Intelligent Control and Automation, 2015, 6, 168-183
Published Online August 2015 in SciRes. http://www.scirp.org/journal/ica
http://dx.doi.org/10.4236/ica.2015.63017

Sigma-Point Filters in Robotic Applications

Mohammad Al-Shabi

Department of Mechatronics Engineering, Philadelphia University, Amman, Jordan
Email: maqas2002@yahoo.com

Received 1 April 2015; accepted 27 July 2015; published 31 July 2015

Abstract

Sigma-Point Kalman Filters (SPKFs) are popular estimation techniques for high nonlinear system applications. The benefits of using SPKFs include (but not limited to) the following: the easiness of linearizing the nonlinear matrices statistically without the need to use the Jacobian matrices, the ability to handle more uncertainties than the Extended Kalman Filter (EKF), the ability to handle different types of noise, having less computational time than the Particle Filter (PF) and most of the adaptive techniques which makes it suitable for online applications, and having acceptable performance compared to other nonlinear estimation techniques. Therefore, SPKFs are a strong candidate for nonlinear industrial applications, *i.e.* robotic arm. Controlling a robotic arm is hard and challenging due to the system nature, which includes sinusoidal functions, and the dependency on the sensors' number, quality, accuracy and functionality. SPKFs provide with a mechanism that reduces the latter issue in terms of numbers of required sensors and their sensitivity. Moreover, they could handle the nonlinearity for a certain degree. This could be used to improve the controller quality while reducing the cost. In this paper, some SPKF algorithms are applied to 4-DOF robotic arm that consists of one prismatic joint and three revolute joints (PRRR). Those include the Unscented Kalman Filter (UKF), the Cubature Kalman Filter (CKF), and the Central Differences Kalman Filter (CDKF). This study gives a study of those filters and their responses, stability, robustness, computational time, complexity and convergences in order to obtain the suitable filter for an experimental setup.

Keywords

Sigma Point, Unscented Kalman Filter, Cubature Kalman Filter, Centeral Difference Kalman Filter, Filtering, Estimation, Robotic Arm, PRRR

1. Introduction

Robotic applications, especially robotic arm, become widely used in industries due to their simplicity and the ability to do multi-task/multi-function with few numbers of settings and/or arrangements. The problem with

How to cite this paper: Al-Shabi, M. (2015) Sigma-Point Filters in Robotic Applications. *Intelligent Control and Automation*, **6**, 168-183. http://dx.doi.org/10.4236/ica.2015.63017

such applications is the necessary to apply nonlinear control signals to achieve the desired trajectories. The latter is not easy to be implemented and has several limitations [1]-[3]. For example, Sliding Mode Control (SMC) [1] is one of the robust control approaches. However, it suffers from chattering. Although several researches have proposed to eliminate the chattering, the problem is still not fully solved. The limitation of such controllers increases as uncertainties present, *i.e.* modeling uncertainties and noise. This becomes worse when the number of measurement is less than the number of states.

Filters, especially model based filters [2]-[7], have been used to remove some of those constrains. It is a cheap method that could be used to obtain the unmeasured-hidden-states, and/or it could be used to reduce the noise effect. The optimal solution for such applications in their linear case is the Kalman Filter (KF) [7]-[12]. When the system is nonlinear, the KF is modified to be applicable for such applications. Several researches have been developed to overcome this limitation. Those include linearizing the system by Taylor Series Approximation (TSA) up to the first order such as the Perturbation Kalman Filter [9] [13] [14], the Extended Kalman filter (EKF) [8] [15]-[17], and the Iterated Extended Kalman filter (IEKF) [7] [15] [18]-[20], or up to higher order such as the Higher Order Extended Kalman Filter (HOEKF) [15] [21]-[23]. The later shows that in order to increase the accuracy of high nonlinear application, TSA is not a suitable approach as it takes long computation time with complicated structure [24]. Therefore, different approaches were developed including the combination of KF with intelligent techniques such as [25]-[29], or finding different approaches to approximate the nonlinearity such as the Sigma-Point Kalman Filter (SPKF) [2] [4] [5] and the Particle Filter (PF) [30]. The rest of the paper will be divided as the following: Section two includes an introduction to the SPKF including the algorithms used in this paper, UKF, CKF and CDKF. The mathematical model of the PRRR robotic arm application is showed in Section three. Results, discussion and conclusion are listed and discussed in Sections four and five.

2. The Sigma-Point Kalman Filter

The SPKFs linearize the nonlinear models statistically using weighted linear regression method. This is done by obtaining a certain number of points, referred to as sigma points, from the state neighborhood using the probability distribution function as shown in **Figure 1**. Those points are projected through the system model, and then combined together using appropriate weights as shown in **Figure 2**. This provides with a mechanism that covers

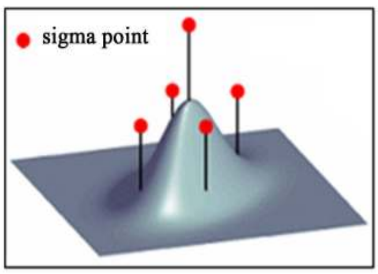

Figure 1. Sigma-Points for n = 2 [31].

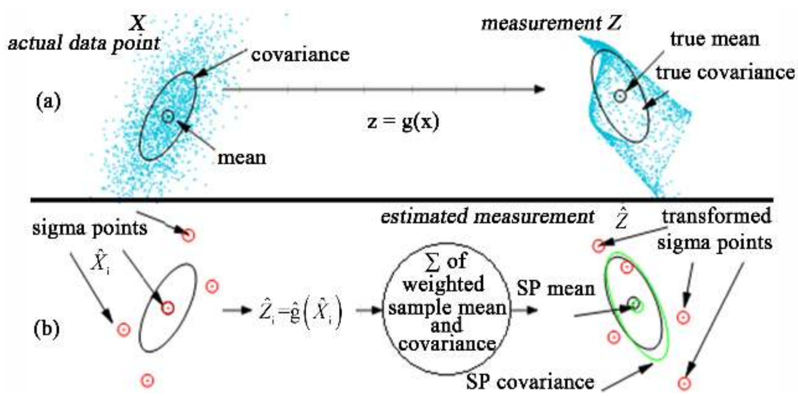

Figure 2. (a) The actual system states and their nonlinear measurement; (b) The Sigma-Points KF's estimates [31].

the actual mean and covariance without the need to linearize the model by TSA and calculate the Jacobian matrices. Moreover, it accommodates noise disturbances that are not Gaussian [4] [5] [15] [31]-[34].

Several algorithms have been created using the above principle. Although, different approaches were used to derive those algorithms, the general outline remain the same as will be proven in the next subsections. The major differences between those methods could be summarized to the number of the sigma points, how to choose them, and what are the appropriate weights for the combining step. Moreover, they may differ on calculating the covariance matrices [35]. Some SPKFs algorithm will be described on the next subsections.

2.1. The Unscented Kalman Filter

The Unscented Kalman Filter is a SPKF that has been developed using the unscented transformations. The latter has several form including ***general unscented*** [15], ***simplex unscented*** [35] [36], and ***spherical unscented*** [36] [37], transformations. The structures of the resulting filters are similar and could be summarized by the pseudo code of **Table 1**, where ϱ_1 and ϱ_2 are parameters used to select the sigma points for the a priori and a posteriori estimates, respectively. Those differ from a filter to another and it result on obtaining different sigma points. Consequently, different number of sigma points and different associated weights are obtained. Those are illustrated by **Table 2**.

Table 1. Thepseudocode of the unscented kalman filter [2] [3] [15] [24].

