Chapter 5

A NOVEL APPROACH FOR INFORMATION PROCESSING AND MAINTAINING ROBUSTNESS OF WSN IN SUBSURFACES

5.1 Introduction

Literature presented in chapter 3 and works proposed in Chapter 4 indicates that innovative deployment of sensors in subsurfaces can beneficially support the production of oil and gas. In order to enhance the oil recovery and diagnosing the conditions prevailing in pipelines and underneath, this work was initiated with the deployment of nodes and further establishing an energy efficient communication among the deployed nodes. Looking at the technological challenges highlighted in previous chapters, this chapter concentrates on the last two phases of this thesis namely, Information Processing i.e. processing and filtering large amounts of raw data for human visualization and Maintaining Robustness i.e. maintaining the topology as well as connectivity of deployed sensors.

The upcoming section describes A Novel Application of Extended Kalman Filter For Efficient Information Processing in Subsurfaces [51]. The work was taken into consideration because the data, which is sensed by such sensors are usually corrupted with noise. Filtering is desirable in such embedded systems in order to smooth out such fluctuations that otherwise would shorten the lifespan of sensors. This contribution presents a unique application of Kalman filtering technique for processing such sensitive information because sensor readings are usually imprecise due to strong variations in environment and also, computation has to be much more energy efficient than communication. Out of the various filtering algorithms available, we have chosen to apply Kalman filter, primarily because it works well both in theory
and practice and moreover, it is able to minimize the variance of estimation error i.e. filters noise from the actual signal more accurately.

5.2 Information Processing Phase

In the past few years, innovative oil monitoring techniques deployed in oil wells or sub-surfaces have proved to be an efficient way of measuring various parameters such as pressure, temperature, volume of oil available at various locations. Such innovative systems comprise of thousands of heterogeneous sensors, which continually sense the environment. Since, the sensors are constrained in terms of energy, therefore, in order to be efficiently benefited from such systems, information that is actually promising shall only be routed so as to avoid useless energy drain and hence shorten the lifespan of WSN. In order to meet, the above stated objective, one must employ the technique of collaborative signal processing and filter the unnecessary noise locally.

Collaborative signal processing leads to information fusion, which in turn reduces the data volume to be routed. However, the authors feel that this fused information must be processed i.e. filtered before routing because sensors readings are usually imprecise due to strong variations in the environment and hence information is corrupted with noise. In order to filter the sensed data, tools known as Kalman filters can be used as these are often being used in embedded control systems for estimating the accurate range of process variables.

Next section provides a broad overview of Kalman Filter and the application of the same has been given in subsequent section.
5.2.1 Kalman Filters: An Overview

The Kalman filter [129] is a mathematical model which describes equations to estimate the state of a process in a recursive computational manner. The filtering algorithm is highly efficient for minimizing the mean of squared error, estimations of past, present and future states. The purpose of this section is to provide an insight to implementation of Kalman Filtering for Information Processing within Wireless Sensor Networks deployed in Sub-Surface.

In order to apply Kalman filter, the process to be measured, shall be such that it can be described as linear system. A linear system can be expressed by a difference equation and a measurement equation as given by eq.5.1 and eq.5.2 respectively.

\[ x_{k+1} = Ax_k + Bu_k + w_k \]  
\[ z_k = Hx_{k+1} + v_k \]

In the above equations,

- \( x \): the state of the system,
- \( A \): \( n \times n \) matrix relates the state at the previous time step \( k - 1 \) to the state at the current step \( k \), in the absence of either a driving function or process noise. Note that in practice, \( A \), might change with each time step, but here we assume it is constant.
- \( B \): \( n \times l \) matrix relates the optional control input \( u \) to the state \( x \).
- \( H \): \( m \times n \) matrix in the measurement equation relates the state to the measurement \( z_k \). In practice, \( H \) might change with
each time step or measurement, but here we assume it is constant.

\( k \) : time index;
\( u \) : a known input to the system;
\( z \) : measured output; and
\( w \) : process noise, and
\( v \) : measurement noise.

Given these two equations, we measure \( z \), which is a function of \( x \) that is corrupted by the noise \( v \) as we cannot measure \( x \) directly. It shall be noted that \( z \) can only be used to obtain an estimate of \( x \), however, the information in \( z \) is also corrupted by noise. Also, the above two equations indicates that Kalman Filter is recursive in nature i.e. it primarily estimates the state of the process at some arbitrary time and the feedback in the form of noise is observed.

