AN ABSTRACT OF THE THESIS OF

Juan Pablo Muñoz Constantine for the degree of Master of Science in Electrical and Computer Engineering presented on September 22\textsuperscript{nd}, 2015.

Title: Grid Voltage Frequency Estimation Using an Adaptive Complex Master-Slave Unscented Kalman Filter.

Abstract approved:

Mario E. Magaña

Due to the popularity of electronically controlled loads and the widespread use of alternative energy sources such as wind turbines and solar cells, the power quality at the distribution level must be carefully monitored. One of the many different means to monitor power quality is through frequency measurements. Nonlinearities cause harmonic generation effectively distorting the signals seen at the load, and so this seemingly simple task becomes a challenge. Recent advances in digital computing have allowed powerful linear estimators like the Kalman filter (KF) to be widely implemented for frequency estimation. In real physical applications, systems exhibit nonlinear behavior raising the need to adapt the Kalman Filter to fit nonlinear models. The Extended Kalman Filter (EKF) deals with nonlinearities in the system by linearizing the model around a known state. However, this makes the estimation process inaccurate because second order or higher nonlinearities are neglected.

The recently developed Unscented Kalman filter (UKF) takes advantage of the Unscented Transformation (UT) to deal with nonlinear models without the need of linearization and with the same computational complexity as the EKF. However, any variations of the Kalman filter exhibit a very similar robustness problem when
modeling uncertainty and are also sensitive to initial conditions. In order to overcome these limitations, an improved UKF algorithm based on the theory of strong tracking filters (STF) has been developed. If tuned properly, this new algorithm improves tracking for sudden changes and avoids divergence. However, if the process and/or measurement noise change with time, the filter will not estimate with the same accuracy it was tuned for. This thesis work proposes an adaptive algorithm based on an Unscented Kalman Filter (UKF), in order to improve the frequency estimation of power signals which undergo changes and are corrupted by white noise. The adaptive algorithm is based on a Master-Slave configuration, where the “master” estimates the state and the “slave”, which operates in parallel, estimates the noise covariance matrix. Since the voltage signal is less distorted than the current signal, the former is employed to derive a complex state-space model to estimate the fundamental frequency. In order to evaluate the performance of the proposed algorithm, several simulations with synthetic data are implemented.
Grid Voltage Frequency Estimation Using an Adaptive Complex Master-Slave Unscented Kalman Filter

by
Juan Pablo Muñoz Constantine

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented September 22\textsuperscript{nd}, 2015
Commencement June 2016
Master of Science thesis of Juan Pablo Muñoz Constantine presented on September 22nd, 2015

APPROVED:

Major Professor, representing Electrical and Computer Engineering

Director of the School of Electrical Engineering and Computer Science

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Juan Pablo Muñoz Constantine, Author
ACKNOWLEDGEMENTS

To begin, I would like to express my most sincere gratitude to my advisor, Dr. Magaña. I probably would not be writing this dissertation if it were not for the opportunity he gave me to join his research group. I had many doubts regarding my future as a graduate student. However, his welcoming and humble attitude quickly reassured me this was something I wanted and needed to pursue. Your vast experience, inside and outside the classroom, has helped me become a much better person all around. Whenever I came to your office with academic questions, and heard that latin music playing in background, I always left with something more valuable than just knowledge. Your patience, guidance, and work ethic have been determining factors to my success. I consider myself very fortunate to have had you as an advisor.

I wish to thank officemates, friends, and colleagues for helpful discussions, insights, and your valuable friendship. I have reflected upon my two years at Oregon State and realized that my time here has been a great one thanks to all of you. I would also like to thank very good friends and mentors Dr. John Natzke and Dr. Ed Godshalk for their encouragement to pursue higher education. Your raw passion for engineering was a huge motivator for me during rough times. My experience as a graduate student would not have been nearly as enjoyable if it had not shared so much of it with my good friend Jose Picado. I will never understand why we spoke in English when we were both native Spanish speakers. Thank you for helping me stay grounded, focused, and most importantly, for encouraging me to enjoy being a graduate student.

Lastly, and foremost, I wish to thank my parents, brothers, and sister for their unconditional support. There are truly no words to thank family for their encouragement. I will be forever in my parents’ debt for their full commitment to my education and the work ethic they instilled in me. This dissertation is as much theirs as it is mine. I was always encouraged to believe I can push my own limits and achieve all my goals. For that and the many more things I am and will be eternally grateful.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1.1 Background</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1.2 Overview of Frequency Estimation in Power Systems</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Modeling</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>2.1 Signal Modeling</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>2.1.1 Single Sinusoid Voltage</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>2.1.2 Three-Phase Voltages</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>2.2 Clarke Trasnformation</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>2.3 State Space Representation</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>The Kalman Filters</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>3.1 The Kalman Filter</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>3.1.1 The Computational Origins of the Filter</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>3.1.2 The KF Algorithm</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>3.2 The Extended Kalman Filter</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>3.2.1 EKF Algorithm</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>3.3 The Unscented Kalman Filter</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>3.3.1 The Unscented Transform</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>3.3.2 UKF Algorithm</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>3.4 Strong Tracking Filter Condition</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>3.4.1 UKF Based on STF Condition</td>
<td>29</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 Master-Slave Unscented Kalman Filter</td>
<td>32</td>
</tr>
<tr>
<td>4.1 Structure</td>
<td>32</td>
</tr>
<tr>
<td>4.2 MS-UKF Algorithm</td>
<td>34</td>
</tr>
<tr>
<td>5 Results</td>
<td>38</td>
</tr>
<tr>
<td>5.1 Filter Comparison I: EKF vs UKF vs MS-UKF</td>
<td>38</td>
</tr>
<tr>
<td>5.1.1 Frequency Step Variation</td>
<td>39</td>
</tr>
<tr>
<td>5.1.2 Linear Frequency Variation</td>
<td>42</td>
</tr>
<tr>
<td>5.1.3 Nonlinear Frequency Variation</td>
<td>44</td>
</tr>
<tr>
<td>5.2 Filter Comparison II: Dynamic Covariance (UKF vs MS-UKF)</td>
<td>46</td>
</tr>
<tr>
<td>5.2.1 Dynamic Covariance &amp; Constant Frequency</td>
<td>46</td>
</tr>
<tr>
<td>5.2.2 Dynamic Covariance &amp; Linear Frequency Variation</td>
<td>48</td>
</tr>
<tr>
<td>5.2.3 Dynamic Covariance &amp; Step Frequency Variations</td>
<td>49</td>
</tr>
<tr>
<td>5.3 Filter Comparison III: UKF vs UKF-STF</td>
<td>51</td>
</tr>
<tr>
<td>5.3.1 Two-Step Frequency Variation</td>
<td>52</td>
</tr>
<tr>
<td>5.3.2 Linear Frequency Variation (UKF vs UKF-STF)</td>
<td>52</td>
</tr>
<tr>
<td>5.3.3 Nonlinear Frequency Variation (UKF vs UKF-STF)</td>
<td>53</td>
</tr>
<tr>
<td>5.4 Filter Comparison IV: Dynamic Covariance (UKF-STF vs MS-UKF-STF)</td>
<td>54</td>
</tr>
<tr>
<td>5.4.1 Nonlinear Frequency Variation &amp; Dynamic Covariance</td>
<td>54</td>
</tr>
<tr>
<td>5.4.2 Linear Frequency Variation &amp; Dynamic Covariance</td>
<td>56</td>
</tr>
<tr>
<td>5.5 Filter Comparison V: MS-UKF-STF vs UKF-STF</td>
<td>58</td>
</tr>
<tr>
<td>5.5.1 Two-Step Frequency Variation (MS-UKF-STF vs UKF-STF)</td>
<td>59</td>
</tr>
</tbody>
</table>
TABLE OF CONTENTS (Continued)

5.5.2 Linear Frequency Variation (MS-UKF-STF vs UKF-STF) ..................60
5.5.3 Nonlinear Frequency Variation (MS-UKF-STF vs UKF-STF) ..........62

6 Conclusions .........................................................................................64
   6.1 The Benefits of an Adaptive Algorithm .........................................64
   6.2 Future Work ..................................................................................65