$k=0 \rightarrow$ Initialize $\hat{\boldsymbol{x}}_{0\|0}$ and $\boldsymbol{P}_{0\|0}$	////Comments
Start $k=k+1$	//// q is the number of the sigma point
for $i=0,1,\cdots,q$	
$\hat{\boldsymbol{X}}_{i_{k-1\|k-1}} = \hat{\boldsymbol{x}}_{k-1\|k-1} + (\varrho_1)_i$	//// draw the sigma points and their weights using **Table 2**
Calculate W_i	
$\hat{\boldsymbol{X}}_{i_{k\|k-1}} = \hat{f}\left(\hat{\boldsymbol{X}}_{i_{k-1\|k-1}}, u_{k-1}\right)$	//// propagate the points through the filter
End	
$\hat{\boldsymbol{x}}_{k\|k-1} = \sum_{i=0}^{q} W_i \hat{\boldsymbol{X}}_{i_{k\|k-1}}$	//// combining the sigma points to obtain the a priori estimate
$\boldsymbol{P}_{k\|k-1} = \sum_{i=0}^{q} W_i \left(\hat{\boldsymbol{X}}_{i_{k\|k-1}} - \hat{\boldsymbol{x}}_{k\|k-1}\right)\left(\hat{\boldsymbol{X}}_{i_{k\|k-1}} - \hat{\boldsymbol{x}}_{k\|k-1}\right)^{\mathrm{T}} + \boldsymbol{Q}_{k-1}$	//// calculating the a priori covariance matrix
for $i=0,1,\cdots,q$	
$\hat{\boldsymbol{X}}_{i_{k\|k-1}} = \hat{\boldsymbol{x}}_{k\|k-1} + (\varrho_2)_i$	//// Redefine the sigma point and their weight from **Table 2** to obtain their a priori measurements
Calculate W_i	
$\hat{\boldsymbol{Z}}_{i_{k\|k-1}} = \hat{\boldsymbol{g}}\left(\hat{\boldsymbol{X}}_{i_{k\|k-1}}\right)$	//// combining the sigma points' measurements to obtain the a priori measurement
End	//// Calculating the output's error covariance matrix
$\hat{\boldsymbol{z}}_{k\|k-1} = \sum_{i=0}^{q} W_i \hat{\boldsymbol{Z}}_{i_{k\|k-1}}$	
$\boldsymbol{P}_{zz} = \sum_{i=0}^{q} W_i \left(\hat{\boldsymbol{Z}}_{i_{k\|k-1}} - \hat{\boldsymbol{z}}_{k\|k-1}\right)\left(\hat{\boldsymbol{Z}}_{i_{k\|k-1}} - \hat{\boldsymbol{z}}_{k\|k-1}\right)^{\mathrm{T}} + \boldsymbol{R}_k$	//// The correction gain //// Updating the estimate and its covariance matrix
$\boldsymbol{P}_{xz} = \sum_{i=0}^{q} W_i \left(\hat{\boldsymbol{X}}_{i_{k\|k-1}} - \hat{\boldsymbol{x}}_{k\|k-1}\right)\left(\hat{\boldsymbol{Z}}_{i_{k\|k-1}} - \hat{\boldsymbol{z}}_{k\|k-1}\right)^{\mathrm{T}}$	//// Repeat Stages
$\boldsymbol{K}_k = \boldsymbol{P}_{xz}\boldsymbol{P}_{zz}^{-1}$	
$\hat{\boldsymbol{x}}_{k\|k} = \hat{\boldsymbol{x}}_{k\|k-1} + \boldsymbol{K}_k\left(z_k - \hat{\boldsymbol{z}}_{k\|k-1}\right)$	
$\boldsymbol{P}_{k\|k} = \left(\boldsymbol{P}_{k\|k-1} - \boldsymbol{K}_k \boldsymbol{P}_{zz} \boldsymbol{K}_k^{\mathrm{T}}\right)$	
Go back to Start	

Table 2. The differences between the UKF methods [15].

Method	$(\varrho_j)_i$, $j=1,2; i=1,2,\cdots,q$	$W_i, i=1,2,\cdots,q$	q
UKF	$(\varrho_j)_i = \begin{cases} 0 & i=0 \\ \left(\sqrt{n\boldsymbol{P}_j}\right)_i^{\mathrm{T}} & 1 \le i \le n \\ -\left(\sqrt{n\boldsymbol{P}_j}\right)_i^{\mathrm{T}} & n+1 \le i \le 2n \end{cases}$ $\boldsymbol{P}_1 = \boldsymbol{P}_{k-1\mid k-1}$, $\boldsymbol{P}_2 = \boldsymbol{P}_{k\mid k-1}$	$W_i = \begin{cases} 0 & i=0 \\ \frac{1}{2n} & i \ne 0 \end{cases}$	$2n+1$
Simplex UKF	$(\varrho_j)_i = \sqrt{\boldsymbol{P}_j}\rho_i^n$, $\boldsymbol{P}_1 = \boldsymbol{P}_{k-1\mid k-1}$, $\boldsymbol{P}_2 = \boldsymbol{P}_{k\mid k-1}$ As ρ_i^n is obtained recursively as follows: $\rho_0^1 = 0$ and $\rho_1^1 = \rho_2^1 = \frac{-1}{\sqrt{2W_1}}$, (the superscript is the recursive index) for $l = 2,\cdots,n$ (number of the states) $\rho_i^l = \begin{cases} \begin{bmatrix} \rho_0^{l-1} \\ 0 \end{bmatrix} & i=0 \\ \begin{bmatrix} \rho_i^{l-1} \\ \frac{-1}{\sqrt{2W_{l+1}}} \end{bmatrix} & 1 \le i \le l \\ \begin{bmatrix} \boldsymbol{0}_{l-1\times 1} \\ \frac{l}{\sqrt{2W_{l+1}}} \end{bmatrix} & i = l+1 \end{cases}$ End	W_0 is chosen as $W_0 \in [0,1)$ $W_i = \begin{cases} 2^{-n}(1-W_0) & 1 \le i \le 2 \\ 2^{i-n}(W_1) & i > 2 \end{cases}$	$n+2$
Spherical UKF	Similar to the simplex UKF except that $\rho_i^l = \begin{cases} \begin{bmatrix} \rho_0^{l-1} \\ 0 \end{bmatrix} & i=0 \\ \begin{bmatrix} \rho_i^{l-1} \\ \frac{-1}{\sqrt{l(l+1)W_1}} \end{bmatrix} & 1 \le i \le l \\ \begin{bmatrix} \boldsymbol{0}_{l-1\times 1} \\ \frac{l}{\sqrt{l(l+1)W_1}} \end{bmatrix} & i = l+1 \end{cases}$	W_0 is chosen as $W_0 \in [0,1)$ $W_i = \frac{1-W_0}{n+1}$	$n+2$

The statistical regression used in unscented filters provides with better approximation that the Jacobian matrices. It has been proven that UKFs approximates up to a third order TSA for Gaussian distributions [15], and second order TSA for non-Gaussian distributions [31]. Both, the simplex and the spherical unscented KFs are used to reduce the computational time; as they use less sigma points. However, their stability is limited for few order of TSA [15] [37]. The general UKF provide with better estimation compared to the previous two. However, it has a larger computational time.

2.2. The Cubature Kalman Filter

The Cubature Kalman filter (CKF) is derived by using the third-degree cubature rule to numerically approximate the Gaussian-weighted integrals defined as [38] [39]:

$$\int_R \boldsymbol{F}(x) W(x) \mathrm{d}x \tag{2.1}$$

where W is the weight function and it is Gaussian with the form $\mathcal{N}\left(x;\bar{x};\sigma\right)$, $\bar{x}$ and σ are the Gaussian's mean and standard deviation. Assuming that the states are Gaussian as well, a scheme similar to the UKF could be obtained. However, due to the Gaussian Nature, the covariance matrices will differ from those obtained from UKF. Those are illustrated by **Table 3**.

2.3. The Central Difference Kalman Filter

The Central Difference Kalman Filter (CDKF), described in [40]-[42], was derived in two major stages. The first stage was to linearize the system model using TSA. In the second stage, the derivatives were replaced with their numerical Stirling's polynomial interpolation forms (NSPI) [43], that is defined as the follow [44]:

Table 3. Thepseudocode of the cubature kalman filter [38] [39].