It is evident that using Kalman filter, one is able to estimate the state of a discrete-time controlled process, which is governed by a linear stochastic difference equation. However, an interesting application of the mentioned filter would be in case of non-linear processes i.e. the measurement relationship is non-linear. This is where Extended Kalman Filter (EKF) comes in. Let us assume that the process under consideration is a non-linear process and for a state vector \( x \), the process is now governed by the two non-linear stochastic difference equations given in eq.5.3 and eq.5.4 known as time-update and measurement-update equations respectively.

\[
\hat{x}_{k+1} = f(\hat{x}_k, u_k, w_k) \quad \text{...............(eq.5.3)}
\]
\[
\hat{z}_{k+1} = h(\hat{x}_{k+1}, v_k) \quad \text{...............(eq.5.4)}
\]

where, the random variables \( w \) and \( v \) again represent the process and measurement noise. \( f \) and \( h \) are two non-linear functions where former relates the state at the
previous time step to the state at the current time step and later relates the state $x_k$ to the measurement $z_k$. Here, matrices $A, B$ and $H$ (as mentioned in eq.5.1) are partial derivatives of functions $f$ and $q$ respectively. In practice, of course one does not know the individual values of the noise $w_k$ and $v_k$ at each time step. However, one can approximate the state and measurement vector without them as given in eq.5.5 and eq.5.6 respectively.

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k, 0) \quad \text{.........(eq.5.5)}$$
$$\hat{z}_{k+1} = q(\hat{x}_{k+1}, 0) \quad \text{.........(eq.5.6)}$$

where, $\hat{x}_{k+1}$ is some a posteriori estimate of the state (from a previous time step $k$).

Since, measurement relationship of parameters such as pressure, temperature, volume of oil in a particular vicinity etc., in oil well is a non-linear, this work aims to apply an EKF that linearize the estimation around the current estimate using the partial derivatives of the process and measurement functions to compute estimates even in the face of non-linear relationships.

5.2.2 The Proposed Work

The section focuses upon the applying EKF for Information Processing in WSN in sub-surfaces and in non-deterministic environments, in general. Our previous works [54] proposed how innovative deployment of sensors can beneficially support the production of oil and gas. Further, a query driven routing protocol [53] also addressed the issue pertaining to routing of information in such non-deterministic environments. One of the primary assumptions in these works are heterogeneous nodes i.e. to measure the modalities present within subsurface, various kind of sensor nodes such as temperature sensor, pressure nodes, acoustics nodes, flow nodes etc. are taken into consideration.
For the efficient production of heavy crude, the requirement is the continuous monitoring of all pertinent parameters throughout the production process. The parameters, which are routinely measured are: annulus pressure, well head pressure, temperature, Oil Volume and flow rate. However, to the author's knowledge, suitable calculation methods or formulas have not been published so far except for a few empirical formulae, which are suitable only over a limited range.

Attaching sensors to the oil is able to provide continuous liquid production rates but also allows to detect hazardous substances such as H$_2$S or to connect sensors for monitoring of critical temperatures such as Boiling point temperatures. In order to avoid complexity of calculations, we limit our scope to process pressure and Oil volume only, rather than processing all stated parameters. Assuming that in standard conditions, pressure ($p$) at any depth ($h$) at time $k$ can be computed (by using Pressure Sensor) as given by eq.5.7

$$p_k = h_k \rho g$$

where, $h =$Depth of area,

$\rho =$ density of oil, constant kg/m$^3$

$g =$ acceleration due to gravity, 9.8m/sec$^2$

But the previous equation, does not give a precise value for $p_k$. Instead, the pressure is perturbed by noise due to changing depth and other unfortunate realities. The pressure noise is a random variable that changes with position/depth. So a more realistic equation for $p$ would be as given by eq.5.8:

$$p_k = h_k \rho g + u_k$$
where, \( u_k^- = h_k \rho g^- \) is the noise in pressure depending upon the value of \( h \) which may also be corrupted. Similarly, Oil Volumes (OV) in standard conditions is computed in accordance to eq.5.9.

\[
V_{sc} = V_s \times SCF \times (1 - SHR) \times (1 - BSW) \times T
eq (eq.5.9)
\]

Where, \( V_{sc} \) = Oil Volume at standard conditions
\( V_s \) = Oil volumes measured by sensors critical
\( SCF \) = Sensor Correction factor = True volume/ Measured volume
\( SHF \) = Shrinkage Factor
\( BSW \) = Water and Sediment % measured
\( T \) = Temperature volume correction factor at standard conditions.

