_{\text{ack}}}{R_t} + (2n - 1)T_{\text{SIFS}} \right) \frac{R_s T_{\text{sleep}}}{nN_f}}
\]

\[141\]
$$4. T \in [T_{\text{DIFS}} + \sigma T_{\text{avg}} + T_{\text{trans}}, +\infty)$$

$$N_{\text{neighbor}} = \left\lfloor \frac{N}{ab} \pi R^2 \right\rfloor - 1$$

$$T_{\text{avg}} = \frac{(1 - p_s) \left( \frac{C W_{\text{min}}(1 - (2 p_a)^m)}{2(1 - 2 p_a)} \right) + \left( \frac{p_u^m 2^m C W_{\text{min}}(1 - p_s^{K - m - 1})}{2(1 - p_s)} \right) - \left( \frac{1 - p_s^K}{2(1 - p_s)} \right)}{1 - p_s^K}$$

$$p_s = 1 - \left( \frac{1}{T_{\text{avg}}} \right)^{N_{\text{neighbor}} - 1}$$

$$M = \frac{T_{\text{avg}} + 2}{2 T_{\text{avg}} + 3 + \left( \frac{T_{\text{avg}} - T_{\text{DIFS}} - \sigma T_{\text{avg}} - T_{\text{trans}}}{\sigma} \right)^{N_{\text{neighbor}}}}$$

$$T = \frac{n N_f}{R_s} - T_{\text{DIFS}} - T_{\text{avg}} - \frac{N_{f1}}{R_t} - \frac{N_{\text{ack}}}{R_t} - \frac{(n-1)N_f}{R_s} - \frac{(n-1)N_{\text{ack}}}{R_s} - \frac{(2n-1)T_{\text{SIFS}}}{R_s}$$

$$T_{\text{trans}} = \frac{N_{f1}}{R_t} + \frac{N_{\text{ack}}}{R_t} - \frac{(n-1)N_f}{R_t} - \frac{(n-1)N_{\text{ack}}}{R_t} + \frac{(2n-1)T_{\text{SIFS}}}{R_s}$$

$$p_u = \frac{M}{N_{\text{neighbor}}} \left( 1 - \left( \frac{1}{T_{\text{avg}}} \right)^{M-1} \right)$$

$$T_{\text{avg}} = \frac{(1 - p_u) \left( \frac{C W_{\text{min}}(1 - (2 p_a)^m)}{2(1 - 2 p_a)} \right) + \left( \frac{p_u^m 2^m C W_{\text{min}}(1 - p_u^{K - m - 1})}{2(1 - p_u)} \right) - \left( \frac{1 - p_u^K}{2(1 - p_u)} \right)}{1 - p_u^K}$$

$$p_s \left( T_{\text{trans}}(1 - D) + \frac{R_s T_{\text{sleep}}}{R_t} + \frac{T_{\text{avg}}}{T_{\text{DIFS}} + T_{\text{avg}} - (2n-1)T_{\text{SIFS}}} \right) \frac{R_s T_{\text{sleep}}}{n N_f}$$

$$p = \frac{T_{\text{frame}}}{1 - \frac{R_s}{R_t} \left( T_{\text{DIFS}} + T_{\text{avg}} - (2n-1)T_{\text{SIFS}} \right) \frac{R_s}{n N_f}}$$

$$p_u \left( T_{\text{frame}} D \frac{R_s T_{\text{sleep}}}{R_t} \left( T_{\text{DIFS}} + T_{\text{avg}} - (2n-1)T_{\text{SIFS}} \right) \frac{R_s}{n N_f} \right)$$

(142)
6.2 Energy model of the first data fragment retransmission due to collision

As we discussed before, instead that the collision positioned in the RTS frame in the medium access mechanism with control packet, the collision occurs during the first data fragment which thus leads to the retransmission of only the first data fragment in the medium access mechanism without control packet. Therefore, the figure 14 must be changed to a new diagram in order to take the account of such fragment retransmission overhead rather than the RTS retransmission overhead. As the following figure 24 presents, in the access technique of the SMAC protocol which does not use control packet, between a particular sender and receiver, only the data fragments and the corresponding ACK frames are exchanged and thus collision will only occur on the first data fragment, since the time period of SIFS between the end of every data packet and the beginning of its neighboring ACK frame is smaller than the time period of DIFS, the contending nodes are unable to detect these free time spaces and thus their DIFS period can not be sensed idle. Therefore, if the first data fragment is successfully transmitted, the following packets regardless of the data fragments or the ACK frames must be sent and received successfully as well in the assumption that no error occurs during the transmission process.

The only difference of the packet transmission between the medium access scheme with and without control packet is whether the RTS and CTS frames are transmitted or not, the retransmission energy consumption of the first data fragment caused by collision will be analyzed in the similar way as that of the RTS frame. So, firstly, we calculate the energy consumption used for transmitting and retransmitting the first data fragment between the node i and its neighbor node j. And then, we evaluate the energy consumption of the transmission and retransmission of such data fragment which is only brought about by the error during the communication process. As a result, the retransmission overhead of the first data fragment caused by collision is able to be obtained by just subtracting the latter energy consumption from the former one.

6.21 Communication energy model of first data fragment between node i and j

In this section, we are going to focus on the two arbitrary nodes i and j in the wireless sensor network where i and j are neighbors one another to evaluate the energy consumption of the transmission, retransmission and reception of the first data fragment in-between, where both the collision and error may occur during the first data fragment transmission. Therefore, the retransmission energy model includes the energy consumed in the retransmission of the first data fragment between the node i and its neighbor node j due to both the collision and the error occurrence.

The notation defined in the situation with control packet is still used in the situation without
control packet. Beside, the new symbol $P_{tx,f}$ means the transmission power for successfully transmitting the first data fragment from the node $i$ which includes the retransmission power caused by both error and collision, and $E_{tx,f}$ is the corresponding energy consumed in the transmission and retransmission of such data fragment. Similarly, $P_{rx,f}$ and $E_{rx,f}$ are the reception power and the reception energy for such data fragment respectively. $\overline{P}$ and $\overline{E}$ are the average value of the power consumption $P$ and the corresponding average energy consumption $E$.

1. Energy model of transmission and retransmission of first data fragment from node $i$

Here, $P_{ij}$ is only the transmission power needed for transmitting data fragments rather than control packet, so the transmission power used for the first data fragment is able to be obtained. Since both collision and errors may happen during the transmission of the first data fragment between the node $i$ and its neighbor node $j$, the phenomenon of retransmissions may appear. On average, the probability that the first data fragment can be successfully received by the node $j$ is $p_{ij}^* p_a^*$, for that only if the following ACK frame is successfully received, does the previous transmission of the data fragment make sense. Otherwise, such data fragment must be retransmitted continuously. Thus the average power needed for transmitting and retransmitting the first data fragment between the node $i$ and $j$ is:

$$\overline{P_{tx,f}}(i,j) = P_{ij} \frac{p_{ij}^* p_a^*}{p_{ij}^* p_a^*}$$

By multiplying such power consumption $\overline{P_{tx,f}}(i,j)$ by the time needed for transmitting the first data fragment $\frac{N_f}{R_f}$, the corresponding average energy consumed in both transmitting and retransmitting the first data fragment between the node $i$ and its neighbor node $j$ in the access mechanism without control packet is able to be obtained:

$$\overline{E_{tx,f}}(i,j) = P_{ij} \frac{N_f}{R_f} \frac{p_{ij}^* p_a^*}{p_{ij}^* p_a^*}$$

2. Energy model of reception of first data fragment by node $j$

The reception power depletion is independent of both the collision and error which occur during the transmission of the first data fragment between the node $i$ and $j$, since whether the first data fragment is successfully received or not only depends on whether the following ACK frame is successfully transmitted. Thus, the reception power consumption of the first data fragment is only
related with the error probability during the transmission of the following ACK frame. In addition, the failure of the transmission of all the data fragments and ACK frames afterwards will never lead to the retransmission of the first data fragment, so only the error probability of the first ACK frame is taken into account when estimating the average power needed for successfully receiving the first data fragment by the node j.

\[
\bar{P}_{rx, f}(i, j) = \frac{P_{rx}}{p_{aji}}
\]

(145)

Thus the corresponding average energy consumed in successfully receiving the first data fragment by the node j can be worked out just by the multiplication of the time needed for the transmission of such data fragment.

\[
E_{rx, f}(i, j) = \frac{P_{rx} N_f}{R_t} \frac{1}{p_{aji}}
\]

(146)

3. Communication energy model of first data fragment between node i and j

After summing up the transmission and the retransmission energy that the node i consumes in sending the first data fragment to the node j with the reception energy that the node j needs for successfully receiving the first data fragment, the total communication energy which is spent on the first data fragment between the node i and j is established. Please note that this communication energy includes the retransmission energy of the first data fragment caused by both error and collision.

\[
E_f(i, j) = E_{tx, f}(i, j) + E_{rx, f}(i, j)
\]

(147)

By inserting the formula (144) presenting the transmission and the retransmission energy consumed in transmitting the first data fragment by the node i and the formula (146) presenting the reception energy consumed in successfully receiving the first data fragment by the node j into the formula (147), the total energy consumed in the successful communication of the first data fragment between the node i and its neighbor node j can be written as:

\[
E_f(i, j) = \frac{P_{yi} N_f}{R_t} \frac{1}{p_{aji}} + \frac{P_{rx} N_f}{R_t} \frac{1}{p_{aji}}
\]

(148)

6.22 Communication energy model of first collision-free data fragment between node i and j

In this subsection, we are going to calculate the communication energy of the first data fragment between the node i and j where the difference comparing with the last subsection is that we assume the collision will never occur during the transmission of the first data fragment. As a result, such communication energy only includes the transmission energy of the first data fragment, the retransmission energy of the first data fragment if error occurs during the previous transmission attempt and the reception energy of such data fragment between the node i and the
node j. By doing so, the total energy consumed in the communication of the first data fragment which is transmitted without the collision occurrence is able to be evaluated finally.