$k=0\rightarrow$ Initialize $\hat{\boldsymbol{x}}_{0\vert 0}$ and $\boldsymbol{P}_{0\vert 0}$	
Start $k=k+1$	//// Comments
for $i=0,1,\cdots,q$	//// q is the number of the sigma point
$\hat{\boldsymbol{X}}_{i_{k-1\vert k-1}}=\hat{\boldsymbol{x}}_{k-1\vert k-1}+\begin{cases}0 & i=0\\ \left(\sqrt{n\boldsymbol{P}_{k-1\vert k-1}}\right)_i^{\mathrm{T}} & 1\le i\le n\\ \left(\sqrt{n\boldsymbol{P}_{k-1\vert k-1}}\right)_i^{\mathrm{T}} & n+1\le i\le 2n\end{cases}$	//// draw the sigma points
$\hat{\boldsymbol{X}}_{i_{k\vert k-1}}=\hat{\boldsymbol{f}}\left(\hat{\boldsymbol{X}}_{i_{k-1\vert k-1}},u_{k-1}\right)$	
end	//// propagate the points through the filter
$\hat{\boldsymbol{x}}_{k\vert k-1}=\frac{1}{2n}\sum_{i=1}^{q}\hat{\boldsymbol{X}}_{i_{k\vert k-1}}$	//// combining the sigma points to obtain the a priori estimate
$\boldsymbol{P}_{k\vert k-1}=\frac{1}{2n}\sum_{i=1}^{q}\left(\hat{\boldsymbol{X}}_{i_{k\vert k-1}}\hat{\boldsymbol{X}}_{i_{k\vert k-1}}^{\mathrm{T}}-\hat{\boldsymbol{x}}_{k\vert k-1}\hat{\boldsymbol{x}}_{k\vert k-1}^{\mathrm{T}}\right)+\boldsymbol{Q}_{k-1}$	//// calculating the a priori covariance matrix
for $i=0,1,\cdots,q$	
$\hat{\boldsymbol{X}}_{i_{k\vert k-1}}=\hat{\boldsymbol{x}}_{k\vert k-1}+\begin{cases}0 & i=0\\ \left(\sqrt{n\boldsymbol{P}_{k\vert k-1}}\right)_i^{\mathrm{T}} & 1\le i\le n\\ \left(\sqrt{n\boldsymbol{P}_{k\vert k-1}}\right)_i^{\mathrm{T}} & n+1\le i\le 2n\end{cases}$	//// Redefine the sigma point to obtain their a priori measurements
$\hat{\boldsymbol{Z}}_{i_{k\vert k-1}}=\hat{\boldsymbol{g}}\left(\hat{\boldsymbol{X}}_{i_{k\vert k-1}}\right)$	
end	//// combining the sigma points' measurements to obtain the a priori measurement
$\hat{\boldsymbol{z}}_{k\vert k-1}=\frac{1}{2n}\sum_{i=1}^{q}\hat{\boldsymbol{Z}}_{i_{k\vert k-1}}$	//// Calculating the output's error covariance matrix
$\boldsymbol{P}_{zz}=\frac{1}{2n}\sum_{i=1}^{q}\left(\hat{\boldsymbol{Z}}_{i_{k\vert k-1}}\hat{\boldsymbol{Z}}_{i_{k\vert k-1}}^{\mathrm{T}}-\hat{\boldsymbol{z}}_{k\vert k-1}\hat{\boldsymbol{z}}_{k\vert k-1}^{\mathrm{T}}\right)+\boldsymbol{R}_k$	//// The correction gain
$\boldsymbol{P}_{xz}=\frac{1}{2n}\sum_{i=1}^{q}\left(\hat{\boldsymbol{X}}_{i_{k\vert k-1}}\hat{\boldsymbol{Z}}_{i_{k\vert k-1}}^{\mathrm{T}}-\hat{\boldsymbol{x}}_{k\vert k-1}\hat{\boldsymbol{z}}_{k\vert k-1}^{\mathrm{T}}\right)$	//// Updating the estimate and its covariance matrix
$\boldsymbol{K}_k=\boldsymbol{P}_{xz}\boldsymbol{P}_{zz}^{-1}$	//// Repeat Stages
$\hat{\boldsymbol{x}}_{k\vert k}=\hat{\boldsymbol{x}}_{k\vert k-1}+\boldsymbol{K}_k\left(z_k-\hat{\boldsymbol{z}}_{k\vert k-1}\right)$	
$\boldsymbol{P}_{k\vert k}=\left(\boldsymbol{P}_{k\vert k-1}-\boldsymbol{K}_k\boldsymbol{P}_{zz}\boldsymbol{K}_k^{\mathrm{T}}\right)$	
Go back to Start	

$$\partial f^{(n)}(x) = \frac{1}{2}\left(f^{(n-1)}\left(x+\frac{T_s}{2}\right) - f^{(n-1)}\left(x-\frac{T_s}{2}\right)\right) \tag{2.2}$$

The previous stages result on a scheme that is similar to the weighted regression of the UKF as shown in **Table 4**. However, it differs from the UKF on how to obtain the sigma points, how to calculate the weights, and how to calculate the covariance matrices. The CDKF has been found to have a superior performance among the other SPKFs [15] [30] [45]. Moreover, the CDKF uses one control parameter, T_{cd}, which derived in [45] to have a value of $\sqrt{3}$ for Gaussian distributions.

3. PRRR-Mathematical Model

The algorithms in section two are applied to a four DOF robotic arm that consists of one prismatic joint and three revolute joints (PRRR) that is presented by **Figure 3** and **Figure 4**. The model has been derived in [1] and [2], and is summarized as follow.

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{V}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) + \boldsymbol{G}(\boldsymbol{\theta}) \tag{3.1}$$

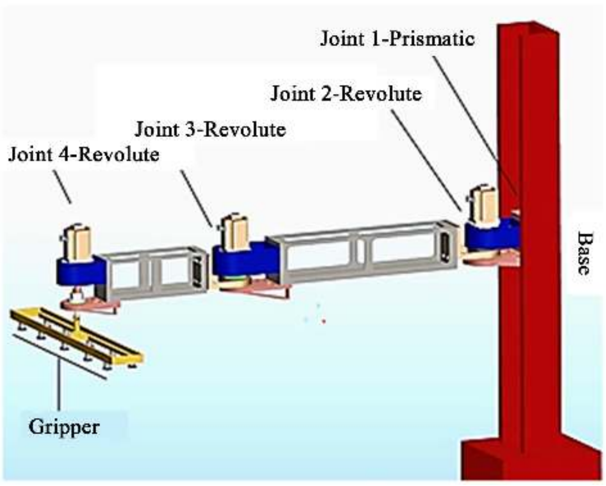

Figure 3. Four-DOFPRRR Robotic Arm [1] [2].

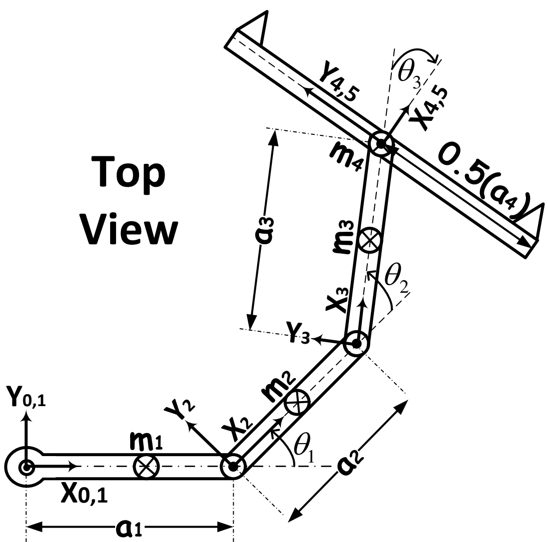

Figure 4. Top view of the PRRR Robotic Arm [1] [2].

