In this chapter we will analyze the performance of a wireless sensor node as a tandem queue. Instead of directly modeling sensor node as a single entity, we have tried to capture inside functioning of a sensor node with the help of tandem queue. Two main power consuming units of a sensor node- computation unit and communication unit we have considered as two servers connected in tandem. Here output of computation unit acts as an input for the communication unit. A wireless sensor node with various capabilities like fixed service rate, only DVFS, only DMS, both DVFS and DMS and both coordinated together is considered and performance analysis carried out. MATLAB simulation of a wireless sensor network node is carried out to check and compare the performance of sensor node with various capabilities.
Chapter 4. Tandem Queue Model and Simulation of Sensor Node

4.1 Tandem queue model of sensor node

Single server, exponential queuing system with finite buffer capacity ($M/M/1/K$ model) has been well established in literature [148] [149]. From the sensor node architecture (see Figure 3.1) it is possible to model individual sensor node as two servers with finite queues, connected in tandem (output of the first becomes input for the second). Equivalent tandem queue model is shown in Figure 4.1. In this model first server refers to the computing unit (processor or microcontroller) and second server is a radio transmitter. Data sensed by various sensors and data received by the receiver comes in at the input buffer and gets processed by the computation unit. Processed data is temporarily stored in output queue before transmission. Input and output buffers are very small in size as wireless sensor nodes are very small in size. We have considered periodic rotational sleep schedule (described in Chapter 2) for sensor nodes as it provides better synchronization, field coverage and network connectivity. Sensor nodes wake up (ON state) for a fixed duration and then go to sleep (power saving state) for a fixed duration. We refer to sleep state as OFF state. Duration of ON state and OFF state are predefined such that required low duty cycle operation can be achieved. Sensor nodes keep on changing between OFF state and ON state. Duty cycle varies application to application and also depends on the sensor node density in the network. Higher the duty cycle more stringent is the need for power optimization.

Figure 4.1: Tandem queue model of Wireless Sensor Node
Let

\[ A_n = \text{number of packets arrived during } n^{th} \text{ slot} \quad 0 \leq A_n \leq 6 \]

\[ B_1 = \text{maximum input buffer size} \quad B_1 = 6 \text{ packets} \]

\[ B_2 = \text{maximum output buffer size} \quad B_2 = 6 \text{ packets} \]

\[ f = \text{service rate of first server} \quad (\text{depends on clock frequency}) \]

\[ b = \text{service rate of second server} \quad (\text{depends on transmission bit rate of transmitter}) \]

Here service rate implies the maximum number of packets that can be served in one time slot.

\[ M_n = \text{input buffer occupancy at the end of } n^{th} \text{ slot} \quad 0 \leq M_n \leq B_1 \text{ packets} \]

\[ N_n = \text{output buffer occupancy at the end of } n^{th} \text{ slot} \quad 0 \leq N_n \leq B_2 \text{ packets} \]

We have considered Late Arrival System (LAS) where data packets are allowed to enter in the system just before the slot ends. These packets get service in the next time slot. Input buffer occupancy at the start of a time slot is dependent on buffer occupancy at the start of previous slot, number of packets served and number of packets arrived during previous slot.

\[ M_n = \min\{\max\{(M_{n-1} - f), 0\} + A_{n-1}, B_1\} \quad (4.1) \]

Similarly output buffer occupancy can be written as,

\[ N_n = \min\{\max\{(N_{n-1} - b), 0\} + \min\{M_{n-1}, f\}, B_2\} \quad (4.2) \]

As discussed in chapter 2, DVFS and DMS are power optimization techniques used for processor and transmitter respectively. Here we will implement these techniques one by
one and then together on a sensor node tandem queue model to study effect on buffer congestion (overflow) and power consumption.