Besides the notations discussed before, \( P'_{tx,f}(i,j) \) indicates the power which ensures the successful transmission of the first data fragment between the node i and j, during which only error may occur. And \( E'_{tx,f}(i,j) \) represents the corresponding energy consumption of such first data fragment. Likewise, \( P_{rx,f}(i,j) \) and \( E_{rx,f}(i,j) \) is the reception power and the corresponding reception energy that guarantees the successfully reception of the first data fragment by the node j from the node i. Please note this reception notation remains the same as it is used in the situation where collision may also occur during the data fragment transmission. Since in both situations, no matter whether the retransmission of the first data fragment is caused by both errors and collision or only by errors, the reception power and energy have nothing to do with the collision probability, instead, they only have relation with the error probability of the next ACK frame. Therefore, instead of the notation \( P'_{rx,f}(i,j) \) and \( E'_{rx,f}(i,j) \), the previous notation \( P_{rx,f}(i,j) \) and \( E_{rx,f}(i,j) \) is still used when considering the reception power and the reception energy of the first data fragment which is collision free between the node i and its neighbor node j.

1. Energy model of transmission and retransmission of first collision-free data fragment from node i

As already mentioned, in this part, we only consider the power consumed in successfully transmitting the first data fragment from the node I where collision will never occur. So it only has something to do with the error rate which happens in the transmission of both the first data fragment and the first ACK frame between the node i and the node j. thus, the average power model of the transmission and the retransmission of the first collision-free data fragment from the node i is:

\[
P'_{tx,f}(i,j) = \frac{P_y}{P_y P_{aji}} \tag{149}
\]

By multiplying the time used for transmission of the first data fragment between the node i and the node j, the average energy model of the transmission and the retransmission of the first collision-free data fragment from the node i is:

\[
E'_{tx,f}(i,j) = \frac{N_f}{R_i} \frac{P_{aji}}{P_y} \tag{150}
\]

2. Energy model of reception of first collision-free data fragment by node j
Similarly, the average reception power needed for successfully receiving the first data fragment by the node j is able to be calculated. And such reception power is the same as it is considered in the situation where collision may also occur during the transmission of the data fragment.

$$P_{rx,f}(i,j) = \frac{P_{rx}}{P_{aji}}$$  \hspace{1cm} (151)

The corresponding average reception energy consumed in successfully receiving the first data fragment which is collision-free by the node j is:

$$E_{rx,f}(i,j) = \frac{P_{rx} N_f}{R_{s}}$$ \hspace{1cm} (152)

3. Communication energy model of first collision-free data fragment between node i and j

The total average communication energy of the first data fragment which is collision-free between the node i and its neighbor node j is obtained by adding together the average energy of the transmission and retransmission of the first collision-free data fragment with the average energy of successful reception of such data fragment between the node i and the node j. And this average communication energy model excludes the energy used for the retransmission of the first data fragment due to the failure caused by collision occurrence.

$$E_f(i,j) = E_{tx,f}(i,j) + E_{rx,f}(i,j)$$  \hspace{1cm} (153)

By inserting the previous formula (150) of the average transmission energy consumed in successfully transmitting the first data fragment where collision will never occur and the formula (152) of the average reception energy of the first data fragment, the formula of the total average energy consumed in the communication of the first data fragment which is collision-free between the node i and its neighbor node j can be written as:

$$E_f(i,j) = \frac{P_{tx} N_f}{R_{s}} + \frac{P_{rx} N_f}{P_{aji} P_{aji}}$$  \hspace{1cm} (154)

6.23 Retransmission energy model of first data fragment due to collision.

In the previous subsection 6.21, the communication energy model of the first data fragment between the node i and the node j is built, which includes the transmission energy of the first data fragment, the retransmission energy of the first data fragment caused by both the collision and error appearance and the reception energy of the first data fragment. In the subsection 6.22, however, besides the transmission and the reception energy of the first data fragment, the retransmission energy of the first data fragment only due to the error appearance is also included in the communication energy model between the node i and its neighbor node j.

83
Obviously, the energy consumed in the transmission and the reception of the first data fragment is the same in both of the situations. The only difference between these two communication energy models lies in the retransmission energy of the first data fragment. In the former case, the retransmission energy is caused by both the error and collision during the transmission of the first data fragment. In the latter case, however, the retransmission energy produced is only due to collision. As a result, by subtracting the latter communication energy from the former one, we are able to obtain the retransmission energy of the first data fragment due to only collision between the node i and its neighbor node j. If such retransmission energy is counted in every hop along a path, the average total retransmission energy of the first data fragment due to collision consumed along the path from the source node to the sink is able to be evaluated. And this is the quantification of one of the arising consequences if control packet is not applied during the packet transmission.

1. Retransmission energy model of first data fragment due to collision between node i and j

The formula of calculating the retransmission energy of the first data fragment between the node i and the node j which is caused only by the collision occurred during the transmission of the first data fragment is:

\[
E_{col}^f (i, j) = E_f(i, j) - E_f(i, j) \tag{155}
\]

By inserting the formula (148) and (154) into the above equation, the formula of the retransmission energy of the first data fragment due to the collision between the node i and the node j can be written as:

\[
E_{col}^f (i, j) = \frac{P_{ij}}{R_t} N_f + \frac{P_{rx}}{R_t} N_f \left( \frac{P_{ij} N_f}{R_t} + \frac{P_{rx}}{R_t} \right) \tag{156}
\]

After simplification, the above formula of the average retransmission energy of the first data fragment happened because of the collision can be transferred as:

\[
E_{col}^f (i, j) = \frac{P_{ij} N_f}{R_t} - \frac{P_{ij} N_f}{R_t} \tag{157}
\]

After changing the format of such formula, the final average retransmission energy of the first data fragment due to collision between the node i and its neighbor node j is written as:

\[
E_{col}^f (i, j) = \frac{P_{ij} N_f}{R_t} - \frac{p_{ij} P_{ajij} N_f}{R_t} \tag{158}
\]

2. Retransmission energy model of first data fragment due to collision between source and
We assume there is \( M+1 \) number of nodes in all in a particular one dimensional routing in the wireless sensor network. Then after adding together the retransmission energy of the first data fragment due to collision in every hop along the path, the total average retransmission energy of the first data fragment due to collision consumed in the entire routing is able to be quantified.

\[
E^{col}_f = \sum_{i=0}^{M-1} E^{col}_f (i, i+1) \quad (159)
\]

In order to explain more clearly, the relationship between the average retransmission energy \( E^{col}_f (i, j) \) of first data fragment due to collision consumed between two one-hop sensor nodes within a one dimensional path and the total average retransmission energy \( E^{col}_f \) of the first data fragment due to collision consumed along the path from the source node (node 0) to the sink (node M) is described in the following figure 25.

![Fig.25. The difference between \( E^{col}_f (i, j) \) and \( E^{col}_f \)](image)

6.3 Energy model of the extra overhearing of the first data fragment and ACK frame

In the situation without the control packet RTS and CTS frame, only data fragments and the corresponding ACK frames are involved in the communication among sensor nodes. Since a NAV vector is inserted into every data fragment and ACK frame, the first data fragment and ACK frame could actually also play the role of the RTS and CTS packet. As a result, the first data fragment and the first ACK frame have the capability of contending for the medium by employing both virtual carrier sense and physical carrier sense. If the first data fragment is transmitted successfully by the transmitter, all the neighbor nodes of the transmitter except for the receiver will turn off their radio and go to sleep until the NAV value decrements to zero in the assumption of no error and collision occurred during the transmission of the first data fragment. In the same way, after the successful transmitting of the first ACK frame by the receiver, all the neighbor nodes of the receiver except for those nodes which are also the neighbors of the transmitter will go to sleep until the medium is determined free when both the virtual and physical carrier sense indicate so if the first ACK frame is also successfully received by all the neighbors of the receiver. Therefore, comparing with the situation with control packet where the overhearing occurs on the RTS and
CTS frame, the overhearing only occurs during the transmission of the first data fragment and the first ACK frame in the situation without control packet. Thus, in this subsection, we will build up the corresponding energy model of the overhearing of the first data fragment and ACK frame by the neighbors of the transmitter and the receiver.

### 6.3.1 Overhearing energy of the first data fragment

In the last paragraph, due to the very low error probability occurred during the transmission of the data packet, we only talked about the overhearing duration in the normal situation where we assume both the first data fragment and the first ACK frame are successfully received. What if the first ACK fragment suffers from errors during its transmission after the receiver successfully received the first data fragment from the transmitter as the following figure 26 describes?

![Fig.26. Bugs in the overhearing avoidance mechanism of SMAC](image)

Obviously, the transmitter will retransmit the first data fragment due to the lack of the feedback ACK frame from the receiver side even though the first data fragment has been successfully received by the receiver. Since the first data fragment is the first part of the whole transmission, it has to be retransmitted after the medium is occupied again by the same particular transmitter rather than being retransmitted directly after its first transmission failure. However, due to the fairness of the medium competition scheme, the probability is very low that the same node can still win the medium at the second reservation attempt. As a result, another neighbor node of the receiver, say, D, might own the channel and will communicate with this receiver. In this very special case, problem arises. Since there is a NAV vector inside each packet being transmitted, the neighbor nodes of the transmitter (node A, B, C in the figure 26) will go to sleep until the NAV value decreases to zero. Therefore, all the neighbor nodes will still keep in the sleep mode even though in fact this round of the transmission has already been terminated long time ago. So suppose one of the neighbors of the transmitter is also a neighbor of the receiver (node C), it should be free to contend for the medium whenever the medium is free. However in this specific
situation, it is still in the sleep mode and thus will never be aware of the idle channel status. As a result, it loses the rights and the opportunity to contend for the medium.

Or suppose one of the neighbor nodes of the receiver, for example D, succeed in occupying the medium in the second competition. The overhearing scheme by applying the NAV vector however will not take effort from the perspective of the overlap neighbor node C since C has already been in the sleep mode. So if many neighbor nodes of the receiver are also the neighbor nodes of the transmitter, it seems the overhearing is totally an unnecessity.