Bibliography .........................................................................................66
**LIST OF FIGURES**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Block diagram of power generation, transmission, and distribution</td>
<td>2</td>
</tr>
<tr>
<td>2.1</td>
<td>Graphical representation of Clarke transform</td>
<td>10</td>
</tr>
<tr>
<td>3.1</td>
<td>Kalman filter recursive algorithm</td>
<td>17</td>
</tr>
<tr>
<td>3.2</td>
<td>Unscented transformation</td>
<td>21</td>
</tr>
<tr>
<td>3.3</td>
<td>Unscented and extended Kalman filter approaches</td>
<td>24</td>
</tr>
<tr>
<td>3.4</td>
<td>Performance of UKF algorithm for $\beta$ different values</td>
<td>30</td>
</tr>
<tr>
<td>3.5</td>
<td>Sensitivity to initial conditions</td>
<td>31</td>
</tr>
<tr>
<td>4.1</td>
<td>MS-UKF structure</td>
<td>33</td>
</tr>
<tr>
<td>5.1</td>
<td>Frequency step-up at $t = 0.25$ s</td>
<td>39</td>
</tr>
<tr>
<td>5.2</td>
<td>Reconstructed signal with frequency step-up at $t = 0.25$ s</td>
<td>40</td>
</tr>
<tr>
<td>5.3</td>
<td>Frequency step-down at $t = 0.25$ s</td>
<td>41</td>
</tr>
<tr>
<td>5.4</td>
<td>Reconstructed signal with frequency step-down at $t = 0.25$ s</td>
<td>41</td>
</tr>
<tr>
<td>5.5</td>
<td>Linear frequency variation (EKF, UKF, MS-UKF)</td>
<td>42</td>
</tr>
<tr>
<td>5.6</td>
<td>Reconstructed signal with linear frequency variation I</td>
<td>43</td>
</tr>
<tr>
<td>5.7</td>
<td>Nonlinear frequency variation (EKF, UKF, MS-UKF)</td>
<td>45</td>
</tr>
<tr>
<td>5.8</td>
<td>Reconstructed signal with nonlinear frequency variation I</td>
<td>45</td>
</tr>
<tr>
<td>5.9</td>
<td>UKF algorithm with SNR variation at $t = 0.5$ s</td>
<td>47</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>5.10</td>
<td>MS-UKF algorithm with SNR variation at $t = 0.5$ s</td>
<td>47</td>
</tr>
<tr>
<td>5.11</td>
<td>Linear frequency variation with SNR step at $t = 0.5$ s (UKF)</td>
<td>49</td>
</tr>
<tr>
<td>5.12</td>
<td>Linear frequency variation with SNR step at $t = 0.5$ s (MS-UKF)</td>
<td>49</td>
</tr>
<tr>
<td>5.13</td>
<td>Frequency step variations with SNR change at $t = 0.5$ s</td>
<td>50</td>
</tr>
<tr>
<td>5.14</td>
<td>Double frequency step variation (UKF, UKF-STF)</td>
<td>52</td>
</tr>
<tr>
<td>5.15</td>
<td>Linear frequency variation (UKF, UKF-STF)</td>
<td>53</td>
</tr>
<tr>
<td>5.16</td>
<td>Nonlinear frequency variation (UKF, UKF-STF)</td>
<td>54</td>
</tr>
<tr>
<td>5.17</td>
<td>Nonlinear frequency variation with SNR step (UKF-STF)</td>
<td>55</td>
</tr>
<tr>
<td>5.18</td>
<td>Nonlinear frequency variation with SNR step at $t = 0.5$ s</td>
<td>56</td>
</tr>
<tr>
<td>5.19</td>
<td>Linear frequency variation with SNR step at $t = 0.5$ s (UKF-STF)</td>
<td>57</td>
</tr>
<tr>
<td>5.20</td>
<td>Linear frequency variation with SNR step at $t = 0.5$ s (MS-UKF-STF)</td>
<td>57</td>
</tr>
<tr>
<td>5.21</td>
<td>Double frequency step variation (MS-UKF-STF, UKF-STF)</td>
<td>59</td>
</tr>
<tr>
<td>5.22</td>
<td>Linear frequency variation (MS-UKF-STF, UKF-STF)</td>
<td>61</td>
</tr>
<tr>
<td>5.23</td>
<td>Nonlinear frequency variation (MS-UKF-STF, UKF-STF)</td>
<td>62</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>MSE Of frequency step variation over 100 independent runs</td>
<td>40</td>
</tr>
<tr>
<td>5.2</td>
<td>MSE of linear frequency variation over 100 independent runs</td>
<td>44</td>
</tr>
<tr>
<td>5.3</td>
<td>MSE of nonlinear frequency variation over 100 independent runs</td>
<td>46</td>
</tr>
<tr>
<td>5.4</td>
<td>MSE of double frequency step over 100 independent runs</td>
<td>60</td>
</tr>
<tr>
<td>5.5</td>
<td>MSE of linear frequency variation over 100 independent runs</td>
<td>61</td>
</tr>
<tr>
<td>5.6</td>
<td>MSE of nonlinear frequency variation over 100 independent runs</td>
<td>63</td>
</tr>
</tbody>
</table>
Chapter 1: Introduction

1.1 Background

The theory and early development of AC power generation dates back to the mid and late 1800’s. It was not until the 1900’s, however, that three phase AC power was fully established as the primary source of power to the world. Since then the world has seen an immense development of complex interconnected networks that generate and distribute power.

Whether the power generated comes from a nuclear plant or a wind turbine, it must be synchronized to a desired frequency (60 Hz in the US, 50 Hz in Europe) and up converted by a step-up transformer for electric transmission. Transmission is the vital link between power generation and power consumption. Finally, these transmission line voltages arrive at a distribution station where the voltage is reduced by step-down transformers and distributed to different companies who bring power to our homes and workplaces. This process is shown in Fig. 1.1

Prior to the technology boom and power electronics, power quality was not a hot topic aside from power factor. Almost all loads were assumed to be “linear” to a certain degree. This assumption does not prove accurate anymore thanks to an increase in electronically controlled loads. Due to these nonlinear loads, harmonic rich voltages and currents now feed the load and what was previously a smooth sinusoid at the load is now a highly distorted signal.
Excessive harmonics in an AC power system can overheat transformers, cause exceedingly high neutral conductor currents in three-phase systems, create electromagnetic “noise” in the form of radio emissions that can interfere with sensitive electronic equipment, reduce electric motor horsepower output, among others, and can be difficult to pinpoint.

In order to use protective equipment such as frequency load shedding or frequency relay protection, we have to be able to accurately estimate the fundamental frequency of the power system in question. Moreover, frequency estimation is also used for protection against loss of synchronism, under-frequency relaying, and power system stabilization [1, 2]. As it will be described in the following sections, several signal processing algorithms have been developed for such task.
1.2 Overview of Frequency Estimation in Power Systems

The grid fundamental frequency is considered a time-varying parameter not only due to harmonics but also due to mismatches between power generation and power consumption [3]. The variation of the fundamental frequency is more likely to occur if a generator which is isolated from the grid is supplying the loads [4]. As explained in [4], the fundamental frequency decreases when large loads are connected or when a large generation source goes offline. The opposite holds true for an increase in the fundamental frequency (e.g. generation exceeds consumption).

The current research work in sustainable energy systems is facing several challenges regarding the estimation of the fundamental frequency. Renewable energy sources like solar energy, wind turbine, among others are more likely to exhibit pronounced system frequency variations than the controllable energy sources like hydroelectricity. This is due to the frequent mismatch between power supply and demand. The importance of frequency estimation in the smart grid is further discussed in [5].

Traditionally, frequency is estimated using the time between two zero crossings and the calculation of the number of cycles [6, 7]. However, due to possible transients and abnormal conditions that may occur, the voltage and current signals will be distorted making the above mentioned estimation method inadequate. In order to overcome this disadvantage, several methods have been proposed recently. Fast Fourier Transform (FFT) [8], Least Squares (LS) [9, 10], Prony’s estimation [11], multiple frequency trackers [12], and variants of the Kalman filter [13-15] are
some of the acknowledged signal processing algorithms used in frequency estimation.

The LS algorithm and its adaptive version, Least Mean Squares (LMS), are very simple and computational efficient algorithms. The downside this algorithm is inaccuracies due to decaying DC components and random noise. Singular Value Decomposition (SVD), used in Prony analysis, has the ability to reduce the effect of noise and can estimate higher order harmonics. However, its rather complex algorithm makes it computationally expensive making its real-time application unrealistic. SVD is usually implemented with offline systems when accuracy is more important than computational efficiency. The computational cost of DFT based algorithms is very low but its performance is adversely affected by low SNR values.

Among the numerical algorithms described above, Kalman filters (KF), specifically the extended Kalman filter (EKF), have attracted widespread attention, as they can accurately estimate the amplitude, frequency, and phase of a signal at relatively low SNR levels. However, in practice, the EKF algorithm possesses several disadvantages. These disadvantages arise from the fact that the EKF linearizes nonlinear systems so that the KF equations can be applied. The first-order linearization may introduce large errors in the mean and covariance of the state vector, leading to a highly unstable filter that may cause a divergence phenomenon [16]. Moreover, for complex state-spaces or very large systems, the calculation of the Jacobian matrix can be computationally expensive.

The EKFs cited above are able to track small linear changes in system frequency in high noise environments, but fail to track sudden large step changes during
Transient conditions. Iterative methods based on Kalman filtering have been proposed in [17, 18]. These approaches can overcome accuracy problems of the EKF which ignores nonlinear terms. However, the iterative procedure leads to a much higher processing time.

This thesis work focuses on the study of a nonlinear derivative-free adaptive filter algorithm that exploits the capabilities of the Unscented Kalman Filter (UKF). This is a novel estimation technique based on Unscented Transformation (UT) theory. This method, which originally appeared in [19], eliminates the disadvantages of the EKF because it does not linearize the system. The complex UKF, just like the extended complex Kalman filter, uses a complex state-space model of the sinusoidal voltage and computes the covariance and the Kalman gain from the measurements corrupted with white noise. This algorithm implements the strong tracking filter condition (STF) in order to desensitize the filter from initial conditions. It also augmented with a second UKF which used in parallel to estimate the noise covariance. This configuration is known as a Master-Slave UKF (MS-UKF).
2.1 Signal Modeling

Power system signals may be represented in many different forms. Several linear and nonlinear models have been proposed to estimate frequency, phase, and amplitude for a single sinusoid voltage and three-phase voltages. The next subsections will expand on the model for both a single phase voltage and three phase voltages.

2.1.1 Single Sinusoid Voltages

In a power system, the observation sampled signal $y_k$ at time instant $k$ can be expressed as a sum containing $m$ sinusoids with white Gaussian noise

$$y_k = z_k + v_k \quad k = 1, 2, \ldots N$$
$$v_k \sim N(0, R)$$

where

$$z_k = \sum_{n=1}^{M} a_n \cos(\omega_n t_k + \phi_n) \quad M = 1, 2, \ldots$$

$$\omega_n = 2\pi f_n \quad \text{and} \quad t_k = kT_s$$
In (2.1), the parameters defined as \( a_n, f_n, \) and \( \phi_n \) are the amplitude, frequency, and phase of the \( n^{th} \) sinusoid, respectively. The observation noise \( v_k \) is defined as Gaussian white noise with zero-mean and variance \( \sigma^2 \), and \( T_s \) is the sampling period.