4.1.1 Wireless sensor node with fixed service rates (No DVFS, No DMS)

In most of the WSN applications during normal periods very less workload needs to be handled. When the number of packets in the buffer is less than the number of packets that can be served in one time slot of duration $\Delta t$, server remains idle for some period. We can define $I_{1n}$ as the Idle period of first processor in the $n^{th}$ time slot and can be given as,

$$I_{1n} = \max\{(f - M_{n-1})/f, 0\} \cdot \Delta t \quad (4.3)$$

Similarly, we can define $I_{2n}$ as the Idle period of second processor in the $n^{th}$ time slot and can be given as,

$$I_{2n} = \max\{(b - N_{n-1})/b, 0\} \cdot \Delta t \quad (4.4)$$

For a fixed service rate sensor node, $M_n$ is a function of service rate and arrival rate. As service rate is fixed $M_n$ depends on arrival rate only. During normal period, arrival rate is very small which keeps the value of $M_n$ also small and as a result processor remains idle over a longer period. On the contrary during catastrophic conditions the arrival (number of packets arrived in one slot) increases but as the service rate is fixed and buffer size is small possibility of data loss due to buffer congestion (buffer overflow) occurs as per the head drop or tail drop scheme. Input buffer overflow ($OV_1$) occurs when $M_n = B_1$ and output
buffer overflow \((OV_2)\) occurs when \(N_n = B_2\).

\[
OV_{1n} = \max\{\max\{M_{n-1} - f, 0\} + A_{n-1} - B_1, 0\}\]  
\[
OV_{2n} = \max\{\max\{N_{n-1} - b, 0\} + \min\{M_{n-1}, f\} - B_2, 0\}\]

For example, consider a sensor node with fixed service rate \(f = 0.5 f_{max}\) and \(b = 0.5 b_{max}\) i.e. \(f = b = 3\) packets per time slot. \(B - 1 = B_2 = 6\) packets. During normal time period, packet arrival rate is very small, if it is just 1 packet per time slot then from above equations we can find out that sensor node remains idle for 66.66% of the time slot and there is no buffer overflow. But when we consider catastrophic time period \((A_n = 5\) packets per time slot), only 3 packets can be served by the first processor and the same 3 packets can be handled by the second processor. So there is no output buffer overflow but packets get rejected at input buffer resulting in 3 packets overflow in each time slot.

### 4.1.2 Wireless sensor node with only DVFS, No DMS (variable \(f\) and fixed \(b\))

In this case input buffer status depends on the arrival rate as well as service rate \(f\). Arrival rate cannot be controlled but service rate can be used as a control knob to reduce idle period during normal times and buffer overflow during catastrophic period. In this case

\[
M_n = \min\{\max\{M_{n-1} - f_{n-1}, 0\} + A_{n-1}, B_1\}\]

This value of \(M_n\) will decide the value of service rate \(f_n\). Smaller the value of \(M_n\) smaller value of \(f_n\) will be selected. Reduction in service rate will reduce the power consumption (smaller value of \(f\) will need lesser supply voltage and reduce power consumption) and
will take more time to complete the service (DVFS). It helps to reduce the idle power wastage. Idle period of first processor in \(n^{th}\) time slot can be given as,

\[
I_{1n} = \max\left\{\left(\frac{f_n - M_{n-1}}{f_{\max}}, 0\right) \cdot \Delta t\right\} \tag{4.8}
\]

By reducing the value of \(f_n\) idle time will be reduced and will save the power. Buffer overflow is given as,

\[
OV_{1n} = \max\left\{\left(\max\left\{\left(M_{n-1} - f_{n-1}\right), 0\right\} + A_{n-1} - B_1\right), 0\right\} \tag{4.9}
\]

During catastrophic condition as the arrival rate increases value of \(M_n\) will be more. Data loss due to input buffer overflow can be reduced by increasing the value of \(f_n\). In order to make service rate buffer adaptive we need to scale \(f_n\) in terms of \(M_n\).