Moreover, we could learn that the error probability of the first ACK frame does not affect the overhearing duration at all due to the imperfect NAV overhearing mechanism. In order words, no matter whether the first ACK frame is successfully received or not by the transmitter, all the neighbor nodes of the transmitter will be in sleep mode for at least an-entire-transmission time period. Maybe it’s a small bug in SMAC that the author has not considered yet, but this small bug will definitely make the energy model setting up of the extra overhearing in the situation without control packet extremely complicated. Therefore, in order to simplify the modeling work, we assume the error will never occur during the transmission of the first ACK frame, since in reality, the error probability mainly depends on the reliability of the wireless network and usually it is very low, as low as 0.001.

As a result, the average power consumption of the extra overhearing of the first data fragment by the neighbor nodes of the transmitter i should be evaluated without considering the error probability during the transmission of the following first ACK frame.

\[ P_{overhear}^{f_1}(i, j) = P_{rx}N_{neighbor} \]  (160)

After taken into the account of the time needed for transmitting the first data fragment \( \frac{N_{f_1}}{R_t} \), the average energy consumption of the extra overhearing by the neighbor nodes of the transmitter i in the situation without control packet is:

\[ E_{overhear}^{f_1}(i, j) = P_{rx} \frac{N_{f_1}}{R_t}N_{neighbor} \]  (161)

6.32 Overhearing energy of the first ACK frame

Concerning to the first ACK frame, since it’s for sure will be successfully received by all the neighbors of the receiver, the power consumption of the overhearing of the first ACK frame only has something to do with the reception power level and the number of the valid neighbor nodes of the receiver where valid means those neighbors which are only the neighbor nodes of the receiver.

Since the SMAC protocol adopts the message passing mechanism, except for the first data fragment and the first ACK frame in the situation without control packet, all the packets should be
retransmitted immediately after their first failed transmission attempts. Thus, the retransmission of the following data packets will not cause the retransmission from the beginning like the 802.11 protocol, that is, will not cause the retransmission of the first data fragment and the first ACK frame. Therefore, whether or not the following data packets transmitted successfully will not influence the power consumption of the first ACK frame overhearing by the valid neighbors of the receiver.

So after estimating the valid neighbor nodes of the receiver \( j \), which equals to the subtraction of the overlap neighbor nodes of both the transmitter and the receiver from the whole neighbor nodes of the receiver, we are able to evaluate the power consumption of the extra overhearing of the first ACK frame by the valid neighbors of the receiver.

\[
P^{\text{overhear}}_{\text{ack1}}(i, j) = P_{rs}(N_{\text{neighbor}} - N_{\text{overlap}})
\]  

(162)

By multiplying by the time needed for transmitting the first ACK frame, the total energy consumption of the extra overhearing of the first ACK frame by the valid neighbor nodes of the receiver in the situation without control packet is able to be calculated.

\[
E^{\text{overhear}}_{\text{ack1}}(i, j) = P_{rs} \frac{N_{\text{ack1}}}{R_i}(N_{\text{neighbor}} - N_{\text{overlap}})
\]  

(163)

6.33 Overhearing energy of the first data fragment and ACK frame

As usual, we firstly evaluate the energy consumption in a hop between two randomly chosen immediate neighbor nodes \( i \) and \( j \), and then extend the way of the energy consumption estimation along a path in the wireless sensor network. Here, we also assume the simplest routing structure, one dimensional linear routing, which is well enough for the comparison of the extra energy consumption between in the situation with control packet and in the situation without control packet.

1 Between node \( i \) and \( j \)

By simply adding together the above formulas which have evaluated the extra energy consumption of the overhearing of the first data fragment and the first ACK frame between the node \( i \) and its neighbor node \( j \) along the path respectively, we could delightedly obtain the total average extra energy consumption of the overhearing of the first data fragment and the first ACK frame by the neighbor nodes of the transmitter and the receiver between the two one-hop neighbor nodes \( i \) and \( j \).

\[
E^{\text{overhear}}_{\text{total}}(i, j) = E^{\text{overhear}}_{\text{f}}(i, j) + E^{\text{overhear}}_{\text{ack1}}(i, j)
\]  

(164)

If we just insert the above formula (161) and (163) into this newly created formula, the concrete presentation of the total average extra energy consumption of the overhearing by the neighbor nodes of both the transmitter and the receiver between a random node \( i \) and its neighbor node \( j \) in the situation where control packet is deactivated is able to be found out. As we could observe, such extra energy consumption has nothing to do with the error probability and the
collision probability during the transmission of any node.

\[
E_{\text{overhear}}(i, j) = P_{\text{rx}} \frac{N_{f1}}{R_j} N_{\text{neighbor}} + P_{\text{rx}} \frac{N_{\text{ack1}}}{R_j} (N_{\text{neighbor}} - N_{\text{overlap}}) \quad (165)
\]

2 Between a source and the sink

There are totally M sensor nodes in the one dimensional routing of the wireless sensor network, so if we add such extra energy consumption of the overhearing of the first data fragment and the first ACK frame in each hop along the path from the source (node 0) to the sink (node M), the total average extra energy consumption of the overhearing of the first data fragment and the first ACK frame of the whole routing in the situation where the control packet RTS and CTS are eliminated will be eventually evaluated.

\[
E_{\text{overhear}} = \sum_{i=0}^{M-1} E_{\text{overhear}}(i, i+1) \quad (166)
\]

The relationship between the extra overhearing energy which is consumed between the node i and its immediate neighbor node j and the extra overhearing energy spent on the entire routing from the source to the sink which includes the sensor nodes i and j is clearly and visually described in the figure 16 which is presented in the previous section 5.5 when we were discussing the total extra overhearing energy consumption in the situation where the control packet RTS and CTS are activated.

6.4 Energy model of the extra idle listening

In the 802.11 protocol, all the sensor nodes never go to sleep mode, in order words, they keep listening all the time. Thus, even if the basic access mechanism consumes less time in occupying the medium than the four way hand-shaking mechanism in the unsaturated situation, the saying of the extra listening time does not make sense. Differently, in the SMAC protocol, since the overhearing avoidance is supported, all the neighbor nodes of the transmitter and the receiver will go to sleep mode after they hear the packets which are not directed to them. In the situation with control packet, all the neighbor nodes of the transmitter and the receiver will go to sleep after they receive the RTS and CTS frame respectively. And in the situation without control packet, all the neighbor nodes of the transmitter and the receiver will go to sleep after they hear the first data fragment and the first ACK frame respectively.

However, in the saturated period of the listen phase, the time interval T between the previous last ACK frame and the current DIFS sensing period is zero, which indicates no matter of the transmitter, the receiver or their neighbors are impossible to go to sleep after a round of a transmission. In order words, the phenomenon of the idle listening is totally avoided during the saturated period. On the contrary, during the unsaturated period of the listen phase, there are no sampled data packets accumulated in the queue any more, so after a data packet finished transmitting by using the transmission rate, the newly sampled data packet hasn’t come yet due to the much lower sampling rate. As a result, the time interval T between the previous last ACK frame and the current DIFS sensing period is nonzero. And the length of such time interval T in
the situation with control packet is different from that in the situation without control packet, as the following figure 27 shows. From the figure, we could observe that the idle listening time $T_2$ in the situation without control packet is longer than the idle listening time $T_1$ in the situation with control packet. The reason is because for both cases, the sampling rate is the same while the total size of the packets being transmitted is different. In the situation with control packet, the extra RTS and CTS frame are required to be sent in order to reserve the medium for the whole transmission comparing with the situation without control packet.

![Diagram](source)

**Fig.27. Extra idle listening in the situation without control packet**

As a result, the extra idle listening time in the situation without control packet is $(T_2 - T_1)$ in the assumption that the retransmission circumstances are exactly the same in both situations, where $T_2$ is the idle listening time in the situation with control packet (a) and $T_1$ is the idle listening time in the situation without control packet (b). The result of this subtraction comes from the extra RTS frame, CTS frame and two SIFS time intervals, which is able to be formulating after considering the transmission rate as:

$$T_2 - T_1 = \frac{N_{\text{rts}} + N_{\text{cts}}}{R_t} + 2T_{\text{SIFS}}$$

(167)

If we use this extra idle listening time to multiply by the idle power consumption $P_{\text{idle}}$, the average energy consumed in the extra idle listening for one of the neighbor nodes is able to be evaluated.

$$\overline{E_{\text{idle\_listening\_one\_node}}} = P_{\text{idle}} \left( \frac{N_{\text{rts}} + N_{\text{cts}}}{R_t} + 2T_{\text{SIFS}} \right)$$

(168)

During the unsaturated period of the listen phase, only the transmitter and the receiver will never go to sleep mode since they are either in the transmission mode, reception mode or listening
mode. Due to the relationship between the sampling rate and the total packet size, the transmitter
and the receiver listen for the longer time in the situation without control packet. In addition, both
the neighbor nodes of the transmitter and receiver turn on their radio after the completion of a
round of the data packet transmission and thus change to the idle listening status. Therefore, not
only the transmitter $i$ and the receiver $j$, but also the neighbor nodes of them are involved in the
extra idle listening. After taken into account of those neighbor nodes $N_{\text{overlap}}$ which are the
neighbors of both the transmitter and the receiver, the total number of nodes which affects the
energy consumed in the extra idle listening is:

$$2N_{\text{neighbor}} - N_{\text{overlap}} + 2$$  \hspace{1cm} (169)

By combining the above formula (168) and (169), the total average energy spent on the idle
listening in the unsaturated situation between the node $i$ and its neighbor node $j$ can be formulated
as:

$$E_{\text{idle, listening}}^{\text{unsaturated}}(i, j) = P_{\text{idle}} \left( \frac{N_{\text{cts}} + N_{\text{ct}}}{R_t} + 2T_{\text{SIFS}} \right) (2N_{\text{neighbor}} - N_{\text{overlap}} + 2) \hspace{1cm} (170)$$

When we evaluate any extra energy consumption no matter in the situation with or without
control packet, we firstly estimate such extra energy consumed between a node $i$ and its immediate
neighbor node $j$ from the perspective of the whole frame period. That is, we do not consider such
extra energy consumption in the saturated situation and in the unsaturated situation separately,
since usually the phenomenon which causes the extra energy consumption occurs in both the
saturated and unsaturated situation. Out of the ordinary, the extra idle listening only exists in the
unsaturated period of the SMAC protocol, thus in order to facilitate the comparison of the extra
energy consumption in the situation with control packet with that in the situation without control
packet, we will give the weighted average energy consumption of the extra idle listening here by
using the total number of the sampled data packets transmitted during the saturated and
unsaturated period. The reason why we average by using the metric of the number of data packets
is because an extra idle listening time $(T_2 - T_1)$ arises for each transmission round in the
unsaturated period and the number of the transmission rounds depends on the number of the
transmitted data packets in the duration.