Table 4. The pseudocode of sigma-point central difference kalman filter [45].

$k=0 \rightarrow$ Initialize $\hat{\boldsymbol{x}}_{0\|0}$ and $\boldsymbol{P}_{0\|0}$	//// Comments
Start $k=k+1$	
for $i=0,1,\cdots,(q=2n)$	//// draw the sigma points
$\hat{\boldsymbol{X}}_{i_{k-1\|k-1}} = \hat{\boldsymbol{x}}_{k-1\|k-1} + \begin{cases} 0 & i=0 \\ T_{cd}\left(\sqrt{\boldsymbol{P}_{k-1\|k-1}ki}\right)_i^{\mathrm{T}} & 1\le i\le n \\ -T_{cd}\left(\sqrt{\boldsymbol{P}_{k-1\|k-1}}\right)_i^{\mathrm{T}} & n+1\le i\le 2n \end{cases}$ $\hat{\boldsymbol{X}}_{i_{k\|k-1}} = \hat{\boldsymbol{f}}\left(\hat{\boldsymbol{X}}_{i_{k-1\|k-1}}, u_{k-1}\right)$	//// propagate the points through the filter
end	//// combining the sigma points to obtain the a priori estimate
$\hat{\boldsymbol{x}}_{k\|k-1} = \sum_{i=0}^{q}\left[\hat{\boldsymbol{X}}_{i_{k\|k-1}} \times \begin{cases} \frac{T_{cd}^2-n}{T_{cd}^2} & i=0 \\ \frac{1}{2T_{cd}^2} & i\ne 0 \end{cases}\right]$	//// calculating the a priori covariance matrix
$\boldsymbol{P}_{k\|k-1} = \sum_{i=1}^{n}\frac{1}{4T_{cd}^2}\left(\hat{\boldsymbol{X}}_{i_{k\|k-1}} - \hat{\boldsymbol{X}}_{i+n_{k\|k-1}}\right)\left(\hat{\boldsymbol{X}}_{i_{k\|k-1}} - \hat{\boldsymbol{X}}_{i+n_{k\|k-1}}\right)^{\mathrm{T}} + \sum_{i=1}^{n}\frac{T_{cd}^2-1}{4T_{cd}^4}\left(\begin{matrix}\hat{\boldsymbol{X}}_{i_{k\|k-1}} + \hat{\boldsymbol{X}}_{i+n_{k\|k-1}} \\ -2\hat{\boldsymbol{X}}_{0_{k\|k-1}}\end{matrix}\right)\left(\begin{matrix}\hat{\boldsymbol{X}}_{i_{k\|k-1}} + \hat{\boldsymbol{X}}_{i+n_{k\|k-1}} \\ -2\hat{\boldsymbol{X}}_{0_{k\|k-1}}\end{matrix}\right)^{\mathrm{T}} + \boldsymbol{Q}_{k-1}$	//// Redefine the sigma point to obtain their a priori measurements
for $i=0,1,\cdots,q$	
$\hat{\boldsymbol{X}}_{i_{k\|k-1}} = \hat{\boldsymbol{x}}_{k\|k-1} + \begin{cases} 0 & i=0 \\ T_{cd}\left(\sqrt{\boldsymbol{P}_{k\|k-1}}\right)_i^{\mathrm{T}} & 1\le i\le n \\ -T_{cd}\left(\sqrt{\boldsymbol{P}_{k\|k-1}}\right)_i^{\mathrm{T}} & n+1\le i\le 2n \end{cases}$	
$\hat{\boldsymbol{Z}}_{i_{k\|k-1}} = \hat{\boldsymbol{g}}\left(\hat{\boldsymbol{X}}_{i_{k\|k-1}}\right)$	//// combining the sigma points' measurements to obtain the a priori measurement
end	//// Calculating the output's error covariance matrix
$\hat{\boldsymbol{z}}_{k\|k-1} = \sum_{i=0}^{q}\left[\hat{\boldsymbol{Z}}_{i_{k\|k-1}} \times \begin{cases} \frac{T_{cd}^2-n}{T_{cd}^2} & i=0 \\ \frac{1}{2T_{cd}^2} & i\ne 0 \end{cases}\right]$	
$\boldsymbol{P}_{zz} = \sum_{i=1}^{n}\frac{1}{4T_{cd}^2}\left(\hat{\boldsymbol{Z}}_{i_{k\|k-1}} - \hat{\boldsymbol{Z}}_{i+n_{k\|k-1}}\right)\left(\hat{\boldsymbol{Z}}_{i_{k\|k-1}} - \hat{\boldsymbol{Z}}_{i+n_{k\|k-1}}\right)^{\mathrm{T}} + \sum_{i=1}^{n}\frac{T_{cd}^2-1}{4T_{cd}^4}\left(\begin{matrix}\hat{\boldsymbol{Z}}_{i_{k\|k-1}} + \hat{\boldsymbol{Z}}_{i+n_{k\|k-1}} \\ -2\hat{\boldsymbol{Z}}_{0_{k\|k-1}}\end{matrix}\right)\left(\begin{matrix}\hat{\boldsymbol{Z}}_{i_{k\|k-1}} + \hat{\boldsymbol{Z}}_{i+n_{k\|k-1}} \\ -2\hat{\boldsymbol{Z}}_{0_{k\|k-1}}\end{matrix}\right)^{\mathrm{T}} + \boldsymbol{R}_k$	//// The correction gain
$\boldsymbol{P}_{xz} = \frac{1}{2T_{cd}}\sqrt{\boldsymbol{P}_{k\|k-1}}\left(\begin{bmatrix}\hat{\boldsymbol{Z}}_{1_{k\|k-1}}^{\mathrm{T}} \\ \vdots \\ \hat{\boldsymbol{Z}}_{n_{k\|k-1}}^{\mathrm{T}}\end{bmatrix} - \begin{bmatrix}\hat{\boldsymbol{Z}}_{1+n_{k\|k-1}}^{\mathrm{T}} \\ \vdots \\ \hat{\boldsymbol{Z}}_{2n_{k\|k-1}}^{\mathrm{T}}\end{bmatrix}\right)$	//// Updating the estimate and its covariance matrix
$\boldsymbol{K}_k = \boldsymbol{P}_{xz}\boldsymbol{P}_{zz}^{-1}$	//// Repeat Stages
$\hat{\boldsymbol{x}}_{k\|k} = \hat{\boldsymbol{x}}_{k\|k-1} + \boldsymbol{K}_k\left(z_k - \hat{z}_{k\|k-1}\right)$	
$\boldsymbol{P}_{k\|k} = \left(\boldsymbol{P}_{k\|k-1} - \boldsymbol{K}_k\boldsymbol{P}_{zz}\boldsymbol{K}_k^{\mathrm{T}}\right)$	
Go back to Start	

$$\begin{bmatrix} F_z \\ \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} m_T & 0 & 0 & 0 \\ 0 & A_1 & A_4 & A_5 \\ 0 & A_4 & A_2 & A_6 \\ 0 & A_5 & A_6 & A_3 \end{bmatrix} \begin{bmatrix} \ddot{d}_1 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ A_7 \\ A_8 \\ 0 \end{bmatrix} + \begin{bmatrix} -gm_T \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.2)$$

where;

$$A_1 = \left[\frac{1}{4}m_2 a_2^2 + m_3\left(a_2^2 + \frac{a_3^2}{4} + a_2 a_3 c_2\right) + (m_4 + m_5)\left(a_2^2 + a_3^2 + 2a_2 a_3 c_2\right) + \left(I_{z2} + I_{z3} + I_{z4} + I_{z5}\right)\right] \quad (3.3)$$