We will assume that the harmonics of the system have been filtered out (i.e. the percentage of frequency components other than the fundamental frequency is low). Therefore these harmonic components can be ignored in the model. Then the resulting model (ignoring harmonics) becomes

\[
z_k = a_1 \cos(k \omega_1 T_s + \phi_1), \quad k = 1, \ldots, N,
\]

where \( a_1 \) is the amplitude of the signal, \( \omega_1 \) is the fundamental frequency and \( \phi_1 \) is the phase angle.

Since the observation signal is real, we use a Hilbert transform [20] to convert the real signal \( z_k \) to an analytic signal of the form

\[
y_k = a_1 e^{i(k \omega_1 T_s + \phi_1)} + v_k
\]

### 2.1.2 Three-Phase Voltages

A widely used and common method to generate, transmit, and distribute AC power is via three-phase voltages. This poly-phase system is usually more economical than a single-phase system because it uses less conductor material to transmit the same amount of power.
In a balanced system the three-phase voltages of a power system can be represented in discrete form as

\[
V_{a_k} = V_m \sin (\omega k T_s + \phi) + v_{a_k}, \\
V_{b_k} = V_m \sin \left(\omega k T_s + \phi - \frac{2\pi}{3}\right) + v_{b_k}, \\
V_{c_k} = V_m \sin \left(\omega k T_s + \phi + \frac{2\pi}{3}\right) + v_{c_k}
\]  

(2.5)

where the parameters defined as \(V_m\), \(\omega\), \(T_s\), and \(\phi\) are the peak amplitude of the fundamental component, angular frequency, sampling period, and phase respectively. The terms \(v_{a_k}\), \(v_{b_k}\), and \(v_{c_k}\) are noise terms associated with the observations and are defined as Gaussian white noise with zero mean and variance \(\sigma_v^2\).

2.2 Clarke Transform

This mathematical transformation, also known as the \(\alpha\beta\) -transform, is employed to make the analysis of three-phase circuits simple.

The \(\alpha\beta\) -transform applied to three phase voltages is expressed as [21]

\[
\begin{bmatrix}
V_{a_k} \\
V_{b_k} \\
V_{c_k}
\end{bmatrix} = T \begin{bmatrix}
V_{a_k} \\
V_{b_k} \\
V_{c_k}
\end{bmatrix} = \begin{bmatrix}
1 & -\frac{1}{2} & -\frac{1}{2} \\
0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\
\frac{1}{2} & \frac{1}{2} & \frac{1}{2}
\end{bmatrix} \begin{bmatrix}
V_{a_k} \\
V_{b_k} \\
V_{c_k}
\end{bmatrix}
\]  

(2.6)
where $V_{a_k}$, $V_{b_k}$, and $V_{c_k}$ are generic three-phase voltage sequences and $V_{\alpha_k}$, $V_{\beta_k}$, and $V_{\gamma_k}$ are the corresponding voltage sequences given by the transformation $T$.

In a balanced system $V_a + V_b + V_c = 0$ and thus $V_{\gamma} = 0$. Therefore, the simplified transformation becomes

$$
\begin{bmatrix}
V_{\alpha_k} \\
V_{\beta_k}
\end{bmatrix} = \sqrt{2} \begin{bmatrix}
 1 & -\frac{1}{2} & -\frac{1}{2} \\
0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2}
\end{bmatrix}
\begin{bmatrix}
V_{a_k} \\
V_{b_k} \\
V_{c_k}
\end{bmatrix}
$$

(2.7)

This transformation allows us to represent the three-phase voltages in a complex form $V_{k}$

$$V_{k} = V_{\alpha_k} + jV_{\beta_k}$$

(2.8)

and the voltage $V_{k}$ can be modeled as

$$V_{k} = A e^{j(\omega t + \phi)} + v_{k}$$

(2.9)

where $A$ is the amplitude of the complex signal $V_{k}$ and $v_{k}$ is its noise component.

We can already see the similarity between the single phase case and the three-phase case using the Clarke transformation.

Shown in Fig. 2.1 is a graphic representation of the alpha-beta transform
2.3 State Space Representation

The nonlinear model for both single phase and three-phase voltages can be represented by a complex state space model using the state variables $x_{1k}$ and $x_{2k}$ as

$$x_k = \begin{bmatrix} x_{1k} \\ x_{2k} \end{bmatrix} = \begin{bmatrix} e^{j\omega T_k} \\ a_i e^{j(k\omega T_k+\phi)} \end{bmatrix}$$  \hspace{1cm} (2.10)

Current DSP processors allow computations in the complex domain to be relatively easy. The complex representation of the signal is much simpler and direct when talking about frequency measurement. It is also worth mentioning that the model can easily be adapted to represent signals containing harmonics. The observation signal $V_k$ can now be modeled in the state space form as

Figure 2.1: Graphical representation of Clarke transform [22].
\[
x_{k+1} = \begin{bmatrix} x_{1_{k+1}} \\ x_{2_{k+1}} \end{bmatrix} = \begin{bmatrix} x_i \\ x_i \\ x_{1_k} \\ x_{2_k} \end{bmatrix} + \eta_k
\]  
(2.11)

\[
y_{k+1} = V_k = [0 \ 1] \begin{bmatrix} x_i \\ x_{2_k} \end{bmatrix} + v_k
\]  
(2.12)

Furthermore, the measurements and observations can be represented as

\[
x_{k+1} = f(x_k) + \eta_k
\]
\[
y_k = h(x_k) + v_k
\]  
(2.13)

where

\[
f(x_k) = \begin{bmatrix} x_i \\ x_{1_k} \\ x_{2_k} \end{bmatrix}
\]  
(2.14)

\[
h(x_k) = [0 \ 1] \begin{bmatrix} x_{1_k} \\ x_{2_k} \end{bmatrix}
\]  
(2.15)

Whether we employ a single phase or three phase voltage, the complex signal state space model in (2.14) and (2.15) will be employed to estimate the frequency of the time-varying sinusoid signals corrupted with white noise [23].
Chapter 3: Kalman Filters

The Kalman Filter (KF) dates back to 1960, when R.E. Kalman first published his famous publication describing a recursive solution to the discrete-data linear filtering problem [24]. Since then, and thanks to advances in digital computing, the Kalman Filter has been the subject of extensive research and application in many fields of Engineering, Economics, and Science. The KF, also known as a linear quadratic estimator, is based on probability theory principles. More specifically, it is based on the properties of conditional Gaussian random variables and it can be thought of a set of mathematical equations that aim to minimize the state vector covariance norm.

3.1 The Kalman Filter

The Kalman filter, in essence, addresses the general problem of trying to estimate the state $\mathbf{x} \in \mathbb{R}^n$ of a discrete time process that is governed by the linear stochastic difference equation

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}$$

(3.1)

where $A$ is an $n \times n$ matrix that relates the state at the previous time step $k-1$ to the state at the current time step $k$, and $B$ relates the optional control input to the state $x$. 
This process contains an \( m \)-set of measurements \( z \in \mathbb{R}^m \), where

\[
z_k = H x_k + v_k
\]  

where \( H \) relates the state \( x \) to the measurement \( z \). Note that, in practice, \( A \), \( B \), and \( H \) might change with time.

The random vectors \( w_k \) and \( v_k \), known as the process and measurement noise respectively, are assumed to be independent, white, and with normal probability distributions

\[
p(w) \sim N(0, Q) \\
p(v) \sim N(0, R)
\]  

(3.3)

It is rather intuitive to think about the measurement noise as the amount of noise introduced by sensors or any other measurement equipment. However, the process noise is not as intuitive to grasp as one might imagine. Here, the process noise represents the amount of uncertainty in the system. In other words, the process noise represents how accurate our model of the system is and any perturbations that might occur on the state.

We’ll take a tangent for a moment and try to illustrate the process noise with a rather simple example. Imagine a cannon ball that has been fired and we want to determine its position at time \( k \). Any measurement instrument that we use to determine the position of the cannon ball will introduce its own noise (i.e. measurement noise) but this will not affect the state (trajectory) of the cannon ball. However if we take into account wind, rain, hail, etc., we can thing of these factors as process noise because they affect the state of the cannon ball. Note that this is an
illustrative example to better understand what is meant by process noise. In reality this example would not be suited for a KF because this not considered white noise.

In many practical applications the measurement noise for a system is very well defined since we have a good idea of the output SNR of our sensors. However, measurement noise is not always well defined. Moreover, $Q$ and $R$ also might change with time degrading the filter performance as we will later see.

### 3.1.1 The Computational Origins of the Filter

Imagine we are given the knowledge of a process at time step $k-1$ and a measurement $z_k$. Then, we can define an a-priori and a-posteriori sate estimate at time state $k$ as $\hat{x}_k \in \mathbb{R}^n$ and $\hat{x}_k \in \mathbb{R}^n$, respectively. The a-priori and a-posteriori estimate errors given these states can now be defined as

$$
e_k^- = x_k - \hat{x}_k^-
$$

$$
e_k^+ = x_k - \hat{x}_k
$$

Therefore, the a-priori estimate error covariance is

$$
P_k^- = E\left[ e_k^- e_k^{-T} \right]
$$

(3.5)

and the a-posteriori estimate error covariance is

$$
P_k^+ = E\left[ e_k e_k^T \right]
$$

(3.6)
The goal is to find an equation that computes an a-posteriori state estimate $\hat{x}_k$ as a linear combination of an a-priori estimate $\hat{x}_{k-1}$ and a weighted difference between a measurement $z_k$ and a measurement prediction $H\hat{x}_{k-1}$ as shown in (3.7) [25]

$$\hat{x}_k = \hat{x}_{k-1} + K_k (z_k - H\hat{x}_{k-1})$$  \hspace{1cm} (3.7)

The residual $(z_k - H\hat{x}_{k-1})$, also called measurement innovation, reflects the discrepancy between the actual and predicted measurements. Hence a residual of zero implies the two are in complete agreement.