\[
f_n = \left(\frac{M_{n-1} \cdot f_{\max}}{B_1}\right) \tag{4.10}
\]

\(f_{\max}\) is the maximum supported service rate. But as the second server in the tandem queue (transmitter) works with fixed service rate, there is possibility of data loss during catastrophe and more power wastage during idle period. Output buffer occupancy is-

\[
N_n = \min\left\{\max\left\{(N_{n-1} - b), 0\right\} + \min\{M_{n-1}, f_n\}, B_2\right\} \tag{4.11}
\]

Output buffer overflow can be written as,

\[
OV_{2n} = \max\left\{\left(\max\left\{(N_{n-1} - b), 0\right\} + \min\{M_{n-1}, f_n\} - B_2\right), 0\right\} \tag{4.12}
\]
In this equation $f_n$ is varying at the first server but there is no control knob at the second server to control the overflow. During catastrophe as $A_n$ increases, $f_n$ at the first server will increase resulting in increased $N_n$. As $b$ is constant and $B_2$ is fixed, output buffer overflow increases. It not only results in data loss but as the processed data gets lost, processing power used for that also goes waste. Similarly during normal conditions as $A_n$ reduces, $f_n$ will be reduced. It will reduce the packet arrival rate in the output buffer but as second server works with fixed rate (which is high enough to handle worst case condition), it will remain idle over longer duration and more power will be wasted. Idle period of transceiver in $n^{th}$ time slot can be given as,

$$I_{2n} = \max\{(b - N_{n-1})/b, 0\} \cdot \Delta t \quad (4.13)$$

$N_{n-1}$ becomes smaller due to reduced $f_n$ but $b$ is constant and moderately high hence $I_{2n}$ increases. Implementation of only DVFS is not enough as it increases the processed data loss and processing power loss during catastrophe and more idle power wastage during normal period.

For example, consider a sensor node with variable $f_n$ (DVFS) and fixed $b= 0.5 \; b_{max}$ (no DMS). During normal time period, packet arrival rate is very small, if it is just 1 packet per time slot then from above equations, value of $f_n$ selected will be 1 packet per time slot and hence $I_1 = 0$ but second server remains idle for 66.66% of the time slot and there is no buffer overflow. But when we consider catastrophic time period ($A_n = 5$ packets per time slot),$f_n$ will be set to 5 packets per time slot and hence $OV_1 = 0$ but second server works with constant service rate and hence can not handle the increased value of $N_n$ and results in the buffer overflow.
4.1.3 Wireless sensor node with only DMS, No DVFS (variable $b$ and fixed $f$)

In this case service rate of processor ($f$) is fixed and kept considerably high in order to handle sufficient number of packets during catastrophic conditions. Transmission rate $b$ can be varied as per the number of packets in the output buffer. During normal conditions $f$ will be much higher than the arrival rate $A_n$, so the first server-processor remains idle over a longer duration. Idle power wastage is more likely in the first server. Number of packets entering in the output buffer is also small during normal conditions. Second server-transmitter will select lower service rate $b$ for the transmission of packets. Lowering the transmission rate will reduce the power consumption (RF power with DMS). Similarly during catastrophic conditions $A_n$ will increase and if $f$ is not sufficiently high then the data loss may occur at input buffer. At the output buffer data loss due to buffer congestion is reduced by increasing the transmission rate (more number of bits per symbol).

$$OV_{2n} = \max\{\max\{N_{n-1} - b_n\}, 0\} + \min\{M_{n-1} - f, B_2\}, 0\} \quad (4.14)$$

Here though $b_n$ is changing, $OV_{2n}$ gets limited by $f$ which is fixed. So implementation of only DMS is also not enough.