As we discussed before, the total number of the data packets transmitted during the
saturated period and the unsaturated period is $n_s$ and $n_u$, respectively. Since the extra idle listening
only occur during the unsaturated period of the listen phase, the total number of the data packets
transmitted during the unsaturated period $n_u$ should be set as the numerator, and the summation
of the data packets transmitted during the saturated $n_s$ and unsaturated period $n_u$ should be of
course acted as the denominator. This coefficient is therefore a number which is smaller than one
but larger than zero.

\[ a = \frac{n_u}{n_s + n_u} \]  \hspace{1cm} (171)

Recall the series of the formulas we built up in the section 6.13 when we calculated the overall average collision probability, the number of the data packets transmitted during the saturated period is:

\[ n_s = \frac{R_s T_{sleep}}{N_{data}} + \frac{R_s T_{saturated}}{N_{data}} \]  \hspace{1cm} (172)

And the formula presenting the total number of the data packets which are transmitted during the unsaturated period of the listen phase in the SMAC protocol is

\[ n_u = \frac{R_s (T_{frame} - T_{sleep} - T_{saturated})}{N_{data}} \]  \hspace{1cm} (173)

Therefore, if we insert the above formulas (172) and (173) into (171), the coefficient \( a \) which used to describe the weighted average energy consumption of the extra idle listening in the situation where control packet is deactivated is able to be found out.

\[ a = \frac{R_s (T_{frame} - T_{sleep} - T_{saturated})}{N_{data}} - \frac{R_s (T_{frame} - T_{sleep} - T_{saturated})}{N_{data}} + \frac{R_s T_{sleep}}{N_{data}} + \frac{R_s T_{saturated}}{N_{data}} \]  \hspace{1cm} (174)

After simplification, the above formula can be transferred to:

\[ a = \frac{T_{frame}}{T_{frame}} \]  \hspace{1cm} (175)

Since the sleep period is determined by the duty cycle, it can be further described with the help of the frame phase \( T_{frame} \) and the duty cycle \( D \)

\[ T_{sleep} = T_{frame} (1 - D) \]  \hspace{1cm} (176)

If we combine this formula with the above formula (175), the simplest formula which presents the specific coefficient will be solved.

\[ a = \frac{T_{frame} D - T_{saturated}}{T_{frame}} \]  \hspace{1cm} (177)
Since the saturated period has already been successfully obtained in the subsection 6.13 when we considering the overall average collision probability in the situation without control packet, if we insert that formula quantifying the saturated time (134) into the above formula (177), the final coefficient will be eventually discovered.

\[
T_{frame} = D - \frac{R_s T_{sleep}}{R_i} - \left( T_{DIFS} + T_{avg} + \frac{nN_{ack}}{R_i} + (2n - 1)T_{SIFS} \right) \frac{R_s T_{sleep}}{nN_f} \\
a = \frac{1 - \frac{R_s}{R_i} - \left( T_{DIFS} + T_{avg} + \frac{nN_{ack}}{R_i} + (2n - 1)T_{SIFS} \right) \frac{R_s}{nN_f}}{T_{frame}}
\]

(178)

So when we add this coefficient before the item presented in the formula (170) which gives the energy consumed in the extra idle listening between the node i and its neighbor j in the unsaturated period where control packet is deactivated, we will get the energy consumption of the extra idle listening between the node i and j in the whole frame cycle, which makes the comparison of the extra energy consumption between the situation with and without control packet very convenient.

\[
E_{total, idle \_listening \,(i,j)} = \left( T_{frame} - D - \frac{R_s T_{sleep}}{R_i} - \left( T_{DIFS} + T_{avg} + \frac{nN_{ack}}{R_i} + (2n - 1)T_{SIFS} \right) \frac{R_s T_{sleep}}{nN_f} \right) \\
\times \frac{1 - \frac{R_s}{R_i} - \left( T_{DIFS} + T_{avg} + \frac{nN_{ack}}{R_i} + (2n - 1)T_{SIFS} \right) \frac{R_s}{nN_f}}{T_{frame}}
\]

(179)

Since the energy of the extra idle listening in the situation without control packet has already been found between the node i and its neighbor node j, that is, such energy consumption has already been successfully evaluated in one hop of a routing. By counting together this energy consumption of the extra idle listening in every hop along a path from the source (node 0) to the sink (node M), the total energy consumed in the extra idle listening in the situation without control packet is able to be quantified, so that the trade off whether to use control packet or not in the SMAC protocol which depends on the total energy consumption in both cases will be easily made.

\[
E_{total, idle \_listening} = \sum_{i=0}^{M-1} E_{total, idle \_listening \,(i,i+1)}
\]

(180)

In order to make the understanding of the relationship between the energy consumption of
the extra idle listening in one hop \( E^{\text{idle\_listening}}_{\text{total}}(i, j) \) and the energy consumption of the extra idle listening for an entire routing \( E^{\text{idle\_listening}}_{\text{total}} \) more clearly, we prefer to use the following figure 28 to illustrate. From it, we will easily figure out that the former energy overhead is only a small part of the latter one and the percentage the former one holds in the entire energy consumption of the extra idle listening depends on the number of the sensor nodes within a one dimensional routing.

**Fig. 28.** The relationship between \( E^{\text{idle\_listening}}_{\text{total}}(i, j) \) and \( E^{\text{idle\_listening}}_{\text{total}} \)

### 6.5 Energy model of the extra radio mode switching

In the section 5.7 when we built up the energy model of the extra radio mode switching in the situation where control packet is applied, we discussed that the energy consumption of the radio mode switching has nothing to do with the size of the transmission and the number of the packets transmitted within a round of the transmission. Instead, such energy consumption is only related with the total number of the data packets transmitted during a certain period of time. In other words, the energy consumption of the radio mode switching by the neighbor nodes of the transmitter and the receiver only has something to do with the number of transmission within a time interval.

In the situation where control packet is activated, the time period \( T_1 \) of the saturated phase is larger than that in the situation where control packet is deactivated due to the larger size of a transmission and the non-stop sampling work. Thus, in order to send out all the data packets accumulated in the queue which are sampled during the whole saturated period, the number of the data packets transmitted during the time interval \( T_1 \) in the situation with control packet is more than that in the situation without control packet.

Since the time period \( T_1 \) in the situation with control packet is larger than the time period \( T_1 \) in the situation without control packet, the saturated period in the situation with control packet will be larger than that in the situation without control packet simply because the saturated period
is the summation of the time phases $T_0$ and $T_1$, where $T_0$ remains the same no matter control packet is used or not.

$$T_{\text{saturated}} = T_0 + T_1$$

(181)

However, the duty cycle and the frame length is the same for both the situations with and without control packet. Since the frame period equals to the summation of the saturated and the unsaturated phase, the unsaturated interval in the situation with control packet will be shorter than that in the situation without control packet.

$$T_{\text{frame}} = T_{\text{saturated}} + T_{\text{unsaturated}}$$

(182)

From the time on of the unsaturated period, there will not be any data packet accumulated in the queue, so the number of the data packets transmitted during the unsaturated period in both situations with and without control packet only depends on the size of the unsaturated time period. As a consequence, the number of the data packets transmitted during the unsaturated period in the situation without control packet will be more than that in the situation with control packet. And this is the exact source of the extra energy consumption of the radio switching between the sleep mode and the reception mode by the neighbors of the transmitter and the receiver in the situation where control packet is deactivated.

In the same manner, we will firstly calculate the energy consumption of the radio switching from the reception mode to the sleep mode by the neighbor nodes of the transmitter. As soon as the neighbor nodes of the transmitter overhear the first data fragment which is not destined to them, they will go to sleep immediately by changing their radio status from the reception level to the sleep level. And this sub energy model can also be built up with the help the figures 19 (b) and 21 which are showed in the previous section 5.7.

$$E_1 = \frac{1}{2} T_{\text{down}} (P_{\text{reception}} - P_{\text{sleep}}) N_{\text{neighbor}}$$

(183)

Similarly, we could evaluate the extra energy consumption of the radio mode switching from the reception status to the sleep status by the left neighbor nodes of the receiver, which equals to the area of the left yellow triangle in the figure 21 times the total number of the left neighbor nodes of the receiver rather than all the neighbor nodes of the receiver, since those neighbor nodes which are both the neighbors of the transmitter and the receiver have already changed from the reception mode to the sleep mode after they overhear the first data fragment. As a result, the extra energy consumption of the radio mode switching by the valid neighbor nodes of the receiver can be estimated as:

$$E_2 = \frac{1}{2} T_{\text{down}} (P_{\text{reception}} - P_{\text{sleep}}) (N_{\text{neighbor}} - N_{\text{overlap}})$$

(184)

As soon as the NAV vector inside the first data fragment and the first ACK frame which are received by the neighbors of the transmitter and the receiver respectively decreases to zero, both
the neighbor nodes of the transmitter and the receiver will wake up by changing their radio statues from the sleep mode back to the reception mode. Since switching the radio from one status to another status does not occur instantaneously, $T_{up}$ means the time needed to handle such radio mode switching. Besides, the total valid neighbor nodes of the transmitter and the receiver equals to the summation of the neighbor nodes of the transmitter and the receiver respectively, however, excludes those intersection neighbor nodes which are the neighbors of both the transmitter and the receiver.

$$N_{total_{neighbor}} = 2N_{neighbor} - N_{overlap} \quad (185)$$

As the result, the energy consumption of the radio switching from the sleep mode to the reception mode by all the neighbor nodes of the transmitter and the receiver can be evaluated after the final ACK frame is successfully received by the transmitter.

$$E_3 = \frac{1}{2} T_{up} (P_{reception} - P_{sleep})(2N_{neighbor} - N_{overlap}) \quad (186)$$