The $n \times m$ gain matrix $K$ in (3.7) is chosen in such a way that the a-posteriori error covariance in (3.6) is minimized. The minimization is accomplished by first substituting (3.7) into (3.4), and then substituting this result into (3.6). The complete derivation can be found in [26]. One of the resulting gain matrices that minimizes (3.7) is given as [26]

$$K_k = P_k^{-1}H^T (HP_k^{-1}H^T + R)^{-1}$$  \hspace{1cm} (3.8)

By inspecting (3.8) and as $R$ approaches zero, the measurement $z_k$ is more “trusted” than the prediction $H\hat{x}_{k-1}$. On the other hand, as $P_k^{-1}$ approaches zero, the prediction $H\hat{x}_{k-1}$ is “trusted” more than the measurement.
3.1.2 The KF Algorithm

The Kalman filter uses a form of feedback control to estimate a process. In other words, the filter estimates the process state at some time $k$ and then obtains feedback in the form of measurements.

The Kalman filter possesses a time update and a measurement update. The equations associated with the time update provide the filter with the a-priori estimates for the next time step. On the other hand, the equations associated with the measurement update are responsible for incorporating a new measurement into the a-priori estimate to obtain an improved a-posteriori estimate. Simply put, the time update equations can be interpreted as a predictor set of equations and the measurement update equations can be interpreted as a set of corrector equations.

**Time Update:**

\[
\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (3.9)
\]

\[
P_k^- = AP_{k-1}A^T + Q_k \quad (3.10)
\]

As can be seen from (3.9) and (3.10), these equations predict the state and covariance estimates forward in time based the previous time step $k - 1$.

**Measurement Update:**

\[
K_k = P_k^-H^T(HP_k^-H^T + R)^{-1} \quad (3.11)
\]
\[ \hat{x}_k = \hat{x}_{k-1} + K_k (z_k - H\hat{x}_{k-1}) \]  

(3.12)

\[ P_k = (I - K_k H) P_{k-1} \]  

(3.13)

Note how the a-posteriori estimate \( \hat{x}_k \), is a version of the a-priori estimate and weighted difference of the error between estimation and measurement. Fig. 3.1 shows the recursive property of the KF.

Although powerful and elegant, this optimal filter works well only with linear models. However, as we know, real world applications are rarely linear. Therefore we need to employ a variant of the Kalman filter algorithm known as the extended Kalman filter to deal with non-linear problems.
3.2 The Extended Kalman Filter

As mentioned before, most real-life systems are nonlinear and we need a way to deal with them. A Kalman filter that linearizes about the current mean and covariance is referred to as an extended Kalman filter (EKF). This variation of the Kalman filter addresses the filtering problem when the system dynamics, that is the state and observations, are nonlinear. Without loss of generality, we consider the system has no external inputs and it is defined by the following nonlinear relations

\[ x_{k+1} = f(x_k) + \eta_k \]  
(3.14)

\[ y_k = h(x_k) + v_k \]  
(3.15)

where \( x_k \) is the \( n \)-dimensional state vector, \( y_k \) is the \( m \)-dimensional observation vector, \( f(\cdot) \) and \( h(\cdot) \) are nonlinear vector functions, and \( \eta_k \) and \( v_k \) are are white Gaussian, independent random processes with zero mean and covariance

\[ R_k = E[v_k v_k^T] \]  
(3.16)

\[ Q_k = E[\eta_k \eta_k^T] \]  
(3.17)

In order to evaluate the first and second moments of (3.14) - (3.15), the optimal nonlinear filter has to propagate the entire probability density function (pdf) which, in the general case, represents a heavy computational burden.
In something akin to a Taylor series, we can 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. For this approximation to be valid, this linearization should be a good approximation of the nonlinear model in the entire uncertainty domain associated with the state estimate.

Let $F_k$ and $H_k$ be the Jacobian matrices of $f(\cdot)$ and $h(\cdot)$ denoted by

\[
F_k = \nabla f_k \bigg|_{\hat{x}_k}
\]

\[
H_{k+1} = \nabla h_k \bigg|_{\hat{x}_{k+1}}
\]

then we can proceed to formulate the EKF algorithm

### 3.2.1 EKF Algorithm

The extended Kalman filter algorithm is an adapted version of the linear KF. Therefore it uses the same recursive method to estimate the state of the system and can be broken down into a time and measurement update as follows

**Time Update:**

\[
\hat{x}_k^- = f(\hat{x}_{k-1})
\]