For example, consider a sensor node with fixed $f = 0.5 \ f_{\text{max}}$ (no DVFS) and variable $b_n$ (DMS). During normal time period, packet arrival rate is small but value of $f$ is fixed, 3 packets per time slot and hence $I_1 = 0.66$ of total time slot but second server can adjust its service rate to 1 packet per slot and idle time period can be reduced to zero and also there is no buffer overflow. But when we consider catastrophic time period ($A_n = 5$ packets per time slot), fixed value of $f$ will make the number of packets arriving in the output buffer constant and hence though second server is capable of increasing the service rate, it will
work with the fixed rate (equal to the service rate of first server) and hence $OV_2 = 0$ but all packets will not be allowed to enter in the sensor node due to input buffer overflow.

### 4.1.4 Both DVFS and DMS integration (variable $b$ and $f$)

From the discussions above it is highly desirable to have both $f$ and $b$ changing w.r.t. the number of packets in the buffer waiting for the service. Figure 4.2 shows the sensor node architecture with variable processing rate and variable transmission rate. Here a monitor checks the queue length and the probability of buffer overflow. Processing rate of the processor is varied as per the principle of dynamic voltage/frequency scaling (DVFS) and the data transmission rate is varied using dynamic modulation scaling (DMS). For DVFS input buffer is monitored while for DMS output buffer is monitored.

For example a sensor node with both DVFS and DMS capabilities can reduce the service rate of both the servers during normal period and will reduce idle time periods and will increase the service rates during catastrophic period and will reduce data loss due to buffer overflows. Same results are checked with MATLAB simulation and presented in the next section.

![Figure 4.2: Sensor node model with DVFS and DMS](image)
As seen earlier input and output buffer occupancies can be given as,

\[ M_n = F\{A_n, f\} \quad 0 \leq M_n \leq B_1 \]  \hspace{1cm} (4.15)

\[ N_n = F\{f, b\} \quad 0 \leq N_n \leq B_2 \]  \hspace{1cm} (4.16)

for the stability of the system it is required to have \( f \geq A_n \), so that the departure rate of the first server is nothing but its arrival rate \( A_n \). So we can approximate,

\[ N_n = F\{A_n, b\} \quad 0 \leq N_n \leq B_2 \]  \hspace{1cm} (4.17)

This shows that the occupancy of input buffer as well as output buffer is a function of arrival rate \( A_n \). Implementing DVFS (on processor) and DMS (on transmitter) together on a sensor node makes the service rates \( f \) and \( b \) to change w.r.t. input and output buffer occupancy respectively. This will save power during normal periods and will reduce data loss due to buffer overflow during catastrophic periods. Also as both the buffer occupancies are function of arrival rate \( A_n \) (directly proportional) there is no need to monitor input and output buffers separately. By monitoring input buffer only it is possible to select required \( f \) and \( b \). Now we can say that \( f \) and \( b \) are changing in coordination.

We implemented both DVFS and DMS together on a sensor node and carried out MATLAB simulation to study the input and output buffer lengths. Figure 4.3 and Figure 4.4 shows input buffer length \((q1)\) and output buffer length \((q2)\) at various time slots.

These figures show that whenever input queue length increases, output queue length also increases (not necessarily by same amount) but after some time interval (time delay). This time delay may be of one time slot or the processing time required for the task. These figures support the idea of monitoring only input buffer and varying service rates of both the buffers with some delay in between. It removes the need for observing output buffer
Chapter 4. Tandem Queue Model and Simulation of Sensor Node

Figure 4.3: Input and output buffer lengths during time slots 1 to 100

Figure 4.4: Input and output buffer lengths during time slots 170 to 270
separately. It results in coordinated adaptive power (CAP) management giving extended lifetime to the sensor nodes and indirectly contributing to the lifetime extension of WSN. Figure 4.5 shows the concept of CAP management where processing rate and transmission rate are varied by monitoring input buffer only and without the need to monitor output buffer. It also ensures QoS by reducing the buffer overflow and data loss because of it.