Therefore, just by adding together the energy consumption $E_1$, $E_2$ and $E_3$, we will obtain the total energy consumption of the radio mode switching by the neighbors of the transmitter and the receiver for a round of the transmission, that is the transmission between the transmitter $i$ and its immediate receiver $j$.

$$E = E_1 + E_2 + E_3 \quad (187)$$

The formula $E$ can be finally solved out if we combine the previous formulas (183) and (184) and (186). After the transformation, the formula $E$ can be written as:

$$E = \frac{1}{2} (P_{reception} - P_{sleep})(2T_{down}N_{neighbor} - T_{down}N_{overlap} + 2T_{up}N_{neighbor} - T_{up}N_{overlap}) \quad (188)$$

Comparing with the situation with control packet, the extra number of the data packets being transmitted in the situation without control packet only appears during the unsaturated period of the listen phase, so we should average the energy consumption $E$ by adding a coefficient in front of the item. Since the total energy consumption of the radio mode switching is only related with the number of the data packets being transmitted, such coefficient should be developed in terms of the total number of the data packets transmitted during the unsaturated period $T_{unsaturated}$ in both situations with and without control packet and the total number of the data packets transmitted during the whole frame.

$$a' = \frac{R_sT_{frame} - R_{i}T_{unsaturated}}{N_{data}} \quad \frac{R_{i}T_{unsaturated}}{N_{data}} \quad (189)$$
The numerator of the fraction represents the extra data packets transmitted during the unsaturated period $T_{\text{unsaturated}}$ in the situation without control packet comparing with the data packets transmitted during $T_{\text{unsaturated}}$ in the situation with control packet. And the denominator describes the total data packets transmitted during the whole frame which is the same for both the situations with and without control packet. After simplification, the above formula can be transferred to:

$$a' = \frac{T_{\text{unsaturated}}^{\text{without}} - T_{\text{unsaturated}}^{\text{with}}}{T_{\text{frame}}}$$  \hspace{1cm} (190)

So if we add this coefficient $a'$ in front of the item in the formula (188). The weighted average extra energy consumption of the radio switching between the sleep mode and the reception mode of the neighbors of the transmitter and the receiver in the situation without control packet will be successfully calculated. Such weighted average energy consumption is only analyzed in a hop between the transmitter $i$ and its immediate receiver $j$ along a routing.

$$E_{\text{total}}^{\text{radio switching}}(i, j) = a' \frac{1}{2} (P_{\text{reception}} - P_{\text{sleep}}) \times (2T_{\text{down}} N_{\text{neighbor down}} - T_{\text{down}} N_{\text{overlap}} + 2T_{\text{up}} N_{\text{neighbor up}} - T_{\text{up}} N_{\text{overlap}})$$  \hspace{1cm} (191)

Assume there are together $M+1$ sensor nodes in an one dimensional routing in the wireless sensor network, if we add this extra energy consumption of the radio mode switching in every hop along the path from the source node (node 0) to the sink (node $M$), the total extra energy consumption of the radio mode switching by the neighbors of the transmitter and the receiver in the situation without control packet will be eventually evaluated.

$$E_{\text{total}}^{\text{radio switching}} = \sum_{i=0}^{M-1} E_{\text{total}}^{\text{radio switching}}(i,i+1)$$  \hspace{1cm} (192)

In order to make the understanding of the relationship between the energy consumption of the extra radio mode switching in one hop $E_{\text{total}}^{\text{radio switching}}(i,j)$ and the energy consumption of the extra radio mode switching for an entire routing $E_{\text{total}}^{\text{radio switching}}$ more clearly, we prefer to also use the figure 22 to illustrate, which was developed in the previous section 5.7 when we were analyzing the extra energy consumption of the radio mode switching in the situation with control packet, since such figure fits for the description of the relationship between the energy consumption of the radio mode switching in one hop and the energy consumption of the radio mode switching of the whole path in both situations.
7 Methodology and Simulation

Before making the decision on whether to employ control packet or not in the field experiment and real deployment, it’s necessary and significant for us to check our design strictly. Simulation is an effective way to examine the result of the energy comparison and evaluate the performance of both methods. In this section, we plan to visualize the comparison of the extra energy consumption that the two types of the medium access mechanisms introduced, study several basic impacts of the network parameters, present the observations of our simulating application as well as show the correlations between the two schemes with respect to control packet.

We simulate our application by using Matlab with the version 7.0.1. To approach the real environment, we use the topology with the given nodes randomly distributed in an area with the size 120 meters by 200 meters, where one of these nodes is referred as the sink, with the endless power supply and with the minimum assumption of its position. We assume the transmission rate is 2Mbps considered in the ideal channel conditions and unchanged status. During the simulation, we firstly set this value by using the unit byte per microsecond in order to be in accord with the time unit of the SIFS, the DIFS, the slot time and the length unit of the control packet and the normal data packet. After the transformation, the transmission rate is 0.25 byte per microsecond. The reason why SMAC introduce duty cycle is because the sampling rate is far smaller than the transmission rate in reality. So in order to prevent nodes listening all the time, SMAC reduces the listen time by making nodes to go into periodic sleep mode. As a result, the sensor nodes will communicate with each other saturatedly in the listen phase even though the sampling rate is smaller than the transmission rate. So the sampling rate must be set smaller than the transmission rate 2Mbps. In addition, due to the limitation T, even the maximum sampling rate is much smaller than the transmission rate. After the simulation figures are outputted, we will change the unit of the sampling rate from byte per microsecond back to megabits per second (Mbps) so as to be in accord with the unified unit. All other values of the parameters we used in our simulation are presented in the following table 1.

In our thesis, we want to know which medium access mechanisms consumes less energy, the one with control packet or the one without control packet in the assumption that the sensor nodes communicate with each other infrequently. Therefore, our final goal is to see how the sampling rate affecting the extra energy consumption in both situations. Intuitively, on one hand, we suppose that when the sampling rate is small enough, the data packets accumulated during the previous sleep phase will be very small, thus the data packets transmitted during the current saturated period of the listen phase will be much less. Therefore, most of the data packets will be transmitted during the unsaturated period where the collision probability will be largely decreased. Since the extra energy consumption of the retransmission is direct proportioned to the collision probability, it seems the extra retransmission energy consumption will become smaller and smaller when the sampling rate decreases. So from this point of view, the control packet RTS and CTS should be deactivated.
On the other hand, the unsaturated phase will become larger as the sampling rate approaches zero in the assumption of the unchanged duty cycle. As a result, the ratio of the number of the data packets transmitted during the unsaturated period to the total number of the data packets transmitted during the whole frame will be increased. As a consequence, the extra energy consumption of the idle listening in the situation without control packet will be increased. From this perspective, the medium access mechanism where control packet is activated is preferable. Whether control packet is an encumbrance or a help at all for the energy saving is still a question. However, before the display of the relationship between the energy consumption and the sampling rate, we will firstly give the figure showing the correlation between the collision probability and the sampling rate since the collision probability plays the role of the intermediate parameter between those two.

<table>
<thead>
<tr>
<th>Basic Parameter</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control message RTS/CTS/ACK</td>
<td>10bytes</td>
</tr>
<tr>
<td>Data message</td>
<td>136bytes</td>
</tr>
<tr>
<td>Fragmentation</td>
<td>4</td>
</tr>
<tr>
<td>Transmission rate</td>
<td>2Mbps</td>
</tr>
<tr>
<td>Maximum backoff stage</td>
<td>5</td>
</tr>
<tr>
<td>Maximum transmission attempt</td>
<td>8</td>
</tr>
<tr>
<td>Minimum contention widow</td>
<td>31</td>
</tr>
<tr>
<td>Sensing field</td>
<td>32000m² (160m × 200m)</td>
</tr>
<tr>
<td>Total sensor node</td>
<td>50</td>
</tr>
<tr>
<td>Nominal transmission range</td>
<td>40m</td>
</tr>
<tr>
<td>SIFS</td>
<td>10 μs</td>
</tr>
<tr>
<td>DIFS</td>
<td>50 μs</td>
</tr>
<tr>
<td>Slot time</td>
<td>20 μs</td>
</tr>
<tr>
<td>Transmission power</td>
<td>31.2mW</td>
</tr>
<tr>
<td>Reception/idle power</td>
<td>22.2mW</td>
</tr>
<tr>
<td>Sleep status</td>
<td>3 μW</td>
</tr>
<tr>
<td>Error rate</td>
<td>0.001</td>
</tr>
<tr>
<td>hops in a routing</td>
<td>5</td>
</tr>
<tr>
<td>Duty cycle</td>
<td>Depends on the sampling rate and the transmission rate</td>
</tr>
<tr>
<td>SMAC frame length</td>
<td>Depends on the message size, the duty cycle and the bandwidth</td>
</tr>
</tbody>
</table>

Table 1. Simulation Parameter List

In the analytical model, the value of the overall average backoff time, the average active transmitting neighbor node and the average neighbor node are required to be integers. However, during the simulation, except that for the average neighbor node, this requirement is very difficult to be satisfied. Since some unsimplified invariables exist in the formulas, which will prevent the solving of the equation set if the floor integers are guaranteed. As a result, in the simulation part, we could only show the asymptotic behavior of the collision probability and the corresponding
energy consumption which are thus showed as curves instead of a series of small straight lines. In
addition, for simplicity and efficiency reasons, we assign the same value of the transmission
power for both control packet and the normal data packet.

7.1 Overall collision probability vs. sampling rate with control packet

1. In saturated period

As we stated before, SMAC is developed under the condition that sensor nodes are in idle
status for a long time during which no sensing event happens in many sensor network applications.
So the fundamental reason SMAC is a more optimized protocol is because it makes use of the fact
that sampling rate is far smaller than the transmission rate in reality. Nevertheless, in the saturated
period of the listen phase, data packets which are accumulated during the previous sleep phase are
transmitted saturatedly.

From the previous equation set presenting the collision probability in the saturated period if
the control packet is applied, we could find the collision probability is only related with the
parameters \( m, K, CW_{\text{min}}, N, a, b, R \) and \( T_{\text{avg}} \) but not \( R_x \). Therefore, we could imagine the figure
showing the relationship between the collision probability and the sampling rate is a straight line
parallel with the axis of the sampling rate. Due to the limitation of the time interval \( T \), the
maximum possible sampling rate in the situation with control packet is 1.177491856 Mbps. This
indicates that the sleep phase must be minimized to zero if the sampling rate is larger than this
value.