\[
P_k^- = F_{k-1} P_{k-1} F_{k-1}^T + Q_k
\]

```
Measurement Update:

\[ K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \]  \hspace{1cm} (3.22)

\[ \hat{x}_k = \hat{x}_k^- + K_k (z_k - h(\hat{x}_k^-)) \]  \hspace{1cm} (3.23)

\[ P_k^- = (I - K_k H_k) P_k^- \]  \hspace{1cm} (3.24)

3.3 The Unscented Kalman Filter

3.3.1 The Unscented Transform

The unscented transformation is a relatively new method for calculating the statistics of a random variable when this on undergoes a nonlinear transformation. This method works under the assumption that is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function [27]. The idea behind the unscented transformation is illustrated in Fig. 3.2.

In this transformation a set of carefully chosen sample points (sigma points) are used to capture the true mean, \( \hat{x} \), and true covariance, \( P_{xx} \), of a Gaussian RV. The nonlinear function is applied to each sigma point to yield a set of transformed points that captures the posterior mean, \( \hat{y} \), and covariance, \( P_{yy} \), to the 3rd order for any nonlinearity.
Figure 3.2: Unscented transformation.

This method has a superficial resemblance to Monte Carlo-type methods. However, there is an important and fundamental difference. The samples are not random samples but instead are based on a specific, deterministic algorithm. Since statistical convergence is not an issue, the higher order information terms about the distribution can be captured using a small number of points [28].

Consider propagating an $n$-dimensional random variable $x$ through a nonlinear function $y = h(x)$. Furthermore, assume that $x$ has a statistical mean $\hat{x}$ and a covariance $P_x$. This random variable can be approximated by $2n+1$ weighted points as

\[
\begin{align*}
\hat{x}_{0,k-1} &= \hat{x}_{k-1} \\
\hat{x}_{i,k-1} &= \hat{x}_{k-1} + \zeta \left( \sqrt{P_{k-1}} \right), \quad i = 1, \ldots, n \\
\hat{x}_{i+n,k-1} &= \hat{x}_{k-1} - \zeta \left( \sqrt{P_{k-1}} \right), \quad i = 1, \ldots, n
\end{align*}
\]

\[
\begin{align*}
W_0^n &= \frac{\lambda}{n + \lambda} \\
W_0^c &= \frac{\lambda}{n + \lambda} + (1 - \alpha^2 + \beta) \\
W_i^n &= W_i^c = \frac{1}{2(n + \lambda)}
\end{align*}
\]
where \( (\sqrt{P_{k-1}})_i \) is the \( i^{th} \) column of the matrix \( \sqrt{P_{k-1}} \), \( n \) is the number of sigma points, 
\[ \zeta = \sqrt{n + \lambda} \], and \( \lambda = n(\alpha^2 - 1) \).

The parameter \( \alpha \) describes the spread of the \( i^{th} \) sigma point around \( \hat{x}_{k-1} \). If \( \lambda > 0 \), the points are scaled further from \( \hat{x}_{k-1} \) and when \( \lambda < 0 \), the points are scaled towards \( \hat{x}_{k-1} \). Moreover, the parameter \( \alpha \) can be chosen to be between \( 10^{-4} \) and 1, and can be used to control the amount of higher-order nonlinearities around \( \hat{x}_{k-1} \).

The matrix \( \hat{H}_{k-1} \) is assumed positive definite. Therefore its square root can be computed using Cholesky decomposition and reduce computational time.

Then the statistics of \( y \) can computed by propagating each \( \chi_i \) through the nonlinear function \( h(x) \)

\[
\gamma_{k-1} = h(\chi_i) \quad (3.26)
\]

The mean is given by the weighted average of the transformer points as on (3.27) and the covariance is the weighed outer product of the transformed points as on (3.28)

\[
\hat{y} = \sum_{i=1}^{2n+1} W_i^m \gamma_i \quad (3.27)
\]

\[
P_{\gamma} = \sum_{i=1}^{2n+1} W_i^c \left[ \gamma_i - \hat{y} \right] \left[ \gamma_i - \hat{y} \right]^* \quad (3.28)
\]

The following summary of this algorithm’s results as stated in [28, 29, 30] is as follow:
1. The mean and covariance of \( x \) are captured precisely up to the second order. Therefore the calculated values of the mean and covariance of \( x \) are correct to the second order as well. Hence the calculated mean has a higher order of accuracy than the EKF, whereas the covariance is calculated to the same order of accuracy.

2. Since the distribution of \( x \) is approximated, its series expansion is not truncated to a particular order. It can be shown that the unscented algorithm is able to partially incorporate information from the higher orders, leading to even greater accuracy.

3. The sigma points capture the same mean and covariance irrespective of the choice of matrix square root which is used. Numerically efficient and stable methods such as the Cholesky decomposition \([31]\) can be used.

4. The mean and covariance are calculated using standard vector and matrix operations. This means that the algorithm is suitable for any choice of process model, and implementation is extremely rapid because it is not necessary to evaluate the Jacobians which are needed in an EKF.

5. The factor \( \lambda \) provides an extra degree of freedom to “fine tune” the higher order moments of the approximation, and can be used to reduce the overall prediction errors.

It is because this and its other properties that the unscented transform is much better suited than linearization when it comes to dealing with nonlinear processes. Fig. 3.3 illustrates the difference, in terms of accuracy, between the EKF and UKF.
3.3.2 UKF Algorithm

The UKF algorithm for estimation, according to the state space mode in Chapter 2, is performed as follows [32]:

**Step-1: Sigma Points Calculation in the $k^{th}$ time instant:** Given an $n-1$ state vector at time $k-1$ and a state error covariance matrix $\hat{P}_{k-1}$, the set of $(2n+1)$ sigma points is computed as follows

\[
\begin{align*}
Z_{0,k-1} &= \hat{x}_{k-1} \\
Z_{i,k-1} &= \hat{x}_{k-1} + \zeta \left( \sqrt{\hat{P}_{k-1}} \right), \quad i = 1, \ldots, n \\
Z_{i,n+k-1} &= \hat{x}_{k-1} - \zeta \left( \sqrt{\hat{P}_{k-1}} \right), \quad i = 1, \ldots, n
\end{align*}
\]  
(3.29)
where \( \begin{pmatrix} \sqrt{P_{k-1}} \end{pmatrix} \) is the \( i^{th} \) column of the matrix \( \sqrt{P_{k-1}} \), \( n \) is the number of sigma points, 
\( \zeta = \sqrt{n + \lambda} \), and \( \lambda = n(\alpha^2 - 1) \).

**Step-2: Propagation of Sigma Points:** The sigma points obtained in (3.29) will be propagated in time through \( f(\cdot) \) in (3.30) in order to obtain the ‘transformed sigma points’ at time instant \( k \).

\[
Z_{i,k|k-1} = f(Z_{i,k-1}) \quad i = 0,1,\ldots,2n
\]  
(3.30)

**Step-3: Calculation of Prior State Estimates:** The prior state estimate \( \hat{x}_{k|k-1} \) and its covariance \( \hat{P}_{k|k-1} \) are approximated by a weighted mean and covariance of the transformed sigma points.

\[
\hat{x}_{k|k-1} = \sum_{i=0}^{2n} W_i Z_{i,k|k-1} \quad \text{(3.31)}
\]

\[
\hat{P}_{k|k-1} = \sum_{i=1}^{2n+1} W_i \left[ Z_{i,k|k-1} - \hat{x}_{k|k-1} \right] \left[ Z_{i,k|k-1} - \hat{x}_{k|k-1} \right]^\ast + Q_{k-1} \quad \text{(3.32)}
\]

where \((\cdot)^\ast\) is the complex conjugate transpose operator and \( Q_{k} \) is the process noise covariance defined as

\[
Q_{k} = E[\eta_{k}\eta_{k}^\ast] = \begin{bmatrix} q_{1} & 0 \\ 0 & q_{2} \end{bmatrix}
\]  
(3.33)

\[
E[\eta_{k}v_{k}^\ast] = \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\]

and for \( i = 0,1,\ldots,2n \)
The constant $\beta$ is used to incorporate part of the prior knowledge of the statistics of $x_k$. It has been shown that $\beta=2$ is optimal for Gaussian distributions [33].

**Step-4: Calculation of Predicted Observations:** The predicted values at time $k$ can be obtained as the weighted sum of the projection of transformed sigma points through the measurement function $h(\cdot)$.

$$y_{4|k-1} = h(x_{4|k-1})$$

$$\hat{y}_{4|k-1} = \sum_{i=0}^{2n} W_i^m y_{4|k-1}$$

where $x_{4|k-1}$ is the sigma points related to the predicted state mean vector and covariance matrix calculated in (3.30).

The a-posteriori estimate is calculated as

$$\hat{x}_k = \hat{x}_{4|k-1} + K_k (y_k - \hat{y}_{4|k-1})$$

where the Kalman gain $K_k$ is given by

$$W_0^m = \frac{\lambda}{n+\lambda}$$

$$W_0^c = \frac{\lambda}{n+\lambda} + (1-\alpha^2 + \beta)$$

$$W_i^m = W_i^c = \frac{1}{2(n+\lambda)}$$

(3.34)
\[ K_k = P_{y_k|y_{k-1}} (P_{y_k|y_{k-1}})^{-1} \]  (3.38)

and

\[ P_{y_k|y_{k-1}} = \sum_{i=0}^{2n} W_{i}^{r} \left[ \mathbf{y}_{i,k|k-1} - \hat{\mathbf{y}}_{i,k|k-1} \right] \left[ \mathbf{y}_{i,k|k-1} - \hat{\mathbf{y}}_{i,k|k-1} \right]^T + R_{k-1} \]  (3.39)

\[ P_{\hat{y}_k|y_{k-1}} = \sum_{i=0}^{2n} W_{i}^{r} \left[ \hat{\mathbf{y}}_{i,k|k-1} - \hat{\hat{\mathbf{y}}}_{i,k|k-1} \right] \left[ \mathbf{y}_{i,k|k-1} - \hat{\mathbf{y}}_{i,k|k-1} \right] \]  (3.40)

Recall that \( R^r \) is the measurement noise covariance and is given by

\[ R^r = E \left[ \mathbf{v}_k \mathbf{v}_k^T \right] \]  (3.41)

then a posteriori estimate of the error covariance matrix is given by

\[ \hat{\hat{\mathbf{P}}}_k = \hat{\hat{\mathbf{P}}}_{k|k-1} - K_k P_{y_k|y_{k-1}} K_k^T \]  (3.42)

### 3.4 Strong Tracking Filter Condition

A filter may be called a strong tracking filter if and only if it satisfies the orthogonal theorem in [34]. As noted in [35], STF can self-adaptively tune the error covariance matrix (\( P_k \)) and the corresponding Kalman gain (\( K_k \)) by introducing a “time-varying fading matrix.” This fading matrix is applied in the form

\[ P_{4|k-1} = L M D_k F_{k-1} P_{k|k-1} F_{k-1}^T + Q_{k-1} \]  (3.42)
where $LMD_k = diag[\lambda_1, \lambda_2, ..., \lambda_n]$ is the so-called fading matrix and $\lambda_i$'s are the fading factors ($\lambda_i \geq 1 \quad i = 1, 2, ..., n$).

Furthermore, when there is a rough knowledge of the system, the fading matrix can be calculated using the following algorithm:

\[
V_0 = \begin{cases} 
\psi_0 \psi_0^T, & k = 0 \\
\rho V_{n-1} + \psi_k \psi_k^T \over 1 + \rho, & k \geq 1
\end{cases} \quad (3.43)
\]

\[
N_k = V_{n-1} - \beta R_k - H_k Q_k H_k^T \quad (3.44)
\]

\[
M_k = F_{k-1} P_{k-1} F_{k-1}^T H_k H_k^T \quad (3.45)
\]

\[
c_k = \frac{tr[N_k]}{\sum_{i=1}^{n} \alpha M_{a_k}} \quad (3.46)
\]

\[
\lambda_k = \begin{cases} 
\alpha c_k, & \alpha c_k > 1 \\
1, & \alpha c_k \leq 1
\end{cases} \quad (3.47)
\]