Here, the temperature used to calculate the volume correction factor to standard conditions is the well temperature measured using standard temperature sensors.

As per eq.5.8, a similar expression for oil volume can be given as in eq.5.10.

\[
V_{k+1} = V_{sc} + w_k^- \ hend (eq.5.10)
\]

where, \( w_k^- = V_{sc}^- \) is the noise in the volume of oil measured. Now, we can define a state vector \( x \) (given by eq.5.11) that consists of pressure and volume of oil as

\[
x_{k+1} = f(p_{k+1}, V_{k+1}, h_{k+1}, u_k^-, w_k^-) \ hend (eq.5.11)
\]

and output vector \( z \) is given by eq.5.12.

\[
z_{k+1} = f(x_{k+1}, m_k^-) \ hend (eq.5.12)
\]
In order to estimate the Pressure $p$ and Volume $V$, we need to estimate $x$ and $z$, hence the need of EKF. Clearly, the requirement is twofold: First, the estimated value should be same as expected value of state and second, measurement update should be accurate i.e. $z$ should be estimated with smallest possible error. Here, in eq.5.11 and eq.5.12, $u_k^\sim, w_k^\sim$ are the process noises and $m_k^\sim$ is the measurement noise respectively.

Now, in order to have the true estimates of the desired parameters, we have to assume that the average value of the stated noise variables should be zero. Further, it is evident that $u_k^\sim, w_k^\sim$ and $m_k^\sim$ are independent random variables. Then the Pressure Noise Covariance and Oil Volume Noise Covariance matrices are defined as in eq.5.13 and eq.5.14 respectively.

$$S_p = E(u_k^\sim u_k^{\sim T})$$ 

…….(eq.5.13)

$$S_v = E(w_k^\sim w_k^{\sim T})$$ 

…….(eq.5.14)

Similarly, Measurement noise covariance is computed as given in eq.5.15:

$$S_m = E(m_k^\sim m_k^{\sim T})$$ 

…….(eq.5.15)

where, $u_k^{\sim T}$, $w_k^{\sim T}$ and $m_k^{\sim T}$ are the transpose of respective noise parameters and $E$ is the expected value. On the basis of these arguments, equations for EKF can be formulated as follows:

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k, 0)$$ 

…………(eq.5.16)

$$P_{k+1} = A_{k+1}P_kA_{k+1}^T + W_{k+1}S_p W_{k+1}^T + W_{k+1}S_v W_{k+1}^T$$ 

…………(eq.5.17)

$$K_{k+1} = P_{k+1}H_{k+1}^T (H_{k+1}P_{k+1}H_{k+1}^T + S_m)^{-1}$$ 

…………(eq.5.18)

$$\hat{z}_{k+1} = K_{k+1}(z_{k+1} - q(\hat{x}_{k+1}, 0))$$ 

…………(eq.5.19)

$$P_{k+1} = (1 - K_k H_k)P_k$$ 

…………(eq.5.20)
Eq.5.16 and eq.5.17 represent time update equations while eq.5.18 to eq.5.20 are measurement update equations. The time update equations compute the state and covariance estimates. $A$ and $W$ are the process Jacobian matrices and are computed as partial derivatives of function $f$. The state and covariance estimates are corrected with measurement $z_{k+1}$. $K$ is the Kalman gain and $H$ is the measurement Jacobian at step $k+1$. It shall be noted that Jacobian $H$ magnifies only the relevant component of the measured parameter. It further, demands that there one to one mapping between the state and the measurement at some of the instances, else the process is treated as unobservable and filter diverges. It is intuitive from eq.5.18 that if measurement noise i.e. $S_m$ is large, $K$ will be small and the measurement $z_{k+1}$ would not be of much importance in computing $\hat{x}_{k+1}$, the next state and vice-versa. Similar arguments may be applied to compute other pertinent parameters. Next subsection presents the implementation of the above work for estimating pressure and associated noise. Representing the computation of all parameters is out of scope of this thesis.