After assigning the values to the above parameters and transforming the equation set, the
collision probability can be described as:

\[
\begin{align*}
T_{\text{avg}} &= \frac{991p^9 - 495p^8 - 496p^6 - 14.5p + 15}{2p^9 - p^8 - 2p + 1} \\
T_{\text{avg}} &= \frac{1}{1 - (1 - p)^{\frac{1}{5}}} 
\end{align*}
\]  
(193)

Where \( (1 - p)^{\frac{1}{5}} \) is able to be solved by applying the exponent series:

\[
(1 + x)^m = 1 + mx + \frac{m(m-1)}{2!}x^2 + \cdots + \frac{m(m-1)\cdots(m-n+1)}{n!}x^n + \cdots 
\]  
(194)

Where \(-1<x<1\) and \( m \) is a constant. So after the transformation and unfolding

\[
(1 - p)^{\frac{1}{5}} = 1 - 0.2p - 0.08p^2 - 0.048p^3 
\]  
(195)

Then the equation set (193) is able to be simplified as:
\[
\frac{991p^8 - 495p^8 - 496p^6 - 14.5p + 15}{2p^9 - p^8 - 2p + 1} = \frac{1}{0.2p + 0.08p^2 + 0.048p^3}
\]  \quad (196)

![Graph showing the relationship between collision probability and sampling rate with control packet]

Fig. 29. Relationship between the collision probability and the sampling rate in the saturated period with control packet

Consequently, it facilitates us to evaluate the value of the collision probability in the saturated period with control packet as the figure 29 shows above. As we can observe, the collision probability stands between 0.2 and 0.3, the exact value of the collision probability is 0.21701 in the saturated period no matter whether control packet is applied or not since such collision probability has nothing to do with the message size.

2. **In unsaturated period**

After all the data packets which are sampled during both the previous sleep period and the current saturated period are finished transmitting, the unsaturated period starts. The collision probability in the unsaturated period behaves differently when the sampling rate acts differently, so we should analyze the collision probability in the unsaturated period under different conditions with respect to the sampling rate.

In the first condition where \( T \in (0, \sigma) \), since the average number of the active transmitting neighbor nodes equals to the average number of neighbor nodes, the way of calculating the collision probability is in fact the same as that in the saturated period. Therefore,
the collision probability has no correlation with the sampling rate as well, thus remains the same value as 0.21701 when the sampling rate belongs to the range (1.152542376, 1.1774891856) megabits per second. The figure 30 gives the close visualization of the relationship between the sampling rate and the collision probability under this first condition.

![Fig. 30. Relationship between the collision probability and the sampling rate in the first condition of the unsaturated period with control packet](image)

In the second condition, the average number of the active transmitting neighbor nodes M is related and the time interval T is sampling rate $R_s$ related. So the number of the active transmitting neighbor node is not a single parameter anymore. Instead, it becomes a complex function in which the sampling rate $R_s$ acts as an argument. As a result, the value of the collision probability is changed continuously rather than being constant in the predetermined range of the sampling rate. After simplification, the equation set used to solve the value of the collision probability is transformed to:

$$
\frac{-184.9 p_u R + 131.2 R + 13.6 p_u - 6.8}{73.9 p_u R - 13.6 p_u - 6.8 - 29.2 R} \log \left( \frac{-12.5 p_u + 14}{-14.5 p_u + 15} \right)
$$

$$
= \log \left( \frac{10.7 R p_u + 17 R - 73.9 p_u^2 R + 13.6 p_u^2 - 6.8 p_u}{-18.5 p_u R + 17 R} \right)
$$

(197)

The above formula gives the relationship between the sampling rate and the collision probability in the unsaturated period, from which we obviously know the collision probability acts
as a curve. Thus, for a single value of the sampling rate in the range, there is a corresponding unique value of the collision probability, which is shown in the figure 31.

If we insert the maximum value of the limited range of the sampling rate 1.152542376 Mbps into the above formula (197), we could get the corresponding maximum value of the collision probability within the required sampling range \( (R_{11}, 1.152542376) \) where \( R_{11} \) is the minimum value needed to be evaluated.

Fig.31. Relationship between the collision probability and the sampling rate in the second condition of the unsaturated period with control packet

In the third condition where \( T \in [T_{\text{avg}} \sigma, T_{\text{avg}} \sigma + T_{\text{DIFS}} + T_{\text{gum}}] \), the average number of the active transmitting neighbor node has nothing to do with the sampling rate though it is a complex function containing the parameter the average number of the neighbor node and the overall average backoff time, that is \( M = \left\lfloor \frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 2} N_{\text{neighbor}} \right\rfloor \). So in this case, the collision probability behaves as a straight line as in the first condition and thus has a constant value 0.068327 after solving the following simplified equation:

\[
\frac{-39p_u + 35}{-16.5p_u + 16} \log\left(\frac{-12.5p_u + 14}{-14.5p_u + 15}\right) = \log\left(\frac{33p_u^2 - 50.5p_u + 17}{-18.5p_u + 17}\right)
\]

(198)
So if we insert this value 0.068327 of the collision probability into the equation (198), it’s easy for us to obtain the corresponding value of $R_s$ as the left limit value of the sampling rate in the condition two, which equals to 0.87144 Mbps after calculation. Therefore, the sampling range in the second condition is (0.87144, 1.152542376) Mbps. The following figure 32 presents the collision probability performance in the third condition of the unsaturated period with control packet.

![Fig.32. Relationship between the collision probability and the sampling rate in the third condition of the unsaturated period with control packet](image)

In the fourth condition of the unsaturated period if control packet is applied where $T \in [T_{\text{DIFS}} + \sigma T_{\text{avg}} + T_{\text{trans}}, +\infty)$, the number of the average active transmitting neighbor node is a function of several parameters including the sampling rate and the transmission time needed for sending packets.

$$M = \left[\frac{T_{\text{avg}} + 2}{2T_{\text{avg}} + 3 + \left[\frac{T - T_{\text{DIFS}} - \sigma T_{\text{avg}} - T_{\text{trans}}}{\sigma}\right]}\right] N_{\text{neighbor}}$$  \hspace{1cm} (199)

Therefore, in this case, the collision probability is both sampling rate and packet size dependent, which thus looks like a similar curve as in the second condition, where the curve of the
collision probability comes from the following simplified equation.

\[
\frac{-275.3 p_u R + 176.4 R + 13.6 p_u - 6.8}{164.3 R p_u - 74.4 R - 13.6 p_u + 6.8} \log \left( \frac{-12.5 p_u + 14}{-14.5 p_u + 15} \right) \\
= \log \left( \frac{55.9 p_u R + 17 R - 164.3 R p_u^2 + 13.6 p_u^2 - 6.8 p_u}{-18.5 p_u R + 17 R} \right)
\]

(200)

So the value of the collision probability varies constantly with the change of the sampling rate value. The figure 33 shows the relationship between the collision probability and the sampling rate in the fourth condition of the unsaturated period with control packet. As the figure 24 presents, the curve starts around 0.3 megabits per second, which indicates the collision will never be caused between the sensor nodes in the unsaturated period where control packet is applied if the sampling rate is smaller than this value.

As we stated before, in the third condition, the value of the collision probability is 0.068327, if this value is inserted into the formula (200), we could obtain the maximum limit value of the sampling rate range in the fourth condition, which is 0.505448 Mbps. Thus the interval of the sampling rate in the fourth condition is (0, 0.505448) Mbps. As a result, the interval of the sampling rate in the third condition can be obtained as well, that is, (0.505448, 0.87144) Mbps.

Fig.33. Relationship between the collision probability and the sampling rate in the fourth condition of the unsaturated situation with control packet

3. Overall collision probability
Up to now, we have already simulated the collision probability in the saturated and in the unsaturated period of the listen phase in SMAC. As we discussed before, for a unique sampling rate which is smaller than the transmission rate, there are two corresponding different collision probabilities which are evaluated in the saturated and in the unsaturated period respectively. Therefore, we should average these two collision probabilities by using the total number of the data packets transmitted during each period to obtain an overall average collision probability. The reason we use the number of the transmitted data packets for the average is simply because the value of the collision probability is mostly packet number dependent. The overall collision probability will result from the arithmetic average of the collision probability in the saturated period and the collision probability in the unsaturated period. However, the weighting of the collision probability in the saturated and in the unsaturated period in the situation with control packet is different from that in the situation without control packet due to the different transmission size. As a result, even if the collision probability in the saturated and in the unsaturated period is the same in both the situations with and without control packet, the final overall average collision probability in these two situations will be slightly different.

In this subsection, focusing on the situation where control packet is activated, we will present the overall collision probability and the corresponding average process by figures, which will be simulated under the four different conditions with respect to the time interval $T$ as well.

In the first condition, since the collision probability in the saturated period and in the unsaturated period are both straight lines which indicates they both have nothing to do with the sampling rate, the overall collision probability in this case after the average is also a straight line paralleled the axis of the sampling rate, which looks the same as the collision probability in the unsaturated period shown by the previous figure (30).

In the second condition, as it has been known already, the collision probability in the saturated period is a straight line while the collision probability in the unsaturated period is a curve. Thus, the overall collision probability will be acted as a smoother curve after the average. This can be observed from the figure 34 where the blue and the red line presents the collision probability in the saturated and in the unsaturated period respectively and the black curve describes the final overall collision probability in the whole listen phase of the SMAC protocol.
In the third condition, the collision probability in both the saturated and the unsaturated period behave as straight lines. This indicates both of these collision probabilities have no correlation with the sampling rate. Nevertheless, the overall collision probability acts as a curve due to that the weighting of these two partial collision probabilities in the average are both sampling rate dependent. As a result, the overall collision probability has a unique value in each sampling rate within the third interval, which is clearly showed as the black curve in the following figure 35.
In the fourth condition where control packet is activated, the collision probability in the saturated period is not related with the sampling rate. As the consequence, it looks as a straight line in the sampling rate interval \((0, 0.505448)\) Mbps. However, affected by the sampling rate, the collision probability in the unsaturated period behaves as a curve which intersects with the axis of the sampling rate at nearly 0.3 Mbps even though the allowed rate interval begins from zero Mbps. After the average, the overall collision probability under the fourth condition appears as a curve as well as that in the unsaturated period. However, it appears from the beginning when the sampling rate is zero. The reason is simply because the collision probability in the saturated period, however, still remains nonzero in this partial interval of the sampling rate ranging from zero Mbps to 0.3 Mbps. The relationship among the collision probability in the saturated period, the collision probability in the unsaturated period and the overall collision probability is visually described in the following simulation figure 36.
Now we will combine together all these four figures showing the average collision probability in the four different conditions with respect to the sampling rate in order to give an overall look of the performance of the collision probability in the whole listen phase if control packet is applied as the following figure 37 shows. Since the maximum value of the overall average collision probability is lower than 0.3, to give a closer and clearer observation, we decide to show the variation of such overall average collision probability in the range of (0, 0.3) where the sampling rate is in the range (0, 1.1774891856) megabits per second, as the figure 38 displays below.