In this algorithm, $H_k$ and $F_k$ are the local linearized matrices of the system and measurement models, respectively. The process and measurement noise covariances are represented by $Q_k$ and $R_k$, respectively. The variable $\psi_k$ represents the difference, also known as the innovation, between the true and estimated values. $\rho$ and $\beta$ are the forgetting and weak factors, respectively and are set according to empirical knowledge. Finally, the coefficient values $\alpha_i \geq 1 \quad i = 1, 2, ..., n$ are predetermined by previous knowledge of the system. If there is no prior knowledge about the system, these coefficients can be selected to be “1”.
3.4.1 UKF Based on STF Condition

As described in [35], when a filter meets the STF condition it possesses a strong tracking ability and it has good robustness for sudden changes in the state. Therefore the UKF filter under STF condition should outperform the regular UKF. Moreover, the UKF filter algorithm remains virtually unchanged with the exception that (3.32) is replaced by (3.42).

It is well known there exists an inherent trade-off between speed of convergence and accuracy of the filter due to discrepancies between the estimated and actual values of the noise covariance. If, as in the following example, the process noise covariance is underestimated then the filter will take a longer to converge. Furthermore, if the process noise covariance is highly underestimated due to lack of knowledge of the system the filter is susceptible to divergence.

In Fig. 3.4 we can see that, even when the noise covariance is highly underestimated, the UKF filters under STF will eventually converge to the actual signal. We can also see that for high $\beta$ values the tracking ability becomes smoother at the price of slower convergence time. In a similar manner, lower $\beta$ values imply a higher sensitivity to uncertainty. Therefore a compromise has to be made between the smoothness of the estimation and convergence time. This compromise can be less critical if we could successfully approximate the process noise covariance, $Q$, to its true value.

It is important to note yet another benefit from the STF condition. We know that the choice of initial conditions can be critical since a poor choice will lead to filter divergence. The plot from Fig. 3.5 shows how the UKF cannot converge to the true
value due to its chosen initial conditions. However, when we examine the UKF under STF we can see that the filter converges to the true value after the time-varying fading matrix settles.

Figure 3.3 Performance of UKF algorithm for different values of $\beta$. 

Figure 3.3 Sensitivity to initial conditions.
Chapter 4: Master-Slave Unscented Kalman Filter

4.1 Structure

In real physical applications, the difference between the a-priori knowledge and the true state statistics is the major factor that degrades the filter’s performance. Therefore, selecting appropriate covariance matrices \((Q^\nu, R^\nu)\) is of utmost importance in order to maintain the desired performance and filter stability of the UKF. We propose the use of a slave UKF to estimate the covariance in real time. Under the assumption that the process and measurement noises are Gaussian and white, one can conclude that the covariance matrices \(Q^\nu\) and \(R^\nu\) are diagonal matrices [23]. Then the estimation of the noise covariance can be simplified to the estimation of the diagonal elements.

As shown in Fig.4.1, the proposed adaptive frequency estimation scheme is composed of two parallel filters. At every time step, the master UKF estimates the states using the noise covariance obtained by the slave UKF, while the slave UKF estimates the noise covariance using the innovations generated by the master UKF. It must be noted that the two UKFs are independent in the MS-UKF structure.
In this configuration, the slave UKF will be used to estimate the measurement covariance, and since $R^v$ is a scalar value, we will let $\theta_k = R_k^v$. Moreover, if the dynamics of $\theta$ are known, the state equation of the slave UKF is

$$\theta_k = f_\theta(\theta_{k-1}) + w_{\theta_k} \quad (4.1)$$

If the dynamics of $\theta$ are unknown, they can be modeled as a non-correlated random drift vector, i.e.

$$\theta_k = \theta_{k-1} + w_{\theta_k} \quad (4.2)$$

where $w_{\theta_k}$ is Gaussian white noise with zero mean. The innovation covariance generated by the master UKF is taken as the observation signal for the slave UKF and then according to equation (3.39) the observation model can be described as:
\[ \hat{s}_k = g(\hat{\theta}_k) = \sum_{i=0}^{2n} W_i \left| y_{i,k} - \hat{y}_{i,k-1} \right|^2 + R^v \] (4.3)

The measurement \( s_k \) received by the slave UKF is

\[ s_k = |v_k|^2 \]
\[ v_k = y_k - \hat{y}_{i,k-1} \] (4.4)

4.2 MS-UKF Algorithm

The algorithm of the Master-Slave UKF can be computed similarly to that of its master counterpart.

**Master-1: Sigma Points Calculation in the \( k^{th} \) time instant:**

\[ \chi_{0,k-1} = \hat{x}_{k-1} \]
\[ \chi_{i,k-1} = \hat{x}_{k-1} + \zeta \left( \sqrt{P_{k-1}} \right)_i, \quad i = 1, \ldots, n \] (4.5)
\[ \chi_{i+n,k-1} = \hat{x}_{k-1} - \zeta \left( \sqrt{P_{k-1}} \right)_i, \quad i = 1, \ldots, n \]

where \( \left( \sqrt{P_{k-1}} \right)_i \) is the \( i^{th} \) column of the matrix \( \sqrt{P_{k-1}} \), \( n \) is the number of sigma points, \( \zeta = \sqrt{n + \lambda} \), and \( \lambda = n(\alpha^2 - 1) \).
Master-2: Time Update:

\[
\begin{align*}
Z_{k|k-1} &= f(Z_{i,k-1}) \\
\hat{x}_{k|k-1} &= \sum_{i=0}^{2n} W_i^m \cdot x_{i,k|k-1} \\
\hat{P}_{k|k-1} &= \sum_{i=0}^{2n} W_i^c \left[ Z_{i,k|k-1} - \hat{x}_{k|k-1} \right] \left[ Z_{i,k|k-1} - \hat{x}_{k|k-1} \right]^T + Q_k \\
Y_{i,k|k-1} &= h(Z_{i,k|k-1}) \\
\hat{y}_{k|k-1} &= \sum_{i=0}^{2n} W_i^m Y_{i,k|k-1}
\end{align*}
\]  

(4.6)

Master-3: Measurement Update:

\[
\begin{align*}
K_k &= P_{y_k y_k} \left( P_{y_k y_k} \right)^{-1} \\
P_{y_k y_k} &= \sum_{i=0}^{2n} W_i^c \left[ Y_{i,k|k-1} - \hat{y}_{k|k-1} \right]^2 + R_k \\
P_{x_k y_k} &= \sum_{i=0}^{2n} W_i^c \left[ Z_{i,k|k-1} - \hat{x}_{k|k-1} \right] \left[ Y_{i,k|k-1} - \hat{y}_{k|k-1} \right]^T \\
\hat{P}_k &= \hat{P}_{k|k-1} - K_k P_{y_k y_k} K_k^T \\
\hat{x}_k &= \hat{x}_{k|k-1} + K_k (y_k - \hat{y}_{k|k-1})
\end{align*}
\]  

(4.7)

where the weights \( W_i^c \), \( W_i^m \), and scalar \( \zeta \) can be calculated by (3.34) and the number of states \( n \) is the number of diagonal elements in \( x_k \).
Slave-1: Sigma Points Calculation in the $k^{th}$ time instant:

\[
\begin{align*}
\varphi_{0,k-1} &= \hat{\theta}_{k-1} \\
\varphi_{i,k-1} &= \hat{\theta}_{k-1} + \rho \left( \sqrt{P_{\theta_{k-1}}} \right)_i \\
\varphi_{i+n,k-1} &= \hat{\theta}_{k-1} - \rho \left( \sqrt{P_{\theta_{k-1}}} \right)_i
\end{align*}
\tag{4.8}
\]

where $\rho = \sqrt{\lambda_0 + 1}$ and $\lambda_0 = \delta^2 - 1$. The parameter $\delta$ is selected in a similar manner as $\alpha$ in the master UKF.

Slave-2: Time Update:

\[
\begin{align*}
\varphi_{i,k|k-1} &= f_\theta (\varphi_{i,k-1}), \ i = 0,1,2 \\
\hat{\theta}_{k|k-1} &= \sum_{i=0}^{2} W^m_{\theta} \varphi_{i,k|k-1} \\
\hat{P}_{\theta_{k|k-1}} &= \sum_{i=0}^{2} W^m_{\theta} \left[ \varphi_{i,k|k-1} - \hat{\theta}_{k|k-1} \right] \left[ \varphi_{i,k|k-1} - \hat{\theta}_{k|k-1} \right]^T + Q^\theta_{k-1} \\
\gamma_{i,k|k-1} &= g \left( \varphi_{i,k|k-1} \right), \ i = 0,1,2 \\
\hat{\gamma}_{k|k-1} &= \sum_{i=0}^{2} W^m_{\gamma} \gamma_{i,k|k-1}
\end{align*}
\tag{4.9}
\]
Slave-3: Measurement Update:

\[
\begin{align*}
K_\theta &= P_{\theta_s} (P_{\theta_s}^{-1} + R^\theta) \\
P_{\theta_s} &= \sum_{i=0}^{2} W_{\theta_i} \left[ \xi_{i,k|k-1} - \hat{s}_{i,k|k-1} \right] \left[ \xi_{i,k|k-1} - \hat{s}_{i,k|k-1} \right]^T \\
P_{\theta_s} &= \sum_{i=0}^{2} W_{\theta_i} \left[ \phi_{i,k|k-1} - \hat{\theta}_{i,k|k-1} \right] \left[ \phi_{i,k|k-1} - \hat{\theta}_{i,k|k-1} \right]^T \\
\hat{\theta}_k &= \hat{\theta}_{k|k-1} + K_{\theta_s} (s_k - \hat{s}_{k|k-1}) \\
\end{align*}
\]

where \( R^\theta \) and \( Q^\theta \) are the slave’s process and measurement noise covariance respectively. Note that the weights \( W_{\theta_i}^c \), \( W_{\theta_i}^m \), and scalar \( \rho \) can be calculated by (3.34).
Chapter 5: Results

The proposed frequency estimation algorithm performance is assessed via extensive computer simulations and compared to other Kalman filters. The voltages are generated in Simulink using a programmable source and the algorithms are implemented in Matlab scripts. The simulations are as follows: First, the EKF, UKF, and MS-UKF are compared under different frequency changes and their MSEs are examined. Next, the error between the actual and estimated values of the UKF, UKF-STF, and MS-UKF-STF are compared. Then, the UKF, MS-UKF, and their STF variations are compared under a dynamic SNR and their MSEs are examined. Finally, the UKF-STF and MS-UKF-STF will be compared under different frequency variation scenarios and their MSEs will be compared.

5.1 Filter Comparison I: EKF vs UKF vs MS-UKF

In order to evaluate the performance of the different estimators, we perform multiple tests such as step, linear, and nonlinear frequency chances. As previously discussed, there exists an inherent trade-off between convergence time and accuracy. It follows that this trade-off may impact the performance of a filter. Therefore, all the estimators will be programmed to start with the same initial conditions and the same a priori knowledge of the process and measurement noise.
5.1.1 Frequency Step Variation

The fundamental frequency of the signal undergoes a step change from 60 Hz to 64 Hz. The performance of the different tracking algorithms and the reconstructed signals are plotted and shown in Figs. 5.1 and 5.2 respectively.

![Graphs showing frequency step-up](image)

Figure 5.1: Frequency step-up at $t = 0.25$ s.

These plots reveal that the EKF possesses the longest convergence time, followed by the UKF. The reconstructed signals in Fig. 5.2 show how the convergence time affects the reconstruction of the signal. It can be seen from Table 5.1 that the UKF and MS-UKF filters are considerable more accurate than the EKF counterpart.