### 5.2.3 Implementation and Result

Let us assume that pressure measurement in an oil well system is done by measurement of the depth $h$ at which the probe reaches at a time $t$ and hence,

$$p(t) = \rho gh(t) \quad \ldots \ldots (eq.5.21)$$

If the probe starts from rest with a constant acceleration $a (=2 \text{ m/s}^2)$, the actual depth and probe velocity at $(k+1)$-th time step maybe expressed as:

$$h_{k+1} = h_k + u \Delta t + \frac{1}{2} a(\Delta t)^2 + z_h \quad \ldots \ldots (eq.5.22)$$

$$u_{k+1} = h_k + a \Delta t + z_u \quad \ldots \ldots (eq.5.23)$$
Where $z_h$ and $z_u$ are the process noises associated with depth and velocity respectively. The measured pressure is evaluated as:

$$p_{k+1} = \rho g (h_{k+1} + v)$$

.........(eq.5.24)

Where $v$ (=5 m, one standard deviation) is the noise associated with the measurement of depth, $\rho =800$ kg/m3 and $g=10$ m/s2. Assuming the acceleration noise to be 0.1 m/s2, the Kalman algorithm is executed in order to filter the noise from the measured pressure signal and obtain the estimated pressure signal as shown in figure 5.1. The errors in the measured and estimated pressures are shown in figure 5.2. It is clear that Kalman filter greatly reduces the measurement noise. Similar, simulations can be performed for any other parameter observed by sensor nodes.
5.3 Maintaining Robustness

Next section discusses the last phase i.e. maintain robustness of the proposed model i.e. A Novel Node Replenishment Strategy for Robustness within Wireless Sensor Nodes in Subsurface [52] is being proposed which adds strength to the proposed structure. The proposed strategy while collecting data from sensor nodes; also acquire meta-information regarding their geo-location, which is be used to determine the correctness of deployed network topology. The strategy is able to describe node replenishment policy in case the meta-information gathered reflect some deviation. The strategy is a unique contribution to subsurface exploration technology as none has proposed replenishment and hence robustness strategy for this field, in particular.

Figure 5.2: Errors in the Measured and Estimated Pressures

Now, WSN being used for critical operations like battlefield surveillance, intrusion detection, or detection and tracking of chemical, biological, radiological, nuclear, explosive agents and subsurfaces in particular, require high degree of reliability mechanism in order to significantly reduce end-to-end packet loss ratio. Drifting and failing nodes due to depletion of battery power or other reasons raise a
significant number of routing issues, demanding the design and implementation of efficient routing protocols. In addition to re-routing data packets in case of nodes failure, researchers have worked on replenishment or replacement of sensor nodes too. These strategies allow the network topology to be re-instated to its original form, and hence making way to a smooth communication. Most current replacement strategies have been laid down for traditional applications on earth surface or even underwater. Therefore there exists a need for networks geared towards subsurface to have an efficient replacement protocol, custom defined for them. So, the current section focuses on developing a replacement strategy which is able to achieve the stated objectives.

5.3.1 The Proposed Node Replacement Protocol

As already discussed and presented in previous chapter, the sensors are being deployed in the form of r-strips. Now, this algorithm executes when either a deviation from the underlined topology is observed or inter-node communication fails. The underlying topology and assumptions have been adopted as it is while proposing the algorithm for replacement of nodes which might have drifted away from their initial deployed position or might have failed (see Figure 5.3).

Figure 5.3: Replacement of Node
The orange node here represents a drifted away node thereby breaking the connectivity. It is important to note that GCS have been programmed so as to sense the location of nodes while collecting data from them. The following two issues may arise:

**i) Nodes drifting away from their originally deployed co-ordinates**

As per the mathematical model mentioned in [18], the standard distance between two communicating nodes is computed as per eq. 4.1 & eq. 4.2. It is evident that GCS, if it is not able to sense within the range specified by the above coordinates, interprets it to be a crisis situation and assumes that the node has drifted away from its location. Though in some cases (scenario where the node displaces from original location and drifts towards the GCS), it still might be able to communicate with GCS.