$$A_2 = \left[\frac{1}{4}m_3 a_3^2 + (m_4 + m_5)a_3^2 + \left(I_{z3} + I_{z4} + I_{z5}\right)\right] \quad (3.4)$$

$$A_3 = A_5 = A_6 = \left[I_{z4} + I_{z5}\right] \quad (3.5)$$

$$A_4 = \left[m_3\left(\frac{a_3^2}{2} + a_2 a_3 c_2\right) + 2(m_4 + m_5)\left(a_3^2 + a_2 a_3 c_2\right) + \left(I_{z3} + I_{z4} + I_{z5}\right)\right] \quad (3.6)$$

$$A_7 = -\left[(m_3 + 2m_4 + 2m_5)\dot{\theta}_1\dot{\theta}_2 + (m_3 + m_4 + 2m_5)\dot{\theta}_2^2\right]a_2 a_3 s_2 \quad (3.7)$$

$$A_8 = -\left[2(m_3 + m_4 + m_5)\dot{\theta}_1\dot{\theta}_2 + \frac{1}{2}(m_3 + 2m_4 + 2m_5)\dot{\theta}_1^2\right]a_2 a_3 s_2 \quad (3.8)$$

$$m_T = m_1 + m_2 + m_3 + m_4 + m_5 \quad (3.9)$$

The system is discretized using the following definition

$$\dot{x}_k = \left(x_{k+1} - x_k\right)/T_s \quad (3.10)$$

where T_s is the sampling time and it is equal to 0.001 sec. If the states defined as the following.

$$\begin{aligned} \boldsymbol{X}_k &= \begin{bmatrix} d_k & \dot{d}_k & \theta_{1k} & \dot{\theta}_{1k} & \theta_{2k} & \dot{\theta}_{2k} & \theta_{3k} & \dot{\theta}_{3k} \end{bmatrix}^{\mathrm{T}} \\ &= \begin{bmatrix} X_{1k} & \dot{X}_{1k} & X_{2k} & \dot{X}_{2k} & X_{3k} & \dot{X}_{3k} & X_{4k} & \dot{X}_{4k} \end{bmatrix}^{\mathrm{T}} \\ &= \begin{bmatrix} X_{1k} & X_{2k} & X_{3k} & X_{4k} & X_{5k} & X_{6k} & X_{7k} & X_{8k} \end{bmatrix}^{\mathrm{T}} \end{aligned} \quad (3.11)$$

And knowing that

$$\begin{bmatrix} \ddot{d}_1 \\ \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix}_k = \begin{bmatrix} m_T & 0 & 0 & 0 \\ 0 & A_1 & A_4 & A_5 \\ 0 & A_4 & A_2 & A_6 \\ 0 & A_5 & A_6 & A_3 \end{bmatrix}_k^{-1} \begin{bmatrix} F_z \\ \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix}_k - \begin{bmatrix} 0 \\ A_7 \\ A_8 \\ 0 \end{bmatrix}_k - \begin{bmatrix} -gm_T \\ 0 \\ 0 \\ 0 \end{bmatrix}_k = \begin{bmatrix} f_1^* \\ f_2^* \\ f_3^* \\ f_4^* \end{bmatrix}_k \quad (3.12)$$

$$\begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}_{k+1} = \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix}_k + T_s\left(\begin{bmatrix} m_T & 0 & 0 & 0 \\ 0 & A_1 & A_4 & A_5 \\ 0 & A_4 & A_2 & A_6 \\ 0 & A_5 & A_6 & A_3 \end{bmatrix}_k^{-1} \begin{bmatrix} F_z \\ \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix}_k - \begin{bmatrix} 0 \\ A_7 \\ A_8 \\ 0 \end{bmatrix}_k - \begin{bmatrix} -gm_T \\ 0 \\ 0 \\ 0 \end{bmatrix}_k\right) \rightarrow \begin{bmatrix} X_2 \\ X_4 \\ X_6 \\ X_8 \end{bmatrix}_{k+1} = \begin{bmatrix} f_2 \\ f_4 \\ f_6 \\ f_8 \end{bmatrix}_k \quad (3.13)$$

$$\begin{bmatrix} d_1 & \theta_1 & \theta_2 & \theta_3 \end{bmatrix}_{k+1}^{\mathrm{T}} = \begin{bmatrix} X_1 & X_3 & X_5 & X_7 \end{bmatrix}_{k+1}^{\mathrm{T}} = \begin{bmatrix} X_1 & X_3 & X_5 & X_7 \end{bmatrix}_k^{\mathrm{T}} + T_s\begin{bmatrix} X_2 & X_4 & X_6 & X_8 \end{bmatrix}_k^{\mathrm{T}} = \begin{bmatrix} f_1 & f_3 & f_5 & f_7 \end{bmatrix}_k^{\mathrm{T}} \quad (3.14)$$

Then the overall state space could be defined as

$$\dot{\boldsymbol{X}}_{k+1} = \dot{\boldsymbol{X}}_k + T_s\begin{bmatrix} f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7 & f_8 \end{bmatrix}_k^{\mathrm{T}} \quad (3.15)$$

Equations (3.3)-(3.9) have several parameters. Those are summarized by **Table 5**.

4. Results

The system in section 3 was simulated several time -for each filter including UKF, CKF and CDKF-. Four cases were obtained as follows:

1. Assuming all the states were measured.
2. Assuming that the position and angles were measured while their derivatives were not measured.
3. Similar to the first case. However, modeling uncertainties were injected; e.g. the masses were multiplied by 1.5.
4. Similar to the second case. However, modeling uncertainties were injected; e.g. the masses were multiplied by 1.5.

4.1. Results for System without Uncertainties; Cases 1 and 2

The results of cases 1 and 2 were summarized by **Table 6** and **Table 7**. The results showed that the filters gave similar performance for all the states when no modeling presented, refer to **Figure 5** and **Figure 6**. The performance of the filters for measured states were better than those obtained for non-measured states.

4.2. Results with Uncertainties

When modeling errors presented, the RMSE increased as shown in **Table 8** and **Table 9**. However, their effect became large, and maybe unstable, for the states that were not measured as shown in **Figure 7**. In such cases, the CDKF showed the superior performance; the filter remained stable. However, the UKF and CKF had a poor performance. The errors were bounded. However, they were high, refer to **Figures 8-10**.

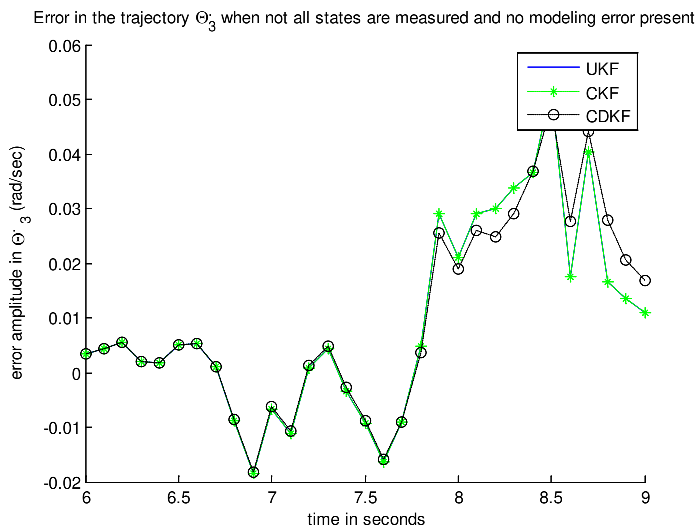

Figure 5. The performance of the filters for the third angler velocity, cases 1 and 2.

Table 5. Parameters' Value for the robotic arm.