Next, in a similar manner as before, the fundamental frequency component of the signal will experience a step change. However, this time, the frequency transition will be from 60 Hz to 56 Hz. The performance of the tracking algorithms and the equivalent reconstructed signals are shown in Figs. 5.3 and 5.4 respectively.

Table 5.1: MSE of frequency step variation over 100 independent runs.
Figure 5.3: Frequency step-down at $t = 0.25\ s$.

Figure 5.4: Reconstructed signal with frequency step-down at $t = 0.25\ s$. 
It can be easily distinguished that the UKF and MS-UkF algorithms, as in Subsection 5.1.1, have a much faster convergence time than the EKF. This directly translates into a reconstructed signal that accurately tracks the Clarke equivalent signal of the system. It must be noted that the MSE for this case is similar to that of Subsection 5.1.1 (Table 5.1).

5.1.2 Linear Frequency Variation

The fundamental frequency of the signal will experience a linear variation starting at $t = 0.25 \, s$ and ending at $t = 0.75 \, s$, resulting in a frequency change from 60 Hz to 62 Hz. The performance of the different tracking algorithms and the equivalent reconstructed signals are shown in Figs. 5.5 and 5.6 respectively.

Figure 5.5: Linear frequency variation (EKF, UKF, MS-UKF).
Here we can begin to appreciate the trend that, under the same circumstances (i.e. same initial conditions and noise covariance), the UKF and MS-UKF algorithms outperform the EKF in terms of convergence time. The results from Table 5.2 quantify what we can visibly appreciate from the plots in Figs. 5.5 and 5.6.

![Figure 5.6: Reconstructed signal with linear frequency variation I.](image)

The reconstructed signal plot from Fig. 5.6 reveals that the EKF algorithm has some trouble with keeping track of the signal even after the variation has stopped. Meanwhile, the UKF and MS-UKF algorithms are able track the signal with a higher degree of accuracy and lock onto it shortly after the variation has come to a stop. Table 5.2 gives us an insight on the errors associated with this particular type of variation.
Table 5.2: MSE of linear frequency variation over 100 independent runs.

<table>
<thead>
<tr>
<th>SNR</th>
<th>EKF</th>
<th>UKF</th>
<th>MS-UKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>1.88E-01</td>
<td>1.72E-01</td>
<td>7.03E-02</td>
</tr>
<tr>
<td>20</td>
<td>1.45E-01</td>
<td>1.30E-01</td>
<td>5.01E-02</td>
</tr>
<tr>
<td>30</td>
<td>7.38E-02</td>
<td>7.13E-02</td>
<td>2.24E-02</td>
</tr>
<tr>
<td>40</td>
<td>1.52E-02</td>
<td>1.45E-02</td>
<td>3.90E-03</td>
</tr>
<tr>
<td>50</td>
<td>8.30E-03</td>
<td>7.30E-03</td>
<td>1.90E-03</td>
</tr>
<tr>
<td>60</td>
<td>4.00E-03</td>
<td>3.00E-03</td>
<td>1.60E-03</td>
</tr>
</tbody>
</table>

5.1.3 Nonlinear Frequency Variation

The fundamental frequency component of the signal will experience a slow nonlinear variation. This particular type of variation is done by superimposing a time-varying component onto the original frequency signal. The performance of the tracking algorithms and the reconstructed signals are shown in Figs. 5.7 and 5.8 respectively.

Following the trends from the previous simulation scenarios we can visually confirm that the MS-UKF algorithm tracks the nonlinear frequency changes with the most accuracy. This can be confirmed by the MSE values found in Table 5.3. So far we have shown how the tracking filters respond under a constant SNR value. Next we will examine the UKF, MS-UKF, and their variations under a dynamic SNR.
Figure 5.7: Nonlinear frequency variation (EKF, UKF, MS-UKF).

Figure 5.8: Reconstructed signal with nonlinear frequency variation I.
5.2 Filter Comparison II: Dynamic Covariance (UKF vs MS-UKF)

In the previous subsection we established the superiority of the UKF and MS-UKF filters over the EKF filter counterpart. These previous simulations dealt with a constant SNR value. However, we know this is not always the case when dealing with real systems. This subsection focuses on the study of the UKF and MS-UKF algorithms under a dynamic SNR situation.

We now simulate situations similar to the ones seen in Subsection 5.1. Moreover, the SNR in all situations will experience a sudden change from 35 dB to 20 dB.

5.2.1 Dynamic Covariance & Constant Frequency

The fundamental frequency component of the signal will remain constant while the SNR experiences a change from 35 dB to 20 dB at $t = 0.5\ s$. The plots from Figs. 5.9 and 5.10 show the UKF and MS-UKF filters react to this change in SNR.

<table>
<thead>
<tr>
<th>SNR</th>
<th>EKF</th>
<th>UKF</th>
<th>MS-UKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>4.83E-01</td>
<td>3.89E-01</td>
<td>3.39E-01</td>
</tr>
<tr>
<td>20</td>
<td>2.99E-01</td>
<td>2.48E-01</td>
<td>2.16E-01</td>
</tr>
<tr>
<td>30</td>
<td>1.31E-01</td>
<td>1.04E-01</td>
<td>8.33E-02</td>
</tr>
<tr>
<td>40</td>
<td>5.34E-02</td>
<td>3.54E-02</td>
<td>2.72E-02</td>
</tr>
<tr>
<td>50</td>
<td>2.39E-02</td>
<td>1.06E-02</td>
<td>9.50E-03</td>
</tr>
<tr>
<td>60</td>
<td>1.17E-02</td>
<td>4.80E-03</td>
<td>3.40E-03</td>
</tr>
</tbody>
</table>

Table 5.2: MSE of nonlinear frequency variation over 100 independent runs.
Figure 5.9: UKF algorithm with SNR variation at $t = 0.5$ s.

Figure 5.10: MS-UKF algorithm with SNR variation at $t = 0.5$ s.
Both filters were optimized to perform an accurate estimation at and SNR of 35 dB. However, when the SNR experiences the step change to 20 dB we can see that the accuracy is of the filter is compromised. The accuracy degradation is more pronounced in the UKF algorithm because the measurement noise covariance can only be programmed once before the filter starts. This is not the case with the MS-UKF algorithm which updates its measurement noise covariance to adapt to the change in SNR. Subsection 5.2.3 will elaborate more on this concept.

5.2.2 Dynamic Covariance & Linear Frequency Variation

The fundamental frequency of the signal will undergo a linear variation from 60 Hz to 62 Hz while simultaneously experiencing a change in SNR from 35 dB to 20 dB at $t = 0.5\, s$. The performance of the UKF and MS-UKF algorithms and the equivalent reconstructed signals are shown in Figs. 5.11 and 5.12, respectively.

In the previous figures we can observe, again, degradation in accuracy after the SNR has changed. The reconstructed signal from Fig. 5.11 shows the impact of the frequency inaccuracy versus the reference signal. Furthermore, the reconstructed signal from Fig. 5.12 shows that the MS-UKF can accurately keep track of the reference signal by adapting to the change in SNR.
Figure 5.11: Linear frequency variation with SNR step at $t = 0.5 \, s$ (UKF).

Figure 5.12: Linear frequency variation with SNR step at $t = 0.5 \, s$ (MS-UKF).
5.2.3 Dynamic Covariance and Step Frequency Variations

The fundamental frequency of the signal will undergo a step variation from 60 Hz to 64 Hz and from 60 Hz to 56 Hz while simultaneously experiencing a change in SNR from 35 dB to 20 dB at \( t = 0.5 \) s in both cases. The performance of the UKF and MS-UKF algorithms is shown in Figs. 5.13.

![Figure 5.13: Frequency step variations with SNR change at \( t = 0.5 \) s.](image)

Recall there exists a direct tradeoff between accuracy and speed of convergence. Also, the values that optimize a filter for accuracy and/or speed of convergence for a given SNR (e.g. 35 dB) may not be the same values that optimize the filter for a different SNR value (e.g. 20 dB).

The UKF algorithm was programmed, with a finite error, to have a relatively fast speed of convergence at 35 dB. Therefore if we focus our attention in the
lower plot of Fig. 5.13, we see that the change in SNR at \( t = 0.5 \) s does not affect the estimation process too much. However, if we anticipated an SNR value of 35 dB and the system experiences the change in SNR before the step change, then the system exhibits a much slower response. This is seen in the top plot of Fig. 5.13.

In contrast, the MS-UKF algorithm is able to adapt to this change in SNR and outperform the convergence time and accuracy of the UKF algorithm. This example shows powerful characteristic of the adaptive filter.

5.3 Filter Comparison III: UKF vs UKF-STF

Using simulations, we showed in the subsection 5.1 that the UKF and MS-UKF algorithms are superior estimators than the EKF algorithm. In this subsection the UKF will be studied under different SNR levels and will be compared to the UKF under STF condition. We expect, because of the nature of the STF condition, that the estimators will have a similar performance when the noise covariance is chosen to be smaller than its true value (i.e. a much faster but less accurate estimation). However when the opposite occurs the STF condition will help the much slower convergence speed. We will concentrate our efforts in the latter case.

The UKF under STF condition is of interest to us because it desensitizes the filter against poorly chosen initial conditions and poorly estimated process and measurement noise covariance. Therefore when the noise changes with time as seen in subsection 5.2, the filter will be able to converge much faster than the stand-alone UKF algorithm.
5.3.1 Two-Step Frequency Variation

The fundamental frequency of the signal will experience two step changes from 60 Hz to 58 Hz and from 60 Hz to 62 Hz. The performance of the UKF and UKF-STF algorithms is shown in Fig. 5.14.

![Figure 5.14: Double frequency step variation (UKF, UKF-STF).](image)

5.3.2 Linear Frequency Variation (UKF vs UKF-STF)

As in previous cases, the fundamental frequency of the signal will undergo a linear variation from 60 Hz to 62 Hz. The performance of the UKF and UKF-STF algorithms and equivalent reconstructed signals is shown in Figs. 5.15.
The two previous results and the following one clearly show the powerful nature of the strong tracking filter condition. This condition sacrifices a relatively small amount of accuracy, dictated by the forgetting and weak factors, in order to aid the filter to react quicker to sudden changes in frequency. As before, these filters are evaluated using a constant SNR value. Subsection 5.4 will study the UKF-STF and MS-UKF-STF under a dynamic SNR.