**ii) Nodes failure**

While performing forward data collection, GCS finds that

\[
\text{Coordinate}_x(\text{current node}) - \text{Coordinate}_x(\text{previous node}) \\
= \begin{cases} 
\pm 2i r & \text{if } j \text{ is odd} \\
\pm (2i - 1) r & \text{if } j \text{ is even}
\end{cases} \quad \ldots \text{(eq.5.25)}
\]

It is implied that a sensor node has failed over in between within current r-strip, since the distance observed in (eq.5.25) is equal to twice as that in (eq. 4.1). It should be noted that for energy efficiency the GCS do not compare \{coordinate \_y\}, since this value, as shown in (eq 4.2) remains fixed for a given r-strip, and is not dependent on variable \(i\).
For both the cases (i) and (ii) stated above, the void which has been generated due to the perpetrator sensor node needs to be replenished. This process of node replenishment will be initiated by GCS upon reaching the r-strip node nearest to the base station.

As depicted in figure 5.3, void created due to node (2, 3) needs to be replenished. The node replacement flow shall take place in the following order:

i) Corresponding GCS (3, 2) moves along its travel path to the node nearest to the base station. Upon reaching the last point it replaces the node (0, 3) in r-strip below it.

ii) node (0, 3) receives the request from GCS and moves forward to replace node (1, 3)

iii) node (1,3) moves forward to replenish the space (void) which had been created by failed / drifted node (2, 3)

iv) node (0, 2) replaces the position vacated by GCS (3, 2)

v) GCS (2, 1) replaces the position vacated by node (0, 2)

vi) node (0, 1) replaces the position vacated by GCS (2, 1)

vii) A new sensor node is deployed near base station to replenish the position vacated by node (0, 1)

In general the node replacements is depicted as

\[
\text{node}(j-1, n) \sim \text{replaces} \sim \text{node}(j, n)
\]

\( \forall j \in (1, 2, \ldots \text{coordinate of drifted node}) \) for an r-strip requiring node replacement
GCS (n, n-1) → node (0, n)  
node (0, n) → GCS (n+1, n)  
for all r-strips

A) Working Algorithm

Input required for the algorithm:

- Value of n, which is the number of r-strips deployed in subsurface
- Value of a, which is offset of farthest node from base station in any given r-strip.
- Time $k_i = d_i + t_i$ is the time for collector to travel from $i$th sensor to the $(i+1)$th sensor (modulo n plus time required collect location data from the $i$th head ($t_i > 0$).

1: INITIALISE
2: set $q=a$ and $w=n$
3: set time=0
4: Method: collect_loc_data ($q$, $w$) {start}
5: GCS ($w$, $w-1$) enquires location data from node ($q$, $w$)
6: Increment time step by $k_i$
7: GCS ($w$, $w-1$) enquires location data from node ($q-1$, $w$)
8: Calculate $current_{offset} = data_{location} \{node (q-1, w)\} - data_{location} \{node (q, w)\}$
9: if $[w \mod 2] = 1$
10: if $current_{offset} \neq \pm ir$
11: set $node_{error} = node (q-1, w-1)$
12: else
if \( \text{current}_{\text{offset}} \neq \pm \frac{(2i-1)r}{2} \)

set \( node_{\text{error}} = \text{node (q-1, w-1)} \)

15: GCS (w, w-1) arrives at node (0, n)

16: Initialize process of node replacement:

17: Loop 1:

18: Replace GCS (w, w-1) with node (0, n)

19: Replace node (j-1, n) with node(j, n) \( \forall j \in \{1, 2, \ldots \} \) coordinate of drifted node)

20: Loop 2:

21: Replace GCS (n, n-1) with node (0, n)

22: Replace node (0, n) with GCS (n+1, n)

23: Process of node replacement completes

24: Stop

### 5.4 Conclusions

The chapter proposed two important phases out of four phases carried out during this thesis. A new dimension of using EKF has been proposed and implemented as this technique is suitable for critical measurements which are not only accurate but also filtered leading to better diagnosis, monitoring of oil wells and a more efficient use of sensor energy. The electronically acquired data can be distributed to the scientists for better analysis and it also gives an opportunity to view the data in real time. On the other side, some sensors are prone to drift and hence a novel protocol for node replenishment within WSN deployed in subsurface exploration is also being proposed. GCS’s which travel parallel to r-strips accumulate location data on their way, and execute upon this information to replenish the void created due to a failed node or drifted away node thus adding robustness to the proposed system.