![Figure 4.5: Coordinating DVFS and DMS](image)

### 4.2 MATLAB simulation of tandem queue model

Tandem queue equivalent model of wireless sensor node we simulated in MATLAB 6.5. Each server is having a limited capacity buffer with capacity $B_1$ and $B_2$ respectively. $f_1$ and $b_1$ are the service rates of server1 and server2 respectively in active low state while $f_2$ and $b_2$ are the service rates in active high state. $\lambda$ is the arrival rate of requests in the system. It is considered that the arrival and service rates are Poisson, capacity of buffers is finite and here considered 6 packets. We simulated the models of sensor node with fixed service rate, sensor node with only DVFS capacity, sensor node with only DMS capacity and sensor node with DVFS and DMS applied together. Figure 4.6, Figure 4.7 and Figure 4.8 shows the comparison of a sensor node with fixed service rate, having only DVFS implemented,
with only DMS implemented and finally both DVFS and DMS integrated together on a wireless sensor node.

### 4.2.1 Simulation results for tandem queue model of sensor node

A sensor node is desired to be capable of handling heavy traffic without much loss of information as well as the power optimized one in order to have longer life. A compromising service rate of 0.5 \((f_1 = f_2 = 0.5 \text{ and } b_1 = b_2 = 0.5)\) is selected for both the servers. Effect of varying data arrival rate is analyzed in terms of overflow probability and expected idle time period. Results obtained with Matlab simulation are as in Table 4.1.

<table>
<thead>
<tr>
<th>Arrival rate</th>
<th>Idle period</th>
<th>Overflow Prob.</th>
<th>Queue length</th>
<th>Lifetime (units)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\lambda)</td>
<td>(I_1)</td>
<td>(I_2)</td>
<td>(OV_1)</td>
<td>(OV_2)</td>
</tr>
<tr>
<td>0.2</td>
<td>0.58</td>
<td>0.61</td>
<td>0.0015</td>
<td>0.0022</td>
</tr>
<tr>
<td>0.3</td>
<td>0.4010</td>
<td>0.4211</td>
<td>0.0125</td>
<td>0.0106</td>
</tr>
<tr>
<td>0.4</td>
<td>0.2527</td>
<td>0.2596</td>
<td>0.033</td>
<td>0.042</td>
</tr>
<tr>
<td>0.5</td>
<td>0.1306</td>
<td>0.1953</td>
<td>0.081</td>
<td>0.075</td>
</tr>
<tr>
<td>0.6</td>
<td>0.0609</td>
<td>0.1780</td>
<td>0.25</td>
<td>0.06</td>
</tr>
<tr>
<td>0.7</td>
<td>0.0421</td>
<td>0.1740</td>
<td>0.324</td>
<td>0.09</td>
</tr>
<tr>
<td>0.8</td>
<td>0.0232</td>
<td>0.1561</td>
<td>0.4040</td>
<td>0.1117</td>
</tr>
<tr>
<td>0.9</td>
<td>0.0127</td>
<td>0.1341</td>
<td>0.436</td>
<td>0.107</td>
</tr>
</tbody>
</table>

From the observation table 4.1, we can divide the traffic in two categories—normal (\(\lambda\) less than 0.5) and catastrophe (\(\lambda\) is 0.5 and above).

**During normal data arrival:**

- Sensor node remains idle over large period (25% to 58% of ON time). Obviously idle power wastage is more.
• Overflow probabilities ($OV_1$ and $OV_2$) remains well within tolerance limit (below 10%).

• Average queue length below 2 (one third of maximum buffer capacity of 6).

During catastrophic data arrival:

• Idle period minimized (6% and below).

• Overflow probability at input buffer ($OV_1$) increases much beyond the tolerance limit (above 25%) but $OV_2$ is within limit.

• Average queue length $q_1$ increases sharply.

From these findings we can think of reducing the service rate during normal period to reduce the power consumption and idle period power wastage. It will save the power and also maintains the QoS parameter (overflow probability). Similarly during catastrophic period, need to decrease data loss by increasing the service rate. It will consume more power but QoS will be maintained.