5.3.3 Nonlinear Frequency Variation (UKF vs UKF-STF)

The fundamental frequency component of the signal will experience a slow nonlinear variation as the one in Subsection 5.1.3. The performance of the UKF and UKF-STF algorithms is shown in Fig. 5.16.
Figure 5.16: Nonlinear frequency variation (UKF, UKF-STF).

5.4 Filter Comparison IV: Dynamic Covariance (UKF-STF vs MS-UKF-STF)

In a similar manner as in subsection 5.2 we wish to study the UKF-STF and MS-UKF-STF algorithms under a varying SNR levels. We will simulate previous scenarios and apply the SNR change at $t = 0.5 \, s$. The SNR levels in all situations will experience a sudden change from 35 dB to 20 dB.

5.4.1 Nonlinear Frequency Variation and Dynamic Covariance

The fundamental frequency component of the signal will experience a nonlinear with time and, as previously mentioned, the SNR level will change at $t = 0.5 \, s$. 
The performance and reconstructed signals of the UKF-STF and MS-UKF-STF algorithms are shown in Figs. 5.17 and 5.18.

Here we see that the tracking algorithm, during the time interval 0.1 s to 0.5 s is similar to the one in Fig. 5.16. However, the change in SNR quickly degrades the accuracy of the estimation translating in a poorly reconstructed signal. This is not the case for the MS-UKF-STF tracking algorithm that adapts to the change in SNR.

Recall in subsection 5.2 we saw a relatively small degradation in tracking accuracy from the MS-UKF when the SNR changed. This time the time-varying fading matrix introduced with the strong tracking filter condition helps improve the tracking accuracy.
5.4.2 Linear Frequency Variation and Dynamic Covariance

The fundamental frequency of the signal will undergo a linear variation from 60 Hz to 62 Hz while simultaneously experiencing a change in SNR from 35 dB to 20 dB at $t = 0.5$ s. The performance and reconstructed signals of the UKF-STF and MS-UKF-STF algorithms are shown in Figs. 5.19 and 5.20 respectively.
Figure 5.19: Linear frequency variation with SNR step at $t = 0.5\, s$ (UKF-STF).

Figure 5.20: Linear frequency variation with SNR step at $t = 0.5\, s$ (MS-UUKF-STF).
The more pronounced degradation in accuracy exhibited after the SNR change, when compared to subsection 5.2, is due to the weak and forgetting factors. These factors are introduced to deal with the amount of uncertainty that we have about the system. Therefore, when the system changed its SNR our uncertainty of the system also changed resulting in much greater degradation in accuracy.

5.5 Filter Comparison V: MS-UKF-STF vs UKF-STF

In subsection 5.3 we demonstrated how powerful the adaptive nature of the MS-UKF-STF can be in terms of accuracy and speed of convergence. Also, because of the STF condition, the filter is desensitized from poorly chosen initial conditions. In this subsection we maintain a constant SNR level but simulate the UKF-STF with both an underestimated and an overestimated measurement noise covariance and examine the results by means of the MSE.

Here we will be able to visually confirm the trade-off between accuracy and speed of convergence depending on the value chosen for the measurement noise covariance $R$. Note that the over and underestimation are given by a factor of 4 on these examples. This is an optimistic scenario since most papers use a factor of 10. However, we will show that even on an optimistic scenario the MS-UKF-STF algorithm is able to outperform its non-adaptive counterpart.
5.5.1 Two-Step Frequency Variation (MS-UKF-STF vs UKF-STF)

The frequency of the signal will experience two step changes from 60 Hz to 58 Hz and from 60 Hz to 62 Hz. The performance and MSEs of the MS-UKF-STF and UKF-STF are shown in Fig. 5.21 and Table 5.4 respectively.

![Graph showing frequency and amplitude vs time](image)

Figure 5.21: Double frequency step variation (MS-UKF-STF, UKF-STF).

It is worth noting that even though we are not simulating a dynamic SNR, the under and overestimation of the measurement noise covariance can be thought of as a static filter reacting to a change in SNR. That is, if we assumed the SNR was 40 dB for \( t < 1 \) s and optimized our filter for such a case, then at \( t > 0.1 \) s the SNR experienced a change and is now at 30 dB. This would mean that our measurement noise covariance is now underestimated by a finite number in this case.
Table 5.4: MSE of double frequency step over 100 Independent Runs.

<table>
<thead>
<tr>
<th>SNR</th>
<th>UKF-STF (4*R)</th>
<th>UKF-STF (1/4*R)</th>
<th>MS-UKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>8.10E-01</td>
<td>5.10E-01</td>
<td>4.50E-01</td>
</tr>
<tr>
<td>20</td>
<td>7.68E-01</td>
<td>3.36E-01</td>
<td>3.25E-01</td>
</tr>
<tr>
<td>30</td>
<td>5.60E-01</td>
<td>1.66E-01</td>
<td>1.84E-01</td>
</tr>
<tr>
<td>40</td>
<td>1.92E-01</td>
<td>8.36E-02</td>
<td>9.33E-02</td>
</tr>
<tr>
<td>50</td>
<td>9.85E-02</td>
<td>5.44E-02</td>
<td>6.10E-02</td>
</tr>
<tr>
<td>60</td>
<td>5.37E-02</td>
<td>4.08E-02</td>
<td>4.60E-02</td>
</tr>
</tbody>
</table>

5.5.2 Linear Frequency Variation (MS-UKF-STF vs UKF-STF)

The fundamental frequency of the signal will experience a linear variation starting at $t = 0.25 \, s$ and ending at $t = 0.75 \, s$, resulting in a frequency change from 60 Hz to 62 Hz. The performance and MSEs of the MS-UKF-STF and UKF-STF tracking algorithms are shown in Fig. 5.22 and Table 5.5, respectively.
Figure 5.22: Linear frequency variation (MS-UKF-STF, UKF-STF).

<table>
<thead>
<tr>
<th>SNR</th>
<th>UKF-STF (4*R)</th>
<th>UKF-STF (1/4*R)</th>
<th>MS-UKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>4.16E-02</td>
<td>2.71E-02</td>
<td>3.58E-02</td>
</tr>
<tr>
<td>20</td>
<td>2.31E-02</td>
<td>1.54E-02</td>
<td>2.03E-02</td>
</tr>
<tr>
<td>30</td>
<td>7.70E-03</td>
<td>5.20E-03</td>
<td>5.30E-03</td>
</tr>
<tr>
<td>40</td>
<td>2.50E-03</td>
<td>2.40E-03</td>
<td>1.90E-03</td>
</tr>
<tr>
<td>50</td>
<td>9.00E-04</td>
<td>1.20E-03</td>
<td>8.00E-04</td>
</tr>
<tr>
<td>60</td>
<td>3.00E-04</td>
<td>7.00E-04</td>
<td>3.00E-04</td>
</tr>
</tbody>
</table>

Table 5.5: MSE of linear frequency variation over 100 Independent Runs.
5.5.3 Nonlinear Frequency Variation (MS-UKF-STF vs UKF-STF)

The fundamental frequency of the signal will experience a linear variation starting at $t = 0.25 \, s$ and ending at $t = 0.75 \, s$, resulting in a frequency change from 60 Hz to 62 Hz. The performance and MSEs of the MS-UKF-STF and UKF-STF tracking algorithms are shown in Fig. 5.23 and Table 5.6 respectively.

The MSE results from Tables 5.4 through 5.6 show that the adaptive MS-UKF-STF algorithm outperforms its nonadaptive counterpart. This is because we can chose weak and forgetting factors that deal with a “known” amount of uncertainty since the filter is also estimating the noise present in the system. This is in contrast to the initial “guess” of noise that the system will experience, but since this noise is dynamic this “guess” can quickly degrade the accuracy and/or speed of the filter.

Figure 5.23: Nonlinear frequency variation (MS-UKF-STF, UKF-STF).
<table>
<thead>
<tr>
<th>SNR</th>
<th>UKF-STF (4*R)</th>
<th>UKF-STF (1/4*R)</th>
<th>MS-UKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>3.83E-01</td>
<td>1.50E-01</td>
<td>1.82E-01</td>
</tr>
<tr>
<td>20</td>
<td>2.14E-01</td>
<td>6.39E-02</td>
<td>1.12E-01</td>
</tr>
<tr>
<td>30</td>
<td>6.61E-02</td>
<td>2.37E-02</td>
<td>3.34E-02</td>
</tr>
<tr>
<td>40</td>
<td>2.13E-02</td>
<td>8.70E-03</td>
<td>1.19E-02</td>
</tr>
<tr>
<td>50</td>
<td>7.00E-03</td>
<td>4.10E-03</td>
<td>4.30E-03</td>
</tr>
<tr>
<td>60</td>
<td>2.30E-03</td>
<td>1.60E-03</td>
<td>1.60E-03</td>
</tr>
</tbody>
</table>

Table 5.6: MSE of nonlinear frequency variation over 100 independent runs.
Chapter 6: Conclusions

An adaptive grid voltage frequency estimation method based on a two UKF structure is presented. The “Master” UKF filter is modified to operate under STF condition. Furthermore, the extra filter, labeled “Slave”, is operated in parallel with the “Master” UKF to estimate measurement noise statistics and feed this information to it.

The accuracy of the proposed filter algorithm is improved and exhibits the same computational burden as the EKF. Also, because of the STF condition, the estimation algorithm is robust and ensures convergence. Extensive simulations under different SNR conditions are analyzed to determine the performance of the proposed approach.

6.1 The Benefits of an Adaptive Algorithm

The tradeoff between speed and accuracy is always be present in Kalman filters. As mentioned before, the tradeoff becomes more pronounced when the a-priori statistics of the system are not know and/or are time-variant. The adaptive nature of the algorithm allows us to reach a much more balanced tradeoff and ensures convergence of the estimation. Moreover, if the process and/or measurements noises are known to be time-variant, the filter algorithm will also be able to estimate an SNR value at each time step.
6.2 Future Work

The results presented in Chapter 5 show that adaptive algorithms can outperform static filters when the process and/or measurement noise values are time-variant. An improvement proposed to this algorithm is to implement a simpler algorithm for the “Slave” estimator. This is possible because the “Slave” operates in parallel and independently from the “Master.” Furthermore, because the dynamics of the noise are modeled by a non-correlated random drift vector, it is possible to use a more computationally efficient filter.

Another topic for future work is the development and deployment of the adaptive algorithm on networks with PMUs (synchrophasors). When the algorithm receives measurements from many different PMUs (in the order of several hundreds), the computational time of the adaptive structure must be carefully analyzed and compared to its non-adaptive counterpart and current estimation algorithms. If the computational expense is high due to the extra filter being used, a decision must be made regarding the tradeoffs between computational expense and accuracy of the filter.
Bibliography