Deserve to be mentioned, there is a gap between the lines of the average collision probability in the first and in the second condition with respect to the sampling rate, which is denoted by a small black straight line paralleled with the axis of the overall collision probability. This phenomenon appears since in the first condition of the unsaturated period with control packet, it requires that $1.1774891856 < R_s < 1.152542376$ Mbps, however, in the second condition of the unsaturated period, $0.87144 < R_s \leq 1.152542376$ Mbps is required. Besides, since the function of evaluating the number of the active transmitting neighbor node is different under these two conditions, the corresponding appearance of the line type of the average collision probability is different. As a result, once the sampling rate decreases to the value 1.152542376 Mbps, the average collision probability will decrease a big step. Not a like, there is no gap between the lines of the average collision probability in the third condition and in the fourth condition, since both of...
these lines are curves rather than one of the lines is straight or both of the lines are straight.

Fig.37. The overall collision probability in the situation with control packet

Fig.38. A closer observation of the variation of the overall collision probability in situation with
7.2 Overall collision probability vs. sampling rate without control packet

1. In saturated period

The figure showing the variation of the collision probability according to the sampling rate in the saturated period without control packet also looks as a straight line paralleled the rate axis, since such collision probability is also independent of the sampling rate. However, due to the different transmission size $T_{trans}$, the time interval $T$ in the two situations with and without control packet is different. As a result, the maximum reasonable sampling rate which evaluated based on the time interval $T$ acts differently in the two situations with respect to control packet. After the calculation, the maximum allowed sampling rate in the situation without control packet is 1.320388248 Mbps, which is larger than the maximum sampling rate in the situation with control packet, showed in the figure 39.

![Graph showing p vs. Rs in the saturated period without control packet](image)

Fig.39. The relationship between the collision probability and the sampling rate in the saturated period without control packet

2. In unsaturated period

In the unsaturated period where control packet is deactivated, since the control packet RTS and CTS in each transmission process are deleted, the total transmission size is minimized. Since the four different conditions under the unsaturated period are classified by the time interval $T$ where $T$ is, however, the transmission size related, the interval of the sampling rate in each of the four conditions in the unsaturated period where control packet is deactivated is different from that in the unsaturated period where control packet is activated.
In addition, besides the change of the sampling rate range in each condition, the shape of the collision probability line simulated in the unsaturated period without control packet changes as well. However, this change only occurs under the second and the fourth condition, since only the formulas which display the average number of the active transmitting neighbor nodes $M$ in these two conditions vary due to the changed time interval $T$. Not a like, the average number of the active transmitting neighbor nodes $M$ is independent of the transmission size under the first and the third condition, thus, the shape of the lines of the corresponding collision probability under these two conditions in the situation without control packet remains the same as that in the situation with control packet.

In the first condition of the unsaturated period without control packet, from the figure 40, apparently, the sampling rate should be smaller than the transmission rate 2Mbps. Both the left limit and the right limit of the sampling rate interval in the first condition can be calculated directly by making use of the time interval $T$, which are 1.28909952 Mbps and 1.320388248 Mbps respectively. Since the number of the active transmitting neighbor nodes equals to the average number of the neighbor nodes, the collision probability in the first condition of the unsaturated period keeps the same value as 0.21701.

![Fig.40. Relationship between the collision probability and the sampling rate in the first condition of the unsaturated period without control packet](image)

In the second condition during the unsaturated period without control packet, the RTS and the CTS frames are eliminated from the packet transmission, thus the transmission size $T_{\text{trans}}$ is decreased, which affects the value of the time interval $T$. Since $T$ is one of the parameters which
influence the number of the active transmitting neighbors $M$ and $M$ is however the very significant variable deciding the value of the collision probability in the unsaturated period, the formula which evaluates the performance of the collision probability in the second condition in the unsaturated period without control packet must be re-analyzed.

$$\frac{-174p_R + 126R + 136p_u - 68}{639p_R - 242p_u + 68 - 136R} \log(\frac{-125p_R + 14}{-145p_u + 15}) = \log\left(\frac{57p_R + 17R - 639p_u^2 + 136p_u^2 - 68p_u}{-185p_R + 17R}\right)$$ (201)

Obviously, the value of the right limit of the sampling rate interval in the second condition equals to the left limit of the sampling rate interval in the first condition in the unsaturated period, which is 1.28909952 Mbps. Unfortunately, the value of the left limit of the sampling rate interval in the second condition can not been known directly after calculation by using the time interval $T$. However, this can be finally solved out after the collision probability in the third condition is successfully estimated. The figure showing the relationship between the collision probability and the sampling rate in the second condition of the unsaturated period without control packet is presented below.

![Fig. 41. The relationship between the collision probability and the sampling rate in the second condition of the unsaturated period without control packet](image)

In the third condition of the unsaturated period, the value of the collision probability does not change at all due to the fact that the number of the active transmitting neighbor node $M$ is independent of the increased time interval $T$ in the situation where control packet is deactivated. Therefore, the collision probability in this case looks the same as that in the situation with control packet although it appears in the different sampling rate zone, as the figure 42 described below.
If we insert the value of this collision probability 0.068327 into the formula which gives the relationship between the collision probability and the sampling rate in the second condition in the situation without control packet, the value of the left limit of the sampling rate interval of the second condition can be found out, which is also the value of the right limit of the sampling rate interval of the third condition. In conclusion, the sampling rate ranges between 0.94728 Mbps and 1.28909952 Mbps under the second condition of the unsaturated period where control packet is deleted.

Comparing with the collision probability in the situation with control packet, the collision probability in the fourth condition of the unsaturated period without control packet changes slightly due to the change of the number of the active transmitting neighbor nodes M caused by the increased time interval $T$ and the reduced transmission size $T_{\text{trans}}$. After the simplification and the transformation, the equation set which gives the relationship between the collision probability and the sampling rate in the fourth condition of the unsaturated period without control packet can be written as:

$$
\begin{align*}
& - \frac{255.3 p_u R + 166.4 R + 13.6 p_u - 6.8}{144.3 p_u R - 64.4 P_u + 6.8 - 13.6 R} \log \left( \frac{-12.5 p_u + 14}{-14.5 p_u + 15} \right) \\
= & \log \left( \frac{45.9 R p_u + 17 R - 144.3 p_u^2 R + 13.6 p_u^2 - 6.8 p_u}{-18.5 p_u R + 17 R} \right)
\end{align*}
$$

(202)
Before simulating the result of this collision probability, we should firstly find out the allowed sampling rate range under this condition. Obviously, the sampling rate starts from zero Mbps, but the right limit of the rate interval is still an unknown number. However, since the value of the collision probability under the third condition is only a constant, the value of the right limit of the sampling rate interval in the fourth condition will be able to be eventually estimated if we insert this constant into the above formula (202). After calculation, the value of the right limit of the sampling rate interval in the fourth condition equals to 0.557224 Mbps, which is also the value of the left limit of the sampling rate interval in the third condition. In short, the sampling rate interval of the third condition and the fourth condition in the unsaturated period without control packet is (0.557224, 0.94827) Mbps and (0, 0.557224) Mbps respectively, which can be clearly noticed from the above figure 42 and following figure 43.

![Fig.43. The relationship between the collision probability and the sampling rate in the fourth condition of the unsaturated period without control packet](image)

3. Overall collision probability

Up to now, we have already successfully simulated the collision probability in the saturated and in the unsaturated period in the situation without control packet. In this subsection, we are going to average this two collision probabilities to achieve an overall collision probability by using the weighting developed in terms of the total number of the transmitted data packets during the saturated and the unsaturated period respectively. Since the weighting evaluated in the situation without control packet is a little changed due to the altered transmission size, the value of it is different from that in the situation with control packet. From now on, we will focus on analyzing the overall collision probability in each condition in the situation where control packet is deactivated and their average process.
In the first condition without control packet, the line representing the average collision probability looks the same as that in the unsaturated period, since the collision probability in the saturated period is also a constant which shares the same value with the collision probability in the unsaturated period. As a result, averaging two one and the same values indicates that the average value equals to both of these values no matter what weighting is used for the average. Therefore, the overall collision probability in the first condition without control packet is still 0.21701 within the sampling rate range (1.28909952, 1.320388348) Mbps.