Table 4.2 shows the simulation results for a sensor node. First row shows the result for a fixed service rate sensor node as the service rates during normal and catastrophic period are same. Other observation rows show the results for a sensor node with different service rates in normal and catastrophic period. We have checked the performance with various service rates by keeping the normal arrival rate 0.2 and catastrophic arrival rate 0.8. Also we ensured to have the same catastrophic duration in all cases.

Figure 4.6 shows the overflow probabilities of input and output buffers ($OV_1$ and $OV_2$) under catastrophic conditions. During catastrophic period data arrival rate suddenly increases. If this increased data arrival is not handled with proper service rate then data loss occurs due to buffer overflow.
Table 4.2: Performance parameters observed by simulation

<table>
<thead>
<tr>
<th>Normal</th>
<th>Catastrophic</th>
<th>Idle period</th>
<th>Normal</th>
<th>Catastrophic</th>
<th>Normal</th>
<th>Catastrophic</th>
<th>Lifetime</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1$</td>
<td>$b_1$</td>
<td>$f_2$</td>
<td>$b_2$</td>
<td>$I_1$</td>
<td>$I_2$</td>
<td>$OV_1$</td>
<td>$OV_2$</td>
</tr>
<tr>
<td>0.55</td>
<td>0.45</td>
<td>0.55</td>
<td>0.45</td>
<td>0.64</td>
<td>0.58</td>
<td>0.002</td>
<td>0.002</td>
</tr>
<tr>
<td>0.3</td>
<td>0.3</td>
<td>0.8</td>
<td>0.8</td>
<td>0.37</td>
<td>0.40</td>
<td>0.03</td>
<td>0.02</td>
</tr>
<tr>
<td>0.3</td>
<td>0.3</td>
<td>0.9</td>
<td>0.9</td>
<td>0.36</td>
<td>0.40</td>
<td>0.02</td>
<td>0.01</td>
</tr>
<tr>
<td>0.3</td>
<td>0.3</td>
<td>1</td>
<td>0.8</td>
<td>0.36</td>
<td>0.40</td>
<td>0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>0.35</td>
<td>0.35</td>
<td>0.85</td>
<td>0.85</td>
<td>0.45</td>
<td>0.47</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>0.4</td>
<td>0.4</td>
<td>0.9</td>
<td>0.9</td>
<td>0.48</td>
<td>0.50</td>
<td>0.01</td>
<td>0.04</td>
</tr>
<tr>
<td>0.4</td>
<td>0.4</td>
<td>1</td>
<td>0.9</td>
<td>0.48</td>
<td>0.50</td>
<td>0.01</td>
<td>0.00</td>
</tr>
</tbody>
</table>
A sensor node with fixed service rate do not have any control knob and hence it is not possible to control buffer overflow. A sensor node with only DVFS facility implemented on it can control the overflow probability of input buffer by increasing the clock frequency of the first server (processor). Increased service rate of the first server increases the data arrival rate in the output buffer but since second server works with fixed service rate overflow probability of the output queue ($OV_2$) increases. This situation is highly undesirable as the processed data gets lost, power used for processing that data also goes waste, which a highly power constrained sensor node can not afford. It is not advisable to have only DVFS implemented on sensor node. Similarly when only DMS is implemented, it results in decreasing the output buffer overflow probability at the cost of increased power but as the first server works with fixed service rate, input buffer overflow can not be controlled. Integrating both DVFS and DMS on a sensor node results in controlling the overflow probabilities of both the buffers. Though more power is consumed by DVFS and DMS during catastrophe by working with higher service rates data loss due to buffer overflow is reduced and can be kept within the tolerance limit which is the highest priority QoS parameter during catastrophe.
In Figure 4.7 overflow probabilities of input and output buffers are compared during normal period and in Figure 4.8 average idle time probabilities of both the servers are shown. As the data arrival rate is very small during normal periods, buffer overflow possibilities are negligible but possibilities of servers remaining idle are more. More the idle period, more is the power wastage. So for power constrained wireless sensor nodes power saving becomes highest priority QoS parameter during normal periods by reducing idle period. It is achieved by reducing the service rates of the servers. Fixed service rate sensor nodes are designed to handle worst case conditions and hence their service rates are set quite high. During
normal periods these servers remain idle most of the time and large amount of power is wasted but there is negligible chance of buffer overflow. When only DVFS is implemented then first server is able to reduce its service rate (processing speed) so that it consumes less power and also reduces idle period by working over a longer period. It slightly increases the overflow probability of the input buffer but which is fairly within the tolerance limit. Similarly having only DMS will reduce the idle time and power consumption of second server with little increase in the overflow probability, which is acceptable. Having both DVFS and DMS reduces the power consumption of both the servers also reduces their idle periods and hence results in power saving. This saved power can used during catastrophe periods to reduce data loss due to buffer overflows. As compared to fixed service rate sensor node lifetime increase of 15% was seen when only DVFS was implemented on a sensor node while implementing only DMS it was 17.5% but DVFS and DMS together applied on a sensor node resulted in 27.22%.