Parameter	Value	Parameter	Value	Parameter	Value
m_1	21.5 kg	I_1	$1.042\ \text{kg}\cdot\text{m}^2$	a_1	0.25 m
m_2	16 kg	I_2	$13\ \text{kg}\cdot\text{m}^2$	a_2	1.2 m
m_3	8.5 kg	I_3	$3.12\ \text{kg}\cdot\text{m}^2$	a_3	0.8m
m_4	7.9 kg	I_4	$1\ \text{kg}\cdot\text{m}^2$	a_4	1.2
m_5	6.3 kg	I_5	$0.84\ \text{kg}\cdot\text{m}^2$	g	$-9.81\ \text{m/s}^2$

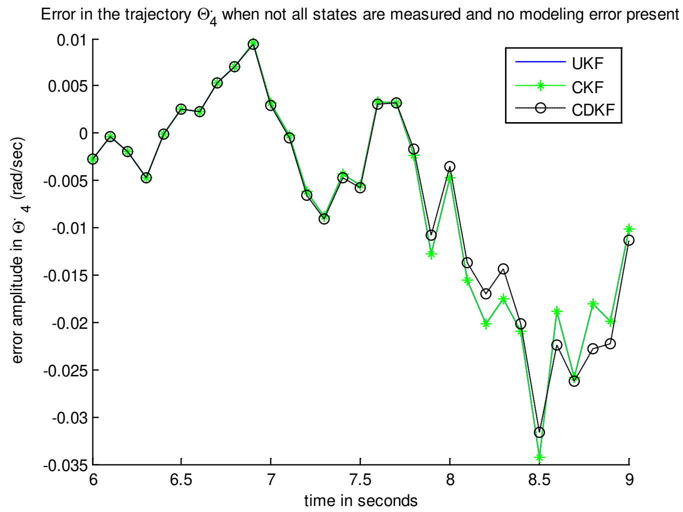

Figure 6. The performance of the filters for the fourth angler velocity, cases 1 and 2.

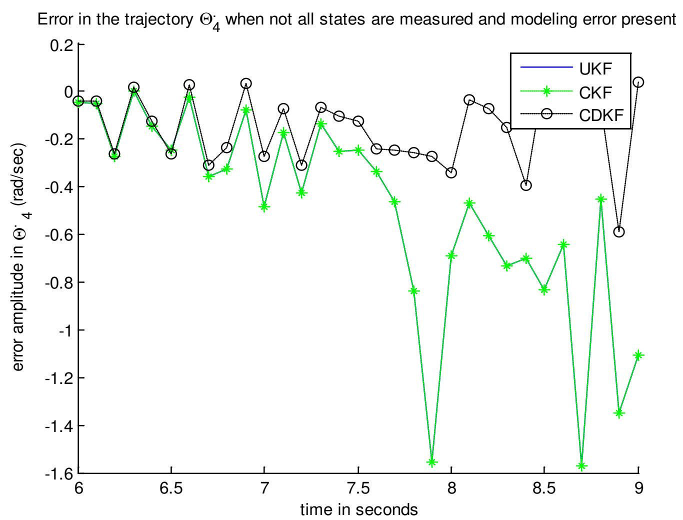

Figure 7. The performance of the filters for the fourth angler velocity, case 4.

Table 6. The root mean square error for the filters UKF, CKF and CDKF for case 1.

RMS in	UKF $\times 10^{-6}$	CKF $\times 10^{-6}$	CDKF $\times 10^{-6}$
d	32.7	32.7	32.7
$\dot{d}$	29.2	29.2	29.2
θ_1	31	31	31
$\dot{\theta}_1$	35.8	35.8	35.8
θ_2	22.2	22.2	22
$\dot{\theta}_2$	46.1	46.1	46.1
θ_3	25.6	25.6	25.6
$\dot{\theta}_3$	44.1	44.1	44.1

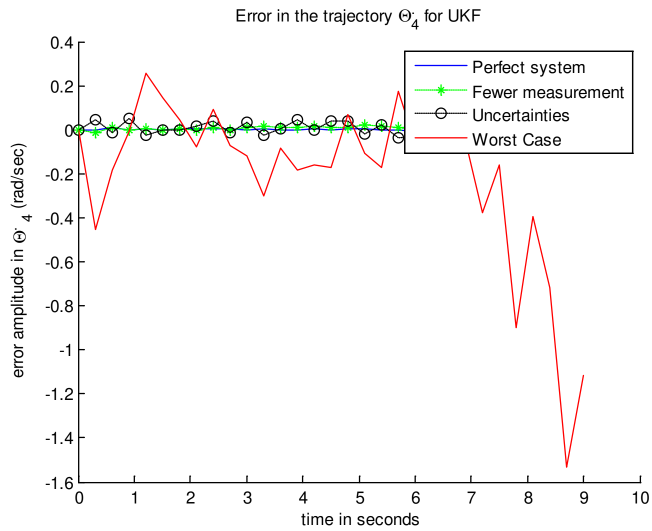

Figure 8. The error in estimating the fourth angular velocity using UKF for all cases.

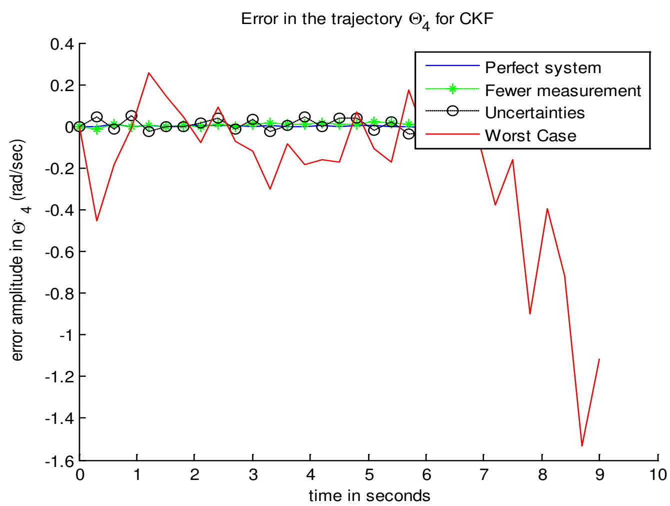

Figure 9. The error in estimating the fourth angular velocity using CKF for all cases.

Table 7. The root mean square error for the filters UKF, CKF and CDKF for case 2.

RMS in	UKF $\times 10^{-6}$	CKF $\times 10^{-6}$	CDKF $\times 10^{-6}$
d	35.9	35.9	35.9
$\dot{d}$	203.4	203.4	203.4
θ_1	41.4	41.4	41.3
$\dot{\theta}_1$	162.8	162.8	161.6
θ_2	30.6	30.6	31.9
$\dot{\theta}_2$	171.4	171.4	171.1
θ_3	49.2	49.2	49.1
$\dot{\theta}_3$	291.7	291.7	291.4

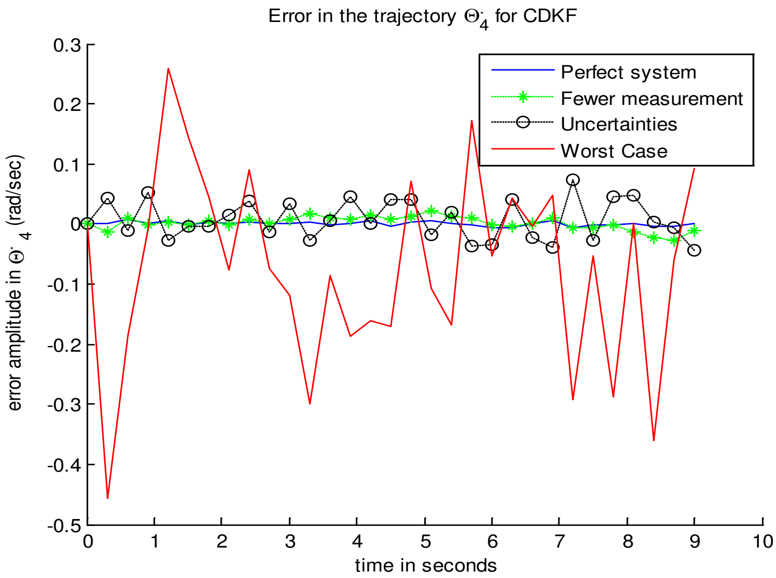

Figure 10. The error in estimating the fourth angular velocity using CDKF for all cases.