Fig.44. The overall collision probability in the first condition in the situation without control packet

In the second condition where control packet is deactivated, the result of the average of the collision probability in the saturated and in the unsaturated period behaves as a smooth curve within the sampling rate range (0.94728, 1.28909952) Mbps. If we insert this left and right limit sampling rate value 0.94728 Mbp and 1.28909952 Mbp into the formula which shows the relationship between the sampling rate and the overall collision probability in the second condition respectively, the minimum and the maximum value of this average collision probability in the second condition is able to be found, which equals to 0.12482 and 0.21118 respectively, which can be generally observed from the figure 45.
In the third condition without control packet, the overall collision probability acts as a curve from the sampling rate 0.55724 Mbps to the sampling rate 0.94728 Mbps even though the collision probability in the saturated period behaves as a straight line paralleled the axis of the sampling rate. If we insert this value of the left limit of the rate interval into the formula presenting the relationship between the overall collision probability and the sampling rate in the third condition, the minimum value of this overall collision probability in the situation without control packet can be successfully evaluated. After calculation, it equals to 0.10955, which is also the maximum value of the overall collision probability in the fourth condition where control packet is deactivated.
In the fourth condition, the overall collision probability changes as the change of the sampling rate from zero Mbps to 0.557224 Mbps, as the black curve displays in the following figure 47, where the blue and the red line presents the variation of the collision probability in the saturated and in the unsaturated period in the fourth condition respectively. If we insert zero which is also the value of the left limit of the sampling rate interval into the formula showing the relationship between the overall collision probability and the sampling rate in the fourth condition, the minimum value of this overall collision probability is able to be discovered, which equals to 0.043404 after calculation.
Now we have already simulated the average collision probability based on the collision probability in the saturated and in the unsaturated period in each condition, if we combine all these four separated average collision probabilities, the final overall collision probability in the situation without control packet displayed within the entire sampling rate range (0, 1.320388248) Mbps can be obtained, as the figure 48 shows below. If you are a careful observer, you might be aware that there is a small vertical black line connecting the yellow line and the green curve. This phenomenon occurs because the sampling rate 1.28909952 Mbps is a critical point of the first and the second condition under which the method for the evaluation of the number of the active transmitting neighbor nodes is different. As we can see, the value of the overall collision probability in the situation without control packet varies between 0.043404 and 0.21701. As the sampling rate within the permitted range increases, the corresponding overall collision probability increases as well.
Since the value of the overall collision probability won’t change anymore if the sampling rate is larger than 1.28909952 Mbps and the value of the overall collision probability is always lower than 0.3, we prefer to give a closer observation of the picture of the overall collision probability in the situation without control packet, as the figure 49 presents.
7.3 Extra energy consumption vs. sampling rate with control packet

Up to now, we have successfully found the relationship between the sampling rate and the overall collision probability. Since the overall collision probability is an intermediate parameter between the sampling rate and the extra energy consumption. If we insert the formula in which the overall collision probability and the sampling rate are the only two parameters into the formula presenting the relationship between extra energy consumption and the overall collision probability, the final relationship between the extra energy consumption and the sampling rate in the situation with control packet is able to be developed.

The line showing the extra energy consumption in the situation with control packet is also divided into four parts where the line acts straight in the first part and the line is shaped into a curve in the remaining second, third and fourth part. When we consider the extra energy consumption, we assume the error probability is the same for all types of the packets, that is, \( p_{rj}, p_{sij}, p_j \) and \( p_{aj} \) has the same value 0.001. Since the value of the extra energy consumption is stable when the sampling rate is larger than 1.152542376 Mbps, we would like to magnify the picture in order to make every four part of the line looks clearly. From the figure 50, we could notice that the larger the sampling rate, the more energy will be consumed in the transmission, retransmission, reception and overhearing of the RTS and CTS frames, the sleep by the neighbors of the transmitter and the receiver as well as the radio mode switching if the sensor node transmits both control packet and normal data fragments during the saturated and the
unsaturated period.

![Graph](image)

**Fig.50. Relationship between the extra energy consumption and the sampling rate in the situation with control packet**

### 7.4 Extra energy consumption vs. sampling rate without control packet

As we can observe from the figure 51, totally different from the variation of the extra energy consumption in the situation with control packet, the extra energy consumption decreases as the sampling rate increases in the situation without control packet, besides, the average value of the total energy consumption in this case is much larger than that in the situation with control packet. The mainly reason of this special phenomenon comes from the extra energy consumed in the idle listening by the transmitter, the receiver and all their neighbors. Since such extra energy consumption only appears during the unsaturated period which has a direct proportion to the number of the transmitted data packets, we should add a coefficient in front of the energy item, where the numerator of the coefficient is the total number of the data packets transmitted during the unsaturated period while the denominator is the total number of the data packets transmitted during the whole listen phase. After the calculation and the simplification, we found that the value of this coefficient is mostly depended on the sampling rate. Moreover, the larger the sampling rate is, the smaller the extra energy of the idle listening will be, as the following formula presents.

\[
E_{\text{idle\_listening}} = \frac{97.68 - 488.4R_s}{1 - 4R_s}
\]  

(203)

However, the comparison of the extra energy consumption in the situation with and without control packet only makes sense when the sampling rate in both cases shares the same value. Since
the maximum permitted value of the sampling rate in the situation with control packet (1.1774891856 Mbps) is smaller than that in the situation without control packet (1.320388248 Mbps), this extra energy consumption of the idle listening only plays the role when the sampling rate is smaller than 1.1774891856 Mbps. And this is the exact reason that this extra energy consumption of the idle listening is not included in the total extra energy consumption in the first condition and in the part of the second condition where the sampling rate is larger than 1.1774891856 Mbps. And this phenomenon can be easily read from the figure 51 where the yellow line and the green line showing the total extra energy consumption have a smaller value than the total extra energy consumption the other curves present. In the other part of the second condition, however, the total energy consumption comprises the extra idle listening energy consumption, showed by the pink curve in the figure 51, from which we might be aware that the extra idle listening energy occupies a high percentage in the total extra energy consumption in the situation without control packet.

If we use the formula to display the relationship between the extra energy consumption and the sampling rate in the situation without control packet, this formula must be a segment function divided by the critical point of the sampling rate 1.774891856 Mbps.

$$E_{\text{without}} = \begin{cases} \frac{21.258517}{1-p} + 82.63748 & 1.1774891856 < R_s < 1.320388248 \\ \frac{21.258517}{1-p} + 82.63748 \left(1 - \frac{97.68 - 4884R_s}{1 - 4R_s}\right) & 0 \leq R_s < 1.1774891856 \end{cases}$$  

(204)

Fig. 51. Relationship between the extra energy consumption and the sampling rate in the situation without control packet.
without control packet

7.5 Comparison between the energy consumption with and without control packet

In our thesis, we focus on the analysis of the tradeoff between the energy consumption if control packet is activated and the energy consumption if control packet is deactivated when the sampling rate is lower than the transmission rate. We will choose one of these two medium access mechanisms depending on the amount of the energy consumption. Intuitively, from the perspective of the extra retransmission energy that collision introduces, we may be aware that when the sampling rate is high, the collision probability will be increased in the unsaturated period, thus huge amount of the energy might be spent on the retransmission of the packet. Therefore, to decrease the retransmission energy, it’s maybe better to introduce control packet since the size of control packet is much smaller than that of the normal data fragment. However, when the sampling rate is very small, the collision will seldom occur on the transmission packet during the unsaturated period, thus the retransmission energy will be very small. In this case, if control packet is used, the energy that control packet brings might be larger than the energy that control packet avoids. Thus, the medium access mechanism without control packet may win the competition.

However, besides the retransmission energy caused by the collision among the packets, there are some other types of the extra energy consumption in both the situation with and without control packet. Among them, the extra energy consumption of the idle listening occurred in the situation without control packet plays a special role, since such extra energy consumption only appears in the unsaturated period. As a result, we should add an additional coefficient in front of the item which presents the extra idle listening energy between the transmissions of two packets communicated during the unsaturated period. After the careful analysis, we found such coefficient is mostly related with the sampling rate. With the help of the simulation by Matlab, the curve showing the extra energy consumption in the situation without control packet tells the fact that such extra energy consumption of the idle listening has an inverse proportion to the sampling rate and the amount of the extra energy that idle listening brings occupies a high percentage in the entire energy consumption in the situation without control packet.

Learning from the simulation figure 52, where the blue curve and the red curve gives the visualization of the total extra energy consumption in the situation with and without control packet respectively, we are able to eventually discover that the medium access mechanism with control packet in the SMAC protocol is a better choice out of question no matter what value the sampling rate is.
8 Conclusion and Future Work

In this thesis, we present an analytic model to analyze the trade off between the energy consumption when control packet is activated and the energy consumption when control packet is deactivated based on the SMAC protocol in the assumption of the infrequent communication among the sensor nodes in the wireless network. When control packet is applied, the extra energy is consumed in the communication of the control packet RTS and CTS frame, the overhearing of the RTS and CTS frame, the long sleep time and the radio mode switching. However, if the control packet is eliminated, the extra energy is consumed in the retransmission of the first data fragment due to collision, the overhearing of the first data fragment and ACK frame, the idle listening and the radio mode switching.

Since the collision probability in the saturated period is a constant in both the situations with and without control packet, the collision probability in the unsaturated period is the only intermediate parameter between the overall collision probability and the sampling rate. Thus, we firstly show the relationship between the collision probability in the unsaturated period and the sampling rate in both the situations with and without control packet. We notice that as the sampling rate increases, the collision probability in the unsaturated period increases as well until it reaches the maximum value, after which such collision probability dose not change anymore. Then we present the relationship between the overall collision probability and the sampling rate.
though simulation figures after we average the collision probabilities in the saturated and in the
unsaturated period in both the situations with and without control packet. By doing so, we find a
method to finally discover the relationship between the extra energy consumption and the
sampling rate in both cases with respect to control packet, since the overall collision probability
acts as the intermediate parameter between the extra energy consumption and the sampling rate.
Observing from the figures, we learn that the two lines which display the extra energy
consumption in the situation with and without control packet never intersect, from which we can
eventually draw the conclusion that the medium access mechanism with control packet in the
SMAC protocol is a better choice regardless of the value of the sampling rate.

In our analytical model, we require that the average number of the neighbor node, the
average number of the active transmitting neighbor node and the overall average backoff time are
all integers. However, during the simulation, this requirement is met only on the average number
of the neighbor node. So in the future, we will attempt to conquer this difficulty to assess the
performance of the energy consumption more accurately. In addition, the duty cycle is another
significant parameter affecting the performance of the extra energy consumption. In our thesis, we
assume the duty cycle in the situation with control packet equals to that in the situation without
control packet. Since the transmission size is different in these two cases, the saturated phase
during which all the data packets accumulated in the queue are sent out is different. As a result, the
unsaturated period is in the situation with control packet is different from that in the situation
without control packet if the sampling rate shares the same value. However, in reality, the duty
cycle should be changed according to the different medium access mechanism. Therefore, it will
be more interesting and meaningful if we can set up the relationship among the sampling rate, the
duty cycle and the extra energy consumption.
References