Figures 4.9 and 4.10 shows the snapshots of the screen showing MATLAB simulation graphs under normal and catastrophic periods.

4.3 Threshold determination

As few discrete service rates are supported by the servers, it becomes important to select a particular service rate by comparing the data in the buffer with the threshold. For convenience we consider a node which has two ON states- $ON_1$ and $ON_2$. In $ON_1$ state service rate is higher and the power consumed is more while in $ON_2$ state service rate is lower as well as power consumed is less but more service time is required. For switching the state between $ON_1$ and $ON_2$ a threshold is set and comparing the number of packets in the buffer with set threshold a decision is made whether to serve packets with $ON_1$ or
Figure 4.9: Comparison between fixed service rate and variable service rate sensor node in normal time period

Figure 4.10: Comparison between fixed service rate and variable service rate sensor node in catastrophe period
Selection of buffer threshold affects the performance of sensor node with finite buffer size. Here performance metric used is lifetime of sensor node and data loss due to buffer overflow. Various threshold policies have been studied in the literature. Threshold policy is used to choose the better one between two or more possibilities. Threshold policies are used for selecting a particular data rate at wireless switch or router in order to increase the throughput [150] [151]. Adaptive threshold policy has been used in [152] but the aim of using adaptive threshold policy is to change the transmission rate with time variable channel conditions. Threshold policies are also used for memory sharing between various sources as per their priority. In this paper we consider buffer threshold policy which enables the sensor node to vary its service rate w.r.t. buffer occupancy so that data loss due to buffer overflow can be minimized along with power optimization resulting in the increased lifetime of the node.

4.3.1 Threshold policies

Various threshold policies used for switching between two states are as below:

1. **Single threshold policy**: For a finite sized buffer, a threshold value is set such that whenever the number of packets in the buffer is more than the threshold k, server switches to the high service rate consuming more power but reduces the possibility of data loss due to buffer overflow. On the other hand if the number of packets in the buffer are less than or equal to the threshold k, server continues in the low power state consuming less power and working with low service rate. It reduces the idle period and power consumption but increases the possibility of data loss due to buffer overflow.
2. **Dual threshold policy**: Single threshold policy is suitable when there is not much variation in the traffic arrival. For random traffic arrival we may need to switch the service rates in consecutive time slots. Switching the state consumes extra time and extra energy which may over weigh the power saved. It is better to have some margin before switching to another state. It helps in reducing switching energy and switching time overheads. In dual threshold policy two thresholds $k_1$ and $k_2$ have been defined such that $k_2 > k_1$. At the start of a new slot if number of packets in the buffer is more than $k_1$ and less than $k_2$ then server will continue with the service rate as in the previous slot. Service rate will change only when buffer occupancy is outside the margin.