Table 8. The root mean square error for the filters UKF, CKF and CDKF for case 3.

RMS in	UKF $\times 10^{-6}$	CKF $\times 10^{-6}$	CDKF $\times 10^{-6}$
d	44.6	44.6	44.6
$\dot{d}$	375.8	375.8	375.8
θ_1	44.5	44.5	44.5
$\dot{\theta}_1$	367.4	367.4	367.4
θ_2	34.9	34.9	34.9
$\dot{\theta}_2$	365.1	365.1	365.1
θ_3	38.9	38.9	38.9
$\dot{\theta}_3$	366.5	366.5	366.5

Table 9. The root mean square error for the filters UKF, CKF and CDKF for case 4.

RMS in	UKF $\times 10^{-6}$	CKF $\times 10^{-6}$	CDKF $\times 10^{-6}$
d	146.9	146.9	146.8
$\dot{d}$	4588.5	4588.5	4588.9
θ_1	111.3	111.3	103.8
$\dot{\theta}_1$	2710.8	2710.8	1693.2
θ_2	120.3	120.3	112.2
$\dot{\theta}_2$	5590.7	5590.7	2447.4
θ_3	133.3	133.3	109.3
$\dot{\theta}_3$	4079.6	4079.6	1982.2

5. Conclusion

This work discussed the benefits of using Sigma-Point Kalman Filters in nonlinear application, *i.e.* PRRR robotic arm. Three types of SPKFs were used, namely Unscented, Cubature, and Central difference Kalman Filters. Four cases were used: the first and the second cases involved with system with no modeling errors; the third and the fourth cases involved with system injected with uncertainties. The first and the third cases assumed all the states were measured which was not the case in the other cases. The results showed that the filters gave good performance when all the states were measured. Reducing the number of measurements affected the results a little bit. The errors became larger than 10 times of those obtained in case 1 when modeling errors were presented and not all the states were measured. However, the CDKF showed stable performance in all cases. The latter gave an indication to use the CDKF in such applications.

References

[1] Hatamleh, K., Al-Shabi, M., Khasawneh, Q.A. and Al-Asal, M.A. (2014) Application of SMC and NLFC into a PRRR Robotic. *ASME* 2014 *International Mechanical Engineering Congress and Exposition*, Montreal, 14-20 November 2014, Paper No. IMECE2014-39136.

[2] Al-Shabi, M. and Hatamleh, K. (2014) The Unscented Smooth Variable Structure Filter Application into a Robotic Arm. *ASME* 2014 *International Mechanical Engineering Congress and Exposition*, Montreal, 14-20 November 2014, Paper No. IMECE2014-40118. http://dx.doi.org/10.1115/imece2014-40118

[3] Al-Shabi, M., Hatamleh, K. and Asad, A. (2013) UAV Dynamics Model Parameters Estimation Techniques: A Comparison Study. 2013 *IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies*, Amman, 3-5 December 2013.

[4] Ali, Z., Deriche, M. and Landolsi, M. (2009) Sigma Point Kalman Filters For Multipath Xhannel Estimation In CDMA Networks. *Proceedings of the* 2009 6*th International Symposium on Wireless Communication Systems*, Tuscany, 7-10 September 2009, 423-427. http://dx.doi.org/10.1109/ISWCS.2009.5285296

[5] Ambadan J. and Tang, Y.M. (2009) Sigma-Point Kalman Filter Data Assimilation Methods for Strongly Non-Linear Systems. *Journal of the Atmospheric Sciences*, **66**, 261-285. http://dx.doi.org/10.1175/2008JAS2681.1

[6] Anderson B. and Moore, J. (1979) Optimal Filtering. Prentice-Hall.

[7] Bar-Shalom, T. Li X.and Kirubarajan, T. (2001) Estimation with Applications to Tracking and Navigation—Theory, Algorithm and Software. John Wiley & Sons, Inc. http://dx.doi.org/10.1002/0471221279

[8] Grewal M. and Andrews, A. (2001) Kalman Filtering—Theory and Practice Using MATLAB. John Wiley & Sons, Inc.

[9] Barker, A., Brown, D. and Martin, W. (1995) Bayesian estimation and the Kalman Filter. *Computers & Mathematics with Applications*, **30**, 55-77.

[10] Welch, G. and Bishop, G. (2006) An Introduction to the Kalman Filter. Department of Computer Science, University of North Carolina, Chapel Hill, TR 95-041.

[11] Maybeck, P. (1979) Stochastic Models, Estimation, and Contro. Mathematics in Science and Engineering, Volume 141, Part 1, Academic Press, Waltham, iii-xix, 1-423.

[12] Kalman, R. (1960) A New Approach to Linear Filtering and Prediction Problems. *ASME Journal of Basic Engineering*, **82**, 35-45. http://dx.doi.org/10.1115/1.3662552

[13] Ormsby, C., Raquet, J. and Maybeck, P. (2006) A Generalized Residual Multiple Model Adaptive Estimator of Parameters and States. *Mathematical and Computer Modelling*, **43**, 1092-1113. http://dx.doi.org/10.1016/j.mcm.2005.12.003

[14] Negenborn, R. (2003) Robot Localization and Kalman Filters—On Finding Your Position in a Noisy World. MS Thesis, Utrecht University, Utrecht.

[15] Simon, D. (2006) Optimal State Estimation: Kalman, H [Infinity] and Nonlinear Approaches. Wiley-Interscience.

[16] Lary, D. and Mussa, H. (2004) Using an Extended Kalman Filter Learning Algorithm for Feed-Forward Neural Networks to Describe Tracer Correlations. *Atmospheric Chemistry and Physics Discussion*, **4**, 3653-3667. http://dx.doi.org/10.5194/acpd-4-3653-2004

[17] Leu, G. and Baratti, R. (2000) An Extended Kalman Filtering Approach with a Criterion to Set Its Tuning Parameters—Application to a Catalytic Reactor. *Computers & Chemical Engineering*, **23**, 1839-1849. http://dx.doi.org/10.1016/S0098-1354(00)00298-2

[18] Shojaie, K., Ahmadi, K. and Shahri, A. (2007) Effects of Iteration in Kalman Filter Family for Improvement of Estimation Accuracy in Simultaneous Localization and Mapping. *IEEE/ASME International Conference on Advanced In-*

telligent Mechatronics, Zurich, 4-7 September 2007, 1-6. http://dx.doi.org/10.1109/AIM.2007.4412453

[19] Zhang, Y., Zhou, D. and Duan, G. (2006) An Adaptive Iterated Kalman Filter. *IMACS Multiconference on Computational Engineering in Systems Applications*, Beijing, 4-6 October 2006, 1727-1730. http://dx.doi.org/10.1109/CESA.2006.4281916

[20] Hyland, J. (2002) An Iterated-Extended Kalman Filter Algorithm for Tracking Surface and Sub-Surface Targets. *OCEANS'02 MTS/IEEE*, **3**, 1283-1290. http://dx.doi.org/10.1109/oceans.2002.1191824

[21] Dungate, D., Theobald, R. and Nurse, F. (1999) Higher-Order Kalman Filter to Support Fast Target Tracking in a Multi-Function Radar System. *IEE ColloquiumTarget Tracking: Algorithms and Applications*, London, 11-12 November 1999, 14/1-14/3.

[22] Bayard, D. and Kang, B. (2003) A High-Order Kalman Filter for Focal Plane Calibration of NASA's Space Infrared Telescope Facility (SIRTF). *AIAA Guidance, Navigation and Control Conference and Exhibit*, Austin, 11-14 August 2003. http://dx.doi.org/10.2514/6.2003-5824

[23] Athans, M., Wishner, R. and Bertolini, A. (1968) Suboptimal State Estimation for Continuous-Time Nonlinear Systems from Discrete Noisy Measurements. *IEEE Transactions on Automatic Control*, **13**, 504-514. http://dx.doi.org/10.1109/TAC.1968.1098986

[24] Al-Shabi, M. (2012) The General Toeplitz/Observability Smooth Variable Structure Filter: Fault Detection and Parameter Estimation. LAP Lambert Academic Publishing, Saarbrücken.

[25] Nguyen, H. and Walker, E. (1996) A First Course in Fuzzy Logic. CRC Press, Boca Raton.

[26] Yager, R. and Zadeh, L. (1992) An Introduction to Fuzzy Logic Applications in Intelligent Systems. Kluwer Academic, location.

[27] Carrasco, R., Cipriano, A. and Carelli, R. (2005) Nonlinear State Estimation in Mobile Robots Using a Fuzzy Observer. *Processing of the 16th IFAC World Congress*, Vol. 16, Part 1, Czech Republic.

[28] Simon, D. (2003) Kalman Filtering for Fuzzy Discrete Time Dynamic Systems. *Applied Soft Computing Journal*, **3**, 191-207. http://dx.doi.org/10.1016/S1568-4946(03)00034-6

[29] Matia, F., Jimenez, A., Rodriguez-Losada, D. and Al-Hadithi, B.M. (2004) A Novel Fuzzy Kalman Filter for Mobile Robots Localization. *IPMU 2004, 10th International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems,* Volume II, Perugia.

[30] Chen, Z. (2003) Bayesian Filtering: From Kalman Filters to Particles and Beyond. McMaster University.

[31] Van Der Merwe, R. and Wan, E. (2004) Sigma-Point Kalman Filters for Integrated Navigation. *Proceedings of the Annual Meeting*, Institute of Navigation, 641-654.

[32] Wang, L., Wang, L., Liao, C. and Liu, J. (2009) Sigma-Point Kalman Filter Application on Estimating Battery SOC. *5th IEEE Vehicle Power and Propulsion Conference*, Dearborn, 7-10 September 2009, 1592-1595.

[33] Sadhu, S., Mondal, S., Srinivasan, M. and Ghoshal, T. (2006) Sigma Point Kalman Filter for Bearing Only Tracking. *Signal Processing*, **86**, 3769-3777. http://dx.doi.org/10.1016/j.sigpro.2006.03.006

[34] Schenkendorf, R., Kremling, A. and Mangold, M. (2009) Optimal Experimental Design with the Sigma Point Method. *IET Systems Biology*, **3**, 10-23. http://dx.doi.org/10.1049/iet-syb:20080094

[35] Tang, X., Zhao, X. and Zhang, X. (2008) The Square-Root Spherical Simplex Unscented Kalman Filter for State and Parameter Estimation. *9th International Conference on Signal Processing*, Beijing, 26-29 October 2008, 260-263.

[36] Kim, J. and Shin, D. (2005) Joint Estimation of Time Delay and Channel Amplitude by Simplex Unscented Filter without Assisted Pilot in CDMA Systems. *The 7th International Conference on Advanced Communication Technology*, **1**, 233-238.

[37] Julier, S. (2003) The Spherical Simplex Unscented Transformation. *Proceedings of the American Control Conference*, **3**, 2430-2434. http://dx.doi.org/10.1109/acc.2003.1243439

[38] Gadsden, S., Al-Shabi, M., Arasaratnam, I. and Habibi, S. (2010) Estimation of an Electrohydrostatic Actuator Using a Combined Cubature Kalman and Smooth Variable Structure Filter. *International Mechanical Engineering Congress and Exposition (IMECE)*, American Society of Mechanical Engineers, Vancouver, British Columbia.

[39] Gadsden, A., Al-Shabi, M., Arasaratnam, I. and Habibi, S. (2014) Combined Cubature Kalman and Smooth Variable Structure Filtering: A Robust Nonlinear Estimation Strategy. *Signal Processing*, **96**, 290-299. http://dx.doi.org/10.1016/j.sigpro.2013.08.015

[40] Nrgaard, M., Poulsen, N. and Ravn, O. (2000) New Developments in State Estimation for Nonlinear Systems. *Automatica*, **36**, 1627-1638. http://dx.doi.org/10.1016/S0005-1098(00)00089-3

[41] Zhang, U., Gao, F. and Tian, L. (2008) INS/GPS Integrated Navigation for Wheeled Agricultural Robot Based on Sigma-Point Kalman Filter. 2008 *Asia Simulation Conference-7th International Conference on System Simulation and*

Scientific Computing, Beijing, 10-12 October 2008, 1425-1431. http://dx.doi.org/10.1109/ASC-ICSC.2008.4675598

[42] Zhu, J.H., Zheng, N.N., Yuan, Z.J. and Zhang, Q. (2009) A SLAM Algorithm Based on the Central Difference Kalman Filter. 2009 *IEEE Intelligent Vehicles Symposium*, Xi'an, 3-5 June 2009, 123-128. http://dx.doi.org/10.1109/IVS.2009.5164264

[43] Sadati, N. and Ghaffarkhah, A. (2007) POLYFILTER: A New State Estimation Filter for Nonlinear Systems. *International Conference on Control*, *Automation and Systems*, Seoul, 17-20 October 2007, 2643-2647.

[44] Henrici, P. (1964) Elements of Numerical Analysis. John Wiley and Sons, New York.

[45] Van Der Merwe, R. (2004) Sigma Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models. PhD Thesis, OGI School of Science & Engineering, Oregon Health & Science University, USA.

Nomenclature

$^{-1}$, $^{\mathrm{T}}$	Inverse, and transpose, respectively.
$(\boldsymbol{a})_i$	The i row of $\boldsymbol{a}$.
a_{i-1}, α_{i-1}, d_i and θ_i	Link-i's length (m), twist (rad), and offset (m), and joint- -i angle (rad), respectively.
c_i and s_i	$\cos(\theta_i)$ and $\sin(\theta_i)$, respectively.
c_{ij} and s_{ij}	$\cos(\theta_i+\theta_j)$ and $\sin(\theta_i+\theta_j)$, respectively.
$\boldsymbol{e}_m$	The estimation error vectors in m.
$\boldsymbol{f}(.)$	The system's model function.
F_z and τ_i	Prismatic joint-1 motor force (N) and Revolute joint-i motor torque (N. M), respectively.
g	Gravity acceleration (m/s^2).
$\boldsymbol{g}(.)$	The sensor's model function.
i, j	Subscripts used to identify elements.
$\boldsymbol{I}_{n\times n}$	The identity matrix with dimensions of $n \times n$.
k	Time step value.
$k\|k-1$	The a priori value at time k.
$k\|k$	The a posteriori value at time k.
$\boldsymbol{K}_X$	The correction gain of the filter X.
$\boldsymbol{M}(\Theta)$	Inertia matrix.
$m_1, m_2, \cdots, m_5$	Masses of links 1, 2, 3 and 4 respectively (kg).
m, n	Number of measurements and states, respectively.
$\boldsymbol{P}_{xx}$	The state's error covariance matrix.
$\boldsymbol{P}_{zz}$	The output's error covariance matrix.
$\boldsymbol{P}$	The error covariance matrix.
q	The number of the sigma points.
$\boldsymbol{Q}$	The process noise covariance matrix.
$\boldsymbol{R}$	The measurements noise covariance matrix.
$\sum$	The summation operator.
T_s	Sampling time, and is equal to 0.001 sec.
$\boldsymbol{\tau}$	Joints force and torques vector.
$V(\Theta,\dot{\Theta})$	Viscous friction vector.
v, w	The measurement and system noise, respectively.
W_i	The assigned weight.
x	The state vector.
z	The output vector.
$\boldsymbol{X}_i$ and $\boldsymbol{Z}_i$:	The estimate and its measurement for the i^{th} sigma point, respectively.