3. **Adaptive threshold policy**: In this policy the threshold level is varied w.r.t. the value of buffer overflow in previous slot along with the buffer occupancy. Buffer overflow value is stored in a overflow register. If the buffer overflow is approaching near the tolerable value then the threshold is decreased so that server can enter in high service state and buffer overflow possibility is reduced. On the other hand if the buffer overflow in the previous slot is negligible then the threshold value can be raised so that server remains in low power state and power is saved.

Figure 4.11 elaborates various threshold policies along with the pseudo codes.

<table>
<thead>
<tr>
<th>Threshold</th>
<th>Avg. idle time</th>
<th>Normal</th>
<th>Catastrophe</th>
<th>Lifetime</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k_1$</td>
<td>$k_2$</td>
<td>$I_1$</td>
<td>$I_2$</td>
<td>$OV_1$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0.4267</td>
<td>0.5428</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0.43</td>
<td>0.47</td>
<td>0.0112</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0.42</td>
<td>0.44</td>
<td>0.03</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0.3789</td>
<td>0.4313</td>
<td>0.018</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>0.3692</td>
<td>0.3928</td>
<td>0.017</td>
</tr>
</tbody>
</table>
Table 4.3 lists the effect of varying threshold levels on the life time of sensor node as well as buffer overflow probability during normal period as well as during catastrophic period. It has been observed that for higher threshold levels lifetime of the sensor node is more than that for lower threshold levels. This increase in lifetime has been achieved at the cost of data loss during catastrophic periods which is not at all desirable. With higher threshold levels sensor node remains in low power state for a longer duration and saves the power. On the other hand with small threshold levels very frequently sensor node switches to high power state giving service with increased speed so more power is consumed but data loss gets reduced. selection of threshold levels is a trade off between lifetime and data loss.

In the Table 4.4, Fixed (3,3) indicates the threshold values for input and output buffer are 3 and 3 and remains fixed. Adaptive (3,3) indicates the starting threshold values are 3 and 3 but adaptively changes with buffer overflow and buffer occupancy.

Figure 4.12 shows comparison of various performance parameters of a sensor node using fixed threshold policy and adaptive threshold policy.
Table 4.4: Comparison of fixed threshold effect with adaptive threshold effect

<table>
<thead>
<tr>
<th>Threshold (k1,k2)</th>
<th>Avg.idle time</th>
<th>Normal</th>
<th>Catastrophe</th>
<th>Lifetime (Time units)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$I_1$</td>
<td>$I_2$</td>
<td>$OV_1$</td>
<td>$OV_2$</td>
</tr>
<tr>
<td>Fixed (3,3)</td>
<td>0.3906</td>
<td>0.4351</td>
<td>0.0186</td>
<td>0.0261</td>
</tr>
<tr>
<td>Adaptive (3,3)</td>
<td>0.3344</td>
<td>0.3470</td>
<td>0.0382</td>
<td>0.0257</td>
</tr>
<tr>
<td>Fixed (4,4)</td>
<td>0.3919</td>
<td>0.4041</td>
<td>0.0167</td>
<td>0.0159</td>
</tr>
<tr>
<td>Adaptive (4,4)</td>
<td>0.3350</td>
<td>0.3709</td>
<td>0.0276</td>
<td>0.0162</td>
</tr>
</tbody>
</table>

Figure 4.12: Comparison of a sensor node using fixed threshold policy and adaptive threshold policy

Simulation results show that the lifetime of sensor node can be increased by using adaptive threshold policy as compared to that with fixed threshold policy. It also reduces idle time period and power loss during it. Overflow probability during catastrophic periods can be reduced further by selecting next higher service rate along with adaptive threshold.

In the next chapter, we will discuss the bulk arrival Markov chain model of a sensor node with two service rates. There, sensor node will be a single entity having DVFS and DMS coordinated within